NOAA TECHNICAL MEMORANDUM NWS CR-68

THE POSTING OF SHEF DATA TO THE RFC GATEWAY DATABASE

Geoffrey M. Bonnin
Missouri Basin River Forecast Center
National Weather Service
Kansas City, Missouri

April 1983

U.S. DEPARTMENT OF COMMERCE / National Oceanic and Atmospheric Administration / National Weather Service

NOAA TECHNICAL MEMORANDA
National Weather Service, Central Region Subseries

The National Weather Service Central Region (CR) subseries provides an informal medium for the documentation and quick dissemination of results not appropriate, or not yet ready, for formal publication. The series is used to report on work in progress, to describe technical procedures and practices, or to relate progress to a limited audience. These Technical Memoranda will report on investigations devoted primarily to regional and local problems of interest mainly to regional personnel, and hence will not be widely distributed.

Papers 1 to 15 are in the former series, ESSA Technical Memoranda, Central Region Technical Memoranda (CRTM); papers 16 to 36 are in the former series, ESSA Technical Memoranda, Weather Bureau Technical Memornada (WBTM). Beginning with #37, the papers are now part of the series, NOAA Technical Memoranda NWS.

Papers that have a PB or COM number are available from the National Technical Information Service, U. S. Department of Commerce, 5285 Port Royal Road, Springfield, VA. 22151. Order by accession number shown in parenthesis at the end of each entry. All other papers are available from the National Weather Service Central Region, Scientific Services, Room 1836, 601 E. 12th St. Kansas City, MO 64106.

ESSA Technical Memoranda

CRTM 1 Precipitation Probability Forecast Verification Summary Nov. 1965-Mar. 1966. SSD Staff, WBCRH - May 1966
CRTM 2 A Study of Summer Showers Over the Colorado Mountains. W. G. Sullivan and James O. Severson - June 1966
CRTM 3 Areal Shower Distribution - Mountain Versus Valley Coverage. Wm C. Sullivan and James O. Severson - June 1966
CRTM 4 Heavy Rains in Colorado June 16 and 17, 1965. SSD Staff, WBCRH - July 1966
CRTM 5 The Plum Fire. Wm. G. Sullivan - August 1966
CRTM 6 Precipitation Probability Forecast Verification Summary Nov. 1965-July 1966. SSD Staff, WBCRH - September 1966
CRTM 7 Effect of Diurnal Weather Variations on Soybean Harvest Efficiency. Leonard F. Hand - October 1966
CRTM 8 Climatic Frequency of Precipitation at Central Region Stations. SSD Staff, WBCRH - November 1966
CRTM 9 Heavy Snow or Glazing. Harry W. Waldheuser - December 1966
CRTM 10 Detection of a Weak Frong by WSR-57 Radar. Harry W. Waldheuser - December 1966
CRTM 11 Public Probability Forecasts. SSD Staff, WBCRH - January 1967
CRTM 12 Heavy Snow Forecasting in the Central United States (An Interim Report). SSD Staff - January 1967
CRTM 13 Diurnal Surface Geostrophic Wind Variations Over the Great Plains. Wayne E. Sangster - March 1967
CRTM 14 Forecasting Probability of Summertime Precipitation at Denver. Wm. G. Sullivan and James O. Severson - March 1967
CRTM 15 Improving Precipitation Probability Forecasts Using the Central Region Verification Printout. Lawrence A. Hughes - May 1967
WBTM CR 16 Small-Scale Circulations Associated with Radiational Cooling. Jack R. Cooley - June 1967
WBTM CR 17 Probability Verification Results (6-month and 18-month) Lawrence A. Hughes - June 1967
WBTM CR 18 On the Use and Misuse of the Brier Verification Score. Lawrence A. Hughes - August 1967 (PB 175 771)
WBTM CR 19 Probability Verification Results (24 Months). Lawrence A. Hughes - February 1968
WBTM CR 20 Radar Prediction of the Topeka Tornado. Norman E. Prosser - April 1968
WBTM CR 21 Wind Waves on the Great Lakes. Lawrence A. Hughes - May 1968
WBTM CR 22 Seasonal Aspects of Probability Forecasts: 1. Summer. Lawrence A. Hughes - June 1968 (PB185 733)
WBTM CR 23 Seasonal Aspects of Probability Forecasts: 2. Fall. Lawrence A. Hughes - September 1968 (PB 185 734)
WBTM CR 24 The Importance of Areal Coverage in Precipitation Probability Forecasting. John T. Curran and Lawrence A. Hughes - Sept. 1968
WBTM CR 25 Meteorological Conditions as Related to Air Pollution Chicago, Illinois, April 12-13, 1963. Charles H. Swan - October 1968
WBTM CR 26 Seasonal Aspects of Probability Forecasts: 3. Winter. Lawrence A. Hughes - December 1968 (PB 185 735)
WBTM CR 27 Seasonal Aspects of Probability Forecasts: 4. Spring. Lawrence A. Hughes - February 1969. (PB 185 736)
WBTM CR 28 Minimum Temperature Forecasting During Possible Frost Periods at Agricultural Weather Stations in Western Michigan. Marshall A. Soderberg March 1969
WBTM CR 29 An Aid for Tornado Warnings. Harry W. Waldheuser and Lawrence A. Hughes - April 1969
WBTM CR 30 An Aid in Forecasting Significant Lake Snows. H.J. Rothrock - November 1969
WBTM CR 31 A Forecast Aid for Boulder Winds. Wayne E. Sangster - February 1970
WBTM CR 32 An Objective Method for Estimating the Probability of Severe Thunderstorms. Clarence L. David - February 1970
WBTM CR 33 Kentucky Air-Soil Temperature Climatology. Clyde B. Lee - February 1970
WBTM CR 34 Effective Use of Non-Structural Methods in Water Management. Verne Alexander - March 1970
WBTM CR 35 A Note on the Categorical Verification of Probability Forecasts. Lawrence A. Hughes and Wayne E. Sangster - August 1970
WBTM CR 36 A Comparison of Observed and Calculated Urban Mixing Depths. Donald E. Huerch - August 1970

NOAA TECHNICAL MEMORANDA NWS

NWS CR 37 Forecasting Maximum and Minimum Surface Temperatures at Topeka, Kansas, Using Guidance from the PE Numerical Prediction Model. (FOUS) Morris S. Webb - November 1970 (COM-71-00118)
NWS CR 38 Snow Forecasting for Southeastern Wisconsin. Rheinhart W. Harms - November 1970 (COM-71-00019)
NWS CR 39 A Synoptic Climatology of Blizzards on the North-Central Plains of the United States. Robert E. Black - Feb. 1971 (COM-71-00369)
NWS CR 40 Forecasting the Spring 1969 Midwest Snowmelt Floods. Herman F. Monschein - February 1971 (COM-71-00489)
NWS CR 41 The Temperature Cycle of Lake Michigan 1. (Spring and Summer). Lawrence A. Hughes - April 1971 (COM-71-00545)
NWS CR 42 Dust Devil Meteorology. Jack R. Cooley - May 1971 (COM-71-00628)
NWS CR 43 Summer Shower Probability in Colorado as Related to Altitude. Alois G. Topil. May 1971 (COM-71-00712)
NWS CR 44 An Investigation of the Resultant Transport Wind Within the Urban Complex. Donald E. Huerch - June 1971 (COM-71-00766)
NWS CR 45 The Relationship of Some Cirrus Formations to Severe Local Storms. William E. Williams - July 1971 (COM-71-00844)
NWS CR 46 The Temperature Cycle of Lake Michigan 2. (Fall and Winter). Lawrence A. Hughes - September 1971 (COM-71-01039)
NWS CR 47 Practical Application of a Graphical Method of Geostrophic Wind Determination. C.B. Johnson - November 1971 (COM-71-01084)

(continued on back inside cover)

NOAA TECHNICAL MEMORANDUM NWS CR-68

THE POSTING OF SHEF DATA TO THE RFC GATEWAY DATABASE

Geoffrey M. Bonnin
Missouri Basin River Forecast Center
National Weather Service
Kansas City, Missouri

April 1983

UNITED STATES
DEPARTMENT OF COMMERCE
Malcolm Baldrige, Secretary

National Oceanic and
Atmospheric Administration
John V. Byrne, Administrator

National Weather
Service
Richard E. Hallgren, Director

THE POSTING OF SHEF DATA TO THE RFC GATEWAY DATABASE

Geoffrey M. Bonnin[a]
Missouri Basin River Forecast Center
National Weather Service
Kansas City, Missouri

ABSTRACT

Standard Hydrologic Exchange Format (SHEF) was developed as a prerequisite for automated hydrologic data exchange. This paper describes software developed for automatically posting data encoded in SHEF to the database system used in the River Forecast Center Gateway computer system.

[a]Hydrologist (Computer Systems), B.E. (Univ. of Qld.), M.I.E., Aust.

TABLE OF CONTENTS

APPENDICES

1. INTRODUCTION

Standard Hydrologic Exchange Format (SHEF) (1) is being adopted by the National Weather Service as the standard format for the encoding of hydrologic data. SHEF is sufficiently flexible to handle most hydro-meteorologic data and as a result, is being adopted by a variety of programs within NWS as well as elements of the U.S. Army Corps of Engineers and the U.S. Geological Survey.

SHEF has been designed to allow for the automation of data handling techniques at the same time as maintaining a format which is visually readable.

Proposed uses for SHEF within the Central Region of NWS have been described in a previous NOAA Technical Memorandum (2). These uses can be summarized conceptually as either manual or automatic encoding of data into SHEF, the transmission of SHEF on communications circuits, the reception and either manual or automatic decoding of SHEF and the posting of the data within a database.

The ability to automatically post data to a database is an outgrowth of the introduction of SHEF and is a replacement of earlier methods of data handling.

This Technical Memorandum will describe the process of posting data to the River Forecast Center DATACOL (3) database.

2. PARSING AND POSTING

The process of taking data in SHEF and putting it in a database can be described as a two step process; parsing and posting. The parsing step takes SHEF and reduces it to a simplified, machine readable format. The posting process takes the reduced data and interacts with the target database in such a way that the data is integrated into the database.

This conceptualization of the process allows software to be developed for the parsing step which is essentially standard and is not database specific thus allowing for a high degree of portability. The software for the posting process is database specific. It is fed the output from the parsing process and is independent of the original SHEF message.

The Missouri Basin RFC has developed such a suite of software. PARSHEF (2) is a standard package for parsing SHEF. It produces a file (SHEFOUT) which is the reduced SHEF data. It is being adopted by NWS as the standard front end parsing package for a number of database and data manipulation systems.

DATACOL is a real time data collection and database management system developed by the California-Nevada RFC for use by NWS RFC's in their Gateway computer systems. MBRFC has developed software for DATACOL which activates the PARSHEF parsing software, reads the file SHEFOUT and then posts the data to the database.

3. DATACOL

DATACOL (3, 4, 5) is a real time data collection and database management system designed for use by NWS RFC's in their Gateway computer systems (Data General S/140 under MRDOS).

The database stores stochastically sequential data for different sensors and different sites in separate, preallocated blocks of a single file. There are separate files which cross reference the site name (ALPHAINDEX) or its NWS I.D. (CODEKEY) with the database block number.

The data is stored on the file with its associated observation time in a packed integer format. All other information relating to the data is stored in a series of header words at the beginning of the block and is fixed for all of the data within the block. The basic description of the type of data is by a numeric code for "sensor type".

This database structure implies that there must be a one to one correspondence between SHEF Parameter codes and database block numbers. Obviously, it is not possible to preallocate a block for every possible parameter code. This restriction has led to a strategy of providing for the possibility of storing within DATACOL, all of the physical elements possible in SHEF, but only observed data for their default durations. The Source code is ignored and those data with Extremum or Probability codes are rejected. There are a few exceptions to this strategy which is described more fully in 6.D.

DATACOL provides a number of utilities for reading and writing the database file as well as packing and unpacking the data and reading the block header.

4. SHEFOUT

This file is the "pipe" for passing the reduced data from the parsing step to the posting step. It is a sequential, unformatted file, with each record being a self sufficient and complete description of an item of data and its attributes. All times on the file are Greenwich Mean Time and all data is in English units as prescribed in SHEF.

Each record of the file is written with the following statement:

```
WRITE (JCHN) IDSTN, NYEAR, NMON, NDAY, NHOUR, NMIN,
1                   KYEAR, KMON, KDAY, KHOUR, KMIN,
2           KODP, KODE, IDUR, KODT, KODS, KODEX, CODP
3           VALUE, LWAL, IREV, MSOURCE, IDOTE
```

The list variables are defined as follows:

IDSTN: Eight element Integer array, containing the eight characters of the station identifier. If there are fewer than eight characters, they are stored starting in the first array element with the remaining elements containing a blank.

NYEAR: Integer, containing the last two digits of the year of the observation date.

NMON: Integer, containing the month number of the observation date. January is month number one.

NDAY: Integer, containing the day of the month of the observation date.

NHOUR: Integer, containing the hour of the day of the observation date on a 24 hour clock.

NMIN: Integer, containing the minute of the observation date.

KYEAR: The creation date of the data coded in the same manner as the
KMON: observation date. If there was no creation date specified,
KDAY: all elements are set to zero.
KHOUR:
KMIN:

KODP: Integer, containing the first character of the Physical Element code.

KODE: Integer, containing the second element of the Physical Element code.

IDUR: Integer, containing a coded number which represents the duration. This duration code specifies the units of time and the number of units as follows:

0XXX	minutes
1XXX	hours
2XXX	days
3XXX	months
4XXX	years
5000	unspecified
5001	seasonal
5002	entire period of record
5004	time period beginning at 7AM local time prior to the observation time and ending at the observation time
5005	unknown
6XXX	months - end of month

where XXX is the number of units. i.e., eight days would be coded as 2008, instantaneous as 0.

KODT: Integer, containing the Type code character.

KODS: Integer, containing the Source code character.

KODEX: Integer, containing the Extremum code character.

CODP: Real, containing the value of the Probability code as a decimal i.e., fifty percent probability is coded as 0.5. The mean (P code of M) is coded as -0.5. If the Probability code is unspecified, it is coded as -1.0.

VALUE: Double Precision Real, containing the value of the piece of data.

LWAL: Integer, containing the Data Qualifier character.

IREV: Integer, set to one if this is data is intended as a revision of previous data. It is set to zero otherwise.

MSOURCE: Eight element Integer array, containing the eight characters of the data source (.B format). The characters are coded in the same way as those in IDSTN. The source is blank for .A and .E format.

IDOTE: Integer, set to one if the data item is the first item in a string of time series data, set to two if it is a subsequent time series item, set to zero otherwise.

5. POSTING TO DATACOL

The DATACOL system, after sensing the need to parse and post, goes to a routine AFSDEC which drives the process.

The parsing software is started as an independent program in another ground of the computer. After it has finished and the SHEFOUT file has been created, the file is read record by record. As each record is read, the following process occurs:

A. The station I.D. is tested to see if it is a stranger station. If so, a cautionary note is printed and this record is discarded.

B. The databank block number for this station is found from the cross reference file and a relationship between SHEF Parameter codes and DATACOL sensor types. If the station sensor is unknown to DATACOL, a cautionary note is printed and the record is discarded.

C. A check is then made to see if data exists for that observation time. If it does and the new data does not have the revision code set a cautionary note is printed and this record is discarded.

D. Finally, the data is inserted into the correct position in the databank using DATACOL utilities.

E. The process then begins again at the next record in SHEFOUT and continues until the end of the file is reached.

Errors and cautionary notes from both PARSHEF and the posting process are accumulated on a single error log file AFSDEC.ER. If there are any messages generated for this file by either parsing or posting, the file is printed on the hard copy device.

6. SOFTWARE MODULES

Those modules which have been developed for the posting process will be described as well as those modules from the standard parsing package (2) which have been modified for DATACOL.

A. AFSDEC

Checkpoints the parsing package PARSHEF into background, communicates with it via interground communication pipes and waits for its completion. It then starts the posting process by calling SHFILE. AFSDEC is started by AFSMST, the control module which activates various processes for AFOS products.

B. SHFILE

The file SHEFOUT is read record by record and tests are made for stranger stations (as defined in SHEF). The databank block number is obtained with SHTATN and the data is filed with SHDBAS.

Checks are then made to see if errors have occurred, if so, the error log file is printed on the hard copy device defined in the DATACOL BLOCKDATA module.

C. SHTATN

File CODEKEY is an alphabetical list of NWS I.D.s for stations recognized by the database. The actual I.D. and sensor type is maintained on file SENSORLIST. SHTATN searches, by interval halving, for a match of the NWS I.D. and then for the correct sensor type on matching I.D.s, finally returning the correct databank block number for this piece of data.

D. SHSENS

SHSENS provides a translation between SHEF Parameter codes and DATACOL sensor types.

DATACOL cannot store data in all the "flavors" provided for in SHEF. For this reason, there has been a relationship set between SHEF parameter codes and DATACOL sensor types. This relationship is as follows:

remember: The generic SHEF parameter code is PEDTSEP where

PE = physical element
D = duration
T = type
S = source
E = extremum
P = probability

(a) Type; only observed data is accepted.
(b) Source; Source is ignored.
(c) Extremum; anything with an extremum code set is rejected except for max and min temperatures.
(d) Probability; anything with a probability is rejected.
(e) The data is stored only if the Duration is the default Duration for that Physical Element, except for those parameter codes listed below.
(f) Physical Element; the PE code is translated to sensor type using the formula (P-1)*26+E+49 where P & E are the position of the letter in the alphabet. i.e. sensor types 50 through 725.

(g) Data Qualifier; the Data Qualifier is ignored.

Exceptions: The following are exceptions to the rules (a) through (f).

SHEF parameter code		DATACOL sensor type
TAIRZXZ	Max temp	43
TAIRZNZ	Min temp	42
QIDRZZZ	Mean daily inflow	49
QTDRZZZ	Mean daily outflow	48
QIQRZZZ	Mean 6 hour inflow	47
QTQTZZZ	Mean 6 hour outflow	46
PPPRZZZ	Precip for a duration beginning at previous 7am local time and ending at observation time	45
PPQRZZZ	6 hourly precip	44 ***
HGIRZZZ	Stage	4

NOTES: (a) Z is used here to imply that anything other than Z is rejected.

*** (b) While the software exists in SHSENS to allow this parameter code (PPQRZZZ), it is commented out. To allow 6 hourly precip to be posted, SHSENS must be edited and recompiled and DATACOL must be reloaded.

The default duration codes are read into the COMMON block SHPARM at the time DATACOL is started. They are read from the file SHEFPARM (2).

E. SHCOMP

Compares names for the interval halving process.

F. SHDBAS

Files the data in the database if possible. The observation time of the data to be filed is converted to a two year Julian minute format, the database block is read and the header information is extracted.

The file is searched to find the correct position for the new data. If valid data already exists for this time and no revision code is set, the new data is rejected.

If there is any more recent data existing, it is written to a temporary file. The new data is converted to the database format and is passed to the filing task. If there was any more recent data written to the temporary file, it is passed to the filing task for reentry to the database.

G. SHMSG

The error reporting module for posting. It makes an error number, reads the appropriate error message from disk and writes it to the error log file.

H. PARSHEF

The mainline module for parsing. Modifications have been made to the standard version (2) to allow for interground communications with DATACOL and for different file names.

I. SHERR

The error reporting module for parsing. Modifications have been made to the standard version (2) to allow the actual error message rather than its number to be written on the error log.

ERRATA:
page 6; change QTQTZZZ to QTQRZZZ

7. REFERENCES

1. "Standard Hydrologic Exchange Format (SHEF)" Version 1; Northwest River Forecast Center, November 18, 1982

2. "An explanation of the Standard Hydrologic Exchange Format (SHEF) and Its Implementation in the Central Region", NOAA Technical Memorandum NWS CR-67; Geoffrey M. BONNIN, Robert S. COX, Jr., April 1983

3. DATACOL Rev. 7.20; David C. LEADER January 1983

4. "User Manual for DATACOL"; David C. LEADER, August 1981

5. "Program Configuration Guide for DATACOL" Rev. 7.20; David C. LEADER, October 1982

APPENDIX A. - Parsing Error Messages

1. This line not decoded
2. No space in positional data
3. Less than 3 characters in I.D. or message source
4. Time Zone code error
5. Date group error
6. Illegal character in I.D. or message source
7. Error in date code
8. Observation time error
9. Date relative code error
10. Julian day error
11. Illegal data string qualifier
12. Units code error
13. Not a date or data type element
14. Not a date or data type element, maybe a missing slash
15. Illegal character in parameter code
16. File read error on SHEFPARM
17. Non-existant parameter code
18. Parameters coded with a send code
19. Continuation of a format does not follow the correct format
20. A format revision continuation follows an original
21. The format that this is continuing had an error
22. Year not in the range 1976-1999 for default time zone
23. Forecast data without creation date
24. Bad date somehow
25. Variable duration not defined
26. Variable duration code error
27. Time increment code error
28. Trace specified for other than PY, PP, PC, SD, SF and SW
29. No time increment specified
30. To many values in .B body line
31. Bad character in the line
32. Not enough values in .B body line
33. No value specified
34. No .END at end of .B format
35. Zulu, DR or DI coded with send code QY, PY or HY
36. Explicit date referenced by DRE or DIE is not the end of month
37. Time between 0201 & 0259 on date of change from standard to daylight

APPENDIX B - Posting Error Messages

1. This is a stranger station
2. Station or sensor does not exist
3. Trying to revise data without revision code set
4. DATACOL will not accept this parameter code

APPENDIX C

SOURCE CODE

```
      OVERLAY MOAFSDEC
      SUBROUTINE AFSDEC(PRODCT,FNAME)
C
C-----------------------------------------------------------------------C
C   VERSION 1.0 SEP  1982     GEOFFREY M BONNIN  MBRFC                  C
C-----------------------------------------------------------------------C
      INTEGER FNAME(4),PRODCT(6)
      COMMON /LUNS/   ICHN,JCHN
      COMMON /IGCA/ IGCM
C
C.... FILE NAMES
C
      DIMENSION PARSHEF(3),SHEFOUT(2)
      PARSHEF(1) = 'PARS'
      PARSHEF(2) = 'HEF.'
      PARSHEF(3) = 'SV'
      SHEFOUT(1) = 'SHEF'
      SHEFOUT(2) = 'OUT'
C
C.... START THE DECODER IN BACKGROUND
C
      IGCM = -1
   10 CALL EXBG (PARSHEF,0,IER)
      IF( IER.EQ.1 ) GO TO 20
      CALL FDELY(50)                          ;WAIT 5 SEC
        TYPE "PARSHEF CHECKPOINT ERROR",IER
      GO TO 10
C
C.... WAIT ON DECODER
C
   20 IF( IGCM.EQ.-2 ) GO TO 25
      CALL FDELY(5)                           ;WAIT FOR PARSHEF TO START
      GO TO 20
C
   25 CALL WRCMN(PRODCT,0,6,IER)              ;PASS PRODUCT NAME TO PARSHEF
      CALL WRCMN(FNAME,6,4,IER)               ;PASS FILE NAME
      IFLAG = 1
      CALL WRCMN(IFLAG,10,1,IER)              ;PASS READY FLAG
      CALL FDELY(50)
   30 IF( IGCM.NE.-2 ) GO TO 40
      CALL FDELY(10)                          ;WAIT FOR PARSHEF TO FINISH
      GO TO 30
C
C.... START POSTING
C
   40 CALL FDELY(20)                          ;WAIT A FEW MORE SECONDS
      OPEN ICHN,SHEFOUT
      CALL SHFILE
      IGCM = 0                                ;RESTORE TO NEUTRAL VALUE
C
C.... THAT'S IT
C
      RETURN
      END
```

```
      SUBROUTINE SHFILE
C
C-----------------------------------------------------------------C
C   VERSION 1.0 SEP  1982      GEOFFREY M BONNIN  MBRFC           C
C-----------------------------------------------------------------C
C
C      This routine reads decoded SHEF format data from the
C      file "SHEFOUT" and files it in the database.
C
      DOUBLE PRECISION VALUE
      INTEGER ILINE(80)
      COMMON /CONON/ ICONON
      COMMON /CAFOS/ LAFOS,JBTX,JETX,IFILE,ICHPP,FNAMEPRT
      COMMON /LUNS/    ICH,JCHN
      COMMON /POSTERR/ NERROR
      COMMON /POSTDATA/ IDSTN(8),NYEAR,NMON,NDAY,NHOUR,NMIN,
     1                        KYEAR,KMON,KDAY,KHOUR,KMIN,
     2                  KODP,KODE,IDUR,KODT,KODS,KODEX,CODP,
     3                  VALUE,LWAL,IREV,MSOURCE(8),IDOTE
      COMMON /CODES/ ICHA,ICHB,ICHC,ICHD,ICHE,ICHF,ICHG,ICHH,ICHI,
     1               ICHJ,ICHK,ICHL,ICHM,ICHN,ICHO,ICHP,ICHQ,ICHR,
     2               ICHS,ICHT,ICHU,ICHV,ICHW,ICHX,ICHY,ICHZ,ICH0,
     3               ICH1,ICH2,ICH3,ICH4,ICH5,ICH6,ICH7,ICH8,ICH9,
     4               IBLNK,ISLASH,ICOLON,IPLUS,IMINUS,IDOT,IARROW,
     5               ICOMMA
      COMMON /IGCA/ IGCM
C
C.... FILE NAMES AND ATTRIBUTES
C
      DIMENSION AFSDEC(3)
      AFSDEC(1) = 'AFSD'
      AFSDEC(2) = 'EC.E'
      AFSDEC(3) = 'R'
      ATTR1 = 'SA'
      ATTR2 = 'S'
C
C.... INITIALISE ERROR COUNT
C
      NERROR = 0
C
C.... READ A RECORD FROM "SHEFOUT"
C
   10 READ(ICH,END=50) IDSTN,NYEAR,NMON,NDAY,NHOUR,NMIN,
     1                        KYEAR,KMON,KDAY,KHOUR,KMIN,
     2                  KODP,KODE,IDUR,KODT,KODS,KODEX,CODP,
     3                  VALUE,LWAL,IREV,MSOURCE,IDOTE
C
C.... TEST FOR STRANGER STATION
C
      IF( IDSTN(1).NE.ICHW .AND.
     1    IDSTN(1).NE.ICHX .AND.
     2    IDSTN(1).NE.ICHY .AND.
     3    IDSTN(1).NE.ICHZ ) GO TO 40
      DO 20 I=2,8
         CALL IRANG(IDSTN(I),ICH0,ICH9,$40)
   20 CONTINUE
C
C.... THIS IS A STRANGER STATION
C
   30 GO TO 910
C
C.... SEARCH FOR THE STATION FILE NUMBER IN "CODEKEY"
C
```

```
   40 CALL SHTATN (ISTN,$10)
C
C.... NOW FILE THE DATA
C
      CALL SHDBAS(ISTN,$10)
      GO TO 10
C
C.... THAT'S IT
C
   50 OPEN JCHN,AFSDEC,ATT=ATTR1
      WRITE(JCHN,60) NERROR
   60 FORMAT(/'NUMBER OF POSTING ERRORS IS',I3)
      CALL CLOSE(JCHN,IER)
      CALL CLOSE(ICH,IER)
C
C.... NOW DEAL WITH THE ERROR LOG
C
      IF( (NERROR+IGCM).EQ.0 ) GO TO 900
C
C.... PRINT THE LOG
C
      OPEN JCHN,AFSDEC,ATT=ATTR2
C
C.... IF PRINT TERMINAL NOT CONSOLE - OPEN CHANNEL TO FILENAME
C
      IF(ICHPP.EQ.10) GO TO 90                ;SKIP IF CONSOLE TERMINAL
      CALL CFILW(FNAMEPRT,1,IER)              ;CREATE FILE IF NECESSARY
      OPEN ICHPP,FNAMEPRT,ERR=930             ;OPEN DESIGNATED CHANNEL
      GO TO 100
C
C BID FOR CONSOLE TERMINAL CONTROL
C
   90 ICNT = 0
   91 CALL XMT(ICONON,1,$920)
C
C.... READ A LINE AND PRINT IT
C
  100 CALL RLINE(JCHN,ILINE,NB,IER)           ;READ LINE
      IF(IER.EQ.9) GO TO 110                  ;STOP IF EOF
      IF(IER.NE.1) GO TO 940                  ;SKIP IF ERROR
      CALL WLINE(ICHPP,ILINE,NB,IER)          ;WRITE LINE
      IF(IER.NE.1) GO TO 940                  ;SKIP IF ERROR
      GO TO 100
C
C.... IF NOT CONSOLE OUTPUT - CLOSE CHANNEL
C
  110 CALL CLOSE(JCHN,IER)
      IF(ICHPP.EQ.10) GO TO 120
      CALL CLOSE(ICHPP,IER)
      GO TO 900
C
C.... RELEASE CONSOLE CONTROL
C
  120 CALL REC(ICONON,IER)
C
C.... NORMAL RETURN
C
  900 RETURN
C
C.... ERRORS ..............................
C
C.... STRANGER STATION
C
```

```
  910 CALL SHMSG(1)
      GO TO 10
C
C UNABLE TO GET CONSOLE TERMINAL CONTROL - WAIT THEN TRY AGAIN (UP TO 10 TIMES)
  920 ICNT=ICNT+1
      IF(ICNT.GT.100) GO TO 925              ;SKIP AFTER 10 TRIES
      CALL FDELY(10)                         ;WAIT A SECOND
      GO TO 91                               ;JUMP TO TRY AGAIN
C
  925 CALL CLOSE(JCHN,IER)
      GO TO 900
C
C.... FILE ERROR
C
  930 CALL GETERR(IER)
  940 IF(ICHPP.EQ.10) CALL REC(ICONON,JER)
      CALL CONMSG('*** AFOS FILE PRINT ERROR ***',IER)
      IF(ICHPP.NE.10) CALL CLOSE(ICHPP,IER)
      GO TO 900
C
      END
```

```
      SUBROUTINE SHTATN(ISTN,$)
C
C-----------------------------------------------------------------C
C   VERSION 1.0 SEP  1982     GEOFFREY M BONNIN  MBRFC            C
C-----------------------------------------------------------------C
C
C     This routine sets the database file number for the station
C     whose NWS identifier is in IDSTN and returns it in ISTN.
C     The sensor type is derived from the PE code in KODP and
C     KODE. The RETURN 2 is used if the station is not recognised.
C
      PARAMETER ICHA = '<000>A'
      DOUBLE PRECISION VALUE
      DIMENSION NWS(8)
      COMMON /MAXSC/ MAXS,IOVS,NESTAB
      COMMON /CSENS/ JCHSEN,JSENFLG
      COMMON /LUNS/    ICHN,JCHN
      COMMON /POSTDATA/ IDSTN(8),NYEAR,NMON,NDAY,NHOUR,NMIN,
     1                        KYEAR,KMON,KDAY,KHOUR,KMIN,
     2                  KODP,KODE,IDUR,KODT,KODS,KODEX,CODP,
     3                  VALUE,LWAL,IREV,MSOURCE(8),IDOTE
C
C.... FILE NAMES AND ATTRIBUTES
C
      DIMENSION CODEKEY(2)
      CODEKEY(1) = 'CODE'
      CODEKEY(2) = 'KEY'
      ATTR1 = 'C'
C
C.... GET THE DATA'S SENSORTYPE
C
      CALL SHSENS(ISENS,$75)
C
C.... TRY TO HOME IN ON THE NAME BY INTERVAL HALVING
C
C      - SET THE LIMITS
C
      IH = NESTAB
      IL = 1
      IT = (NESTAB+1)/2
C
C.... OPEN THE CODEKEY FILE WITH THE INDEX OF NWS ID'S
C
      OPEN JCHN,CODEKEY,ATT=ATTR1,LEN = 2
C
C.... GET THE TRIAL ID
C
   10 READ(JCHN,REC=IT) ISTN
      CALL REC(JSENFLG,IER)
      READ(JCHSEN,20,REC=ISTN) NWS,ITYPE
   20 FORMAT(28X,8A1,19X,I3)
      CALL XMT(JSENFLG,1,$25)
C
C.... NOW COMPARE THE IDENTIFIER
C
   25 CALL SHCOMP(IDSTN,NWS,ITEST,$30)
C
C.... INTERVAL HALVE
C
      CALL IHALF(IT,IH,IL,ITEST)
      IF( IH.LT.IL ) GO TO 70                    ;FINISHED NO MATCH
      GO TO 10
C
```

```
C.... GOT A HIT, TEST FOR SENSOR TYPE HERE AND
C     ON EITHER SIDE IN 'CODEKEY'.
C
   30 IF( ISENS.EQ.ITYPE ) GO TO 80
      INC = 1
      IH = IT
C
   40 IT = IT + INC
      READ(JCHN,REC=IT,ERR=60) ISTN
      CALL REC(JSENFLG,IER)
      READ(JCHSEN,20,REC=ISTN) NWS,ITYPE
      CALL XMT(JSENFLG,1,$45)
C
C.... COMPARE
C
   45 CALL SHCOMP(IDSTN,NWS,ITEST,$50)
      IF( INC.EQ.-1 ) GO TO 70
      GO TO 60
C
C.... COMPARE SENSOR
C
   50 IF( ISENS.EQ.ITYPE ) GO TO 80
      GO TO 40
C
C.... TRY THE OTHER DIRECTION
C
   60 INC = -1
      IT = IH
      GO TO 40
C
C.... THERE IS NO SUCH STATION
C
   70 CALL CLOSE(JCHN,IER)
      CALL SHMSG(2)
   75 IRTN = 2
      GO TO 100
C
C.... GOT THE STATION NUMBER
C
   80 IRTN = 0
C
C.... CLOSE CODEKEY
C
   90 CALL CLOSE(JCHN,IER)
C
C.... RETURN
C
  100 IF( IRTN.NE.0 ) RETURN IRTN
      RETURN
C
      END
```

```
      SUBROUTINE SHSENS(ISENS,$)
C
C-----------------------------------------------------------------C
C   VERSION 1.0 JAN  1983      GEOFFREY M BONNIN  MBRFC           C
C-----------------------------------------------------------------C
C
C       Get the correct DATACOL sensor type from the parameter code.
C
      DOUBLE PRECISION VALUE
      COMMON /LUNS/    ICH,JCHN
      COMMON /SHPARM/ IPARM(25,2),JPARM(26)
      COMMON /POSTDATA/ IDSTN(8),NYEAR,NMON,NDAY,NHOUR,NMIN,
     1                        KYEAR,KMON,KDAY,KHOUR,KMIN,
     2                  KODP,KODE,IDUR,KODT,KODS,KODEX,CODP,
     3                  VALUE,LWAL,IREV,MSOURCE(8),IDOTE
      COMMON /CODES/ ICHA,ICHB,ICHC,ICHD,ICHE,ICHF,ICHG,ICHH,ICHI,
     1               ICHJ,ICHK,ICHL,ICHM,ICHN,ICHO,ICHP,ICHQ,ICHR,
     2               ICHS,ICHT,ICHU,ICHV,ICHW,ICHX,ICHY,ICHZ,ICHO,
     3               ICH1,ICH2,ICH3,ICH4,ICH5,ICH6,ICH7,ICH8,ICH9,
     4               IBLNK,ISLASH,ICOLON,IPLUS,IMINUS,IDOT,IARROW,
     5               ICOMMA
C
C.... INITIALISE ISENS
C
      ISENS = 0
C
C.... ACCEPT ONLY OBSERVED DATA
C
      IF( KODT.NE.ICHR ) GO TO 910
C
C.... ACCEPT ANYTHING FOR SOURCE
C
C       ......
C
C.... REJECT ANYTHING WITH A PROBABILITY
C
      IF( CODP.NE.-1.0 ) GO TO 910
C
C.... ACCEPT ANY QUALIFIER
C
C       ......
C
C.... REJECT ANYTHING WITH AN EXTREMUM EXCEPT MAX & MIN TEMPS
C     TAIRZXZ & TAIRZNZ
C
      IF( KODEX.EQ.ICHZ ) GO TO 30
C
      IF( KODP.NE.ICHT ) GO TO 910
      IF( KODE.NE.ICHA ) GO TO 910
C
      IF( KODEX.EQ.ICHX ) GO TO 10
      IF( KODEX.EQ.ICHN ) GO TO 20
C
   10 ISENS = 43                  ;TAIRZXZ
      GO TO 900
C
   20 ISENS = 42                  ;TAIRZNZ
      GO TO 900
C
C.... GET THE DEFAULT DURATION
C
   30 II = (KODP-ICHA)*26 + (KODE-ICHA) + 1
      DO 40 I=1,25
```

```
         IF( IPARM(I,1).EQ.II ) GO TO 50
   40 CONTINUE
      IVAL = 0
      GO TO 55
C
   50 II = IPARM(I) - ICHA + 1
      IVAL = JPARM(II)
C
C.... TEST FOR DEFAULT DURATION
C
   55 IF( IDUR.NE.IVAL ) GO TO 60
C
C.... YES; SENSOR NUMBER IS (P-1)*26 + E + 49
C
      ISENS = (KODP-ICHA)*26 + (KODE-ICHA) + 50
C
C.... EXCEPT FOR HGIRZZZ WHICH IS #4
C
      IF( KODP.NE.ICHH ) GO TO 900
      IF( KODE.NE.ICHG ) GO TO 900
      ISENS = 4
      GO TO 900
C
C.... CERTAIN SENSOR TYPES THAT AREN'T DEFAULT DURATION ARE ALLOWED
C     QIDRZZZ            DAILY INFLOW              #49
C     QTDRZZZ            DAILY OUTFLOW             #48
C     QIQRZZZ            6 HOURLY INFLOW           #47
C     QTQRZZZ            6 HOURLY INFLOW           #46
C
   60 IF( KODP.NE.ICHQ ) GO TO 90
      IF( KODE.EQ.ICHI ) GO TO 80
      IF( KODE.EQ.ICHT ) GO TO 70
      GO TO 910
C
   70 ISENS = -1
   80 IF( IDUR.EQ.2001 ) ISENS = ISENS + 44
      IF( IDUR.EQ.1024 ) ISENS = ISENS + 44
      IF( IDUR.EQ.1006 ) ISENS = ISENS + 47
      IF( ISENS.LE.0 ) GO TO 910
      GO TO 900
C
C      PPPRZZZ  PRECIP SINCE PREV 7AM            #45
C      PPQRZZZ  6 HOURLY PRECIP                  #44
C
   90 IF( KODP.NE.ICHP ) GO TO 910
      IF( KODE.NE.ICHP ) GO TO 910
C
      IF( IDUR.EQ.5004 ) ISENS = 45
C
CCCCCCC REMOVE COMMENTS FROM THE FOLLOWING CODE IF YOU WANT
CCCCCCC PPQRZZZ TO BE POSTED.
C     IF( IDUR.EQ.1006 ) ISENS = 44              ;NOT FOR MBRFC
      IF( ISENS.LE.0   ) GO TO 910
      GO TO 900
C
C.... GOT IT
C
  900 CALL CLOSE(JCHN,IER)
      RETURN
C
C.... SENSOR DOESN'T EXIST
C
  910 CALL CLOSE(JCHN,IER)
```

```
      CALL SHMSG(4)
      RETURN 2
C
      END
```

```
      SUBROUTINE SHCOMP(IDSTN,NWS,ITEST,$)
C-------------------------------------------------------------C
C   VERSION 1.0  SEP 1982  GEOFFREY M BONNIN  MBRFC           C
C-------------------------------------------------------------C
C
C       Compare names for a match.
C       RETURN 2 if no match, normal RETURN with ITEST set for
C       interval halving if no match.
C       IDSTN has the characters in the right byte with a null in
C       the left. NWS was read with an A format and so a shift from
C       left to right byte is required for compatability.
C
      DIMENSION IDSTN(8),NWS(8)
C
      DO 10 I=1,8
         NWS(I) = ISHFT(NWS(I),-8)
         IF( IDSTN(I).EQ.NWS(I) ) GO TO 10
         IF( IDSTN(I).GT.NWS(I) ) ITEST = -1
         IF( IDSTN(I).LT.NWS(I) ) ITEST =  1
         GO TO 20
   10 CONTINUE
C
C.... GOT A MATCH
C
      RETURN 4
C
C.... NO MATCH
C
   20 RETURN
C
      END
```

```
      SUBROUTINE SHDBAS(ISTN,$)
C
C-----------------------------------------------------------------------C
C   VERSION 1.0 OCT  1982     GEOFFREY M BONNIN  MBRFC                  C
C-----------------------------------------------------------------------C
C
C       This routine files the data
C
      DOUBLE PRECISION VALUE
      COMMON /LUNS/    ICHN,JCHN
      COMMON /POSTDATA/ IDSTN(8),NYEAR,NMON,NDAY,NHOUR,NMIN,
     1                        KYEAR,KMON,KDAY,KHOUR,KMIN,
     2                  KODP,KODE,IDUR,KODT,KODS,KODEX,CODP,
     3                  VALUE,LWAL,IREV,MSOURCE(8),IDOTE
      COMMON /DF/ DFNAME
      COMMON /CWNDW/ IAW(1024)
C
C.... FILE NAMES
C
      DIMENSION SHEFWORK(3)
      SHEFWORK(1) = 'SHEF'
      SHEFWORK(2) = 'WORK'
      SHEFWORK(3) = '<000><000><000><000>'
C
      JREV = 0
C
C.... GET THE TWO YEAR JULIAN MINUTE FOR THIS ENTRY AND THE NEXT
C
      CALL JULHC(NMON,NDAY,NHOUR,JULH)
      RMIN = JULH*60.0 + NMIN
C
C.... OPEN AN RDOS CHANNEL TO THE DATABASE
C
      CALL AOPEN(ICHDB,DFNAME,0,IER)
C
C.... GET THE TASKS WINDOW BLOCK ASSIGNMENT
C
      CALL MYWDR(MYWD)
      IWB = MYWD*4
C
C.... READ THE STATIONS SENSOR FILE
C
      NBLK = (ISTN-1)*2
      CALL AERDB(ICHDB,NBLK,IWB,2,IER)
C
C.... GET THE DATA FILE HEADER INFO
C
      CALL FILMA(IAW,RDIV,RADD)          ;CONVERSION TERMS
      MPT = IAW(6)-2                     ;LAST ENTRY
      IF( MPT.EQ.7 ) GO TO 105           ;ANY DATA ON FILE?
C
C.... SEARCH THE FILE FOR THE ENTRY BEFORE RMIN
C
      DO 100 I=MPT,9,-2
         CALL DECODE(IAW(I),IAW(I+1),IHF,IMNF,IDF,IMF,IDATA)
         CALL JULHC(IMF,IDF,IHF,JULF)
         RMINF = JULF*60.0 + IMNF
         IF( RMINF.LE.RMIN ) GO TO 110
  100 CONTINUE
C
C.... THIS IS THE OLDEST DATA IN THE PRIMARY FILE
C
  105 IPT = 9
```

```
      GO TO 140
C
C.... CHECK IF VALID DATA ALREADY EXISTS
C
  110 IF( RMINF.NE.RMIN ) GO TO 130
      IF( IREV.EQ.1 ) GO TO 120
      IF( IDATA.EQ.-9999 ) GO TO 120
      GO TO 901                    ;DATA EXISTS
C
C.... WE'VE FOUND WHERE RMIN IS
C
  120 JREV = 1                     ;SET REVISED DATA FLAG
  130 IPT = I + 2
C
C.... WRITE THE MORE RECENT DATA (IF ANY) TO A TEMPORARY FILE
C     FOR LATER STORAGE.
C
  140 IFL = -1
      IF( IPT.EQ.MPT+2 ) GO TO 160
      IFL = (MPT+2-IPT)/2
C
      OPEN JCHN,SHEFWORK                          ;TEMP WORK FILE
      DO 150 I=IPT,MPT+1
         JULH = IAW(I)
         WRITE(JCHN) JULH
  150 CONTINUE
      REWIND JCHN
C
C.... NOW RESET DATA POINTER
C
      IF( JREV.EQ.1 ) IPT = IPT - 2
      IAW(6) = IPT
      CALL AEWRB(ICHDB,NBLK,IWB,2,IER)
C
C.... CLOSE THE CHANNEL TO THE DATABASE
C
  160 CALL ACLOSE(ICHDB)
C
C.... FILE THE NEW ENTRY BY PASSING INFO TO THE FILING TASK
C
      IDATA = (VALUE - RADD) * RDIV + 0.5
      IF( VALUE.LT.-9998.9 ) IDATA = -9999
      CALL BUFIN(ISTN,NHOUR,NMIN,NDAY,NMON,IDATA,0)
C
C.... FILE THE DATA ON THE WORK FILE IF NECESSARY
C
      IF( IFL.EQ.-1 ) GO TO 900
      CALL MYID(IDNUM)
      IFL = IFL * 2
      CALL BUFIN(-1,JCHN,ISTN,IFL,IDNUM,0,0)
      WAIT IDNUM
C
C.... NOW CLOSE THE WORK FILE
C
  170 CALL CLOSE(JCHN,IER)
C
C.... THAT'S IT
C
  900 RETURN
C
C.... ERRORS
C
  901 CALL SHMSG(3)                    ;DATA ALREADY EXISTS
```

```
      RETURN 2
C
      END
```

```
      SUBROUTINE SHMSG(JER)
C
C-----------------------------------------------------------------------C
C   VERSION 1.0 SEP  1982      GEOFFREY M BONNIN  MBRFC                 C
C-----------------------------------------------------------------------C
      INTEGER ILINE(82)
      COMMON /LUNS/    ICHN,JCHN
      DOUBLE PRECISION VALUE
      COMMON /POSTERR/ NERROR
      COMMON /POSTDATA/ IDSTN(8),NYEAR,NMON,NDAY,NHOUR,NMIN,
     1                        KYEAR,KMON,KDAY,KHOUR,KMIN,
     2                  KODP,KODE,IDUR,KODT,KODS,KODEX,CODP,
     3                  VALUE,LWAL,IREV,MSOURCE(8),IDOTE
C
C.... FILE NAMES AND ATTRIBUTES
C
      DIMENSION AFSDEC(3)
      AFSDEC(1) = 'AFSD'
      AFSDEC(2) = 'EC.E'
      AFSDEC(3) = 'R'
      ATTR1 = 'SA'
C
C.... UPDATE ERROR COUNT
C
      NERROR = NERROR + 1
C
C.... PUT ERROR MESSAGES ON FILE
C
      GO TO (10,20,30,40),JER
      RETURN
C
   10 FILE = 'ER1'
      GO TO 500
C
   20 FILE = 'ER2'
      GO TO 500
C
   30 FILE = 'ER3'
      GO TO 500
C
   40 FILE = 'ER4'
C
  500 OPEN JCHN,FILE,ERR=600
      CALL RLINE(JCHN,ILINE,NBYTE,IER)
      CALL CLOSE(JCHN,IER)
C
      OPEN JCHN,AFSDEC,ATT=ATTR1
      WRITE(JCHN,510) IDSTN,NYEAR,NMON,NDAY,NHOUR,NMIN,KODP,KODE,VALUE
  510 FORMAT(/3R1,2X,4(I2,':'),I2,2X,2R1,G15.6)
      CALL WLINE(JCHN,ILINE,NBYTE,IER)
      CALL CLOSE(JCHN,IER)
C
  600 RETURN
      END
```

```
C      PARSHEF .. DRIVER ROUTINE FOR SHEF PARSING
C
C-----------------------------------------------------------------C
C    VERSION 1.0 APRIL 1982     GEOFFREY M BONNIN  MBRFC          C
C-----------------------------------------------------------------C
C
C       PRODCT(1-6) .. AFOS ID
C       PRODCT(7-10).. FILE NAME
C       PRODCT(11)  .. READY FLAG
C
       INTEGER PRODCT(11)
       INTEGER FNAME(4)
       EQUIVALENCE (FNAME(1),PRODCT(7))
       COMMON /LUNS/   LCHN,JCHN,KCHN,MCHN,MREC,ICHER
       COMMON /ERROR/ NERROR
C
C.... SET UP INTER GROUND COMMS AREA
C
       CALL ICMN(PRODCT,11,IER)
       IGCM = -2
       PRODCT(11) = 0                          ;SET READY FLAG
       CALL WRCMN(IGCM,0,1,IER)
C
C.... SET UP CHANNELS FOR THE INPUT MESSAGE FILE AND
C       THE OUTPUT DATA FILE.
C
       LCHN = 26
       JCHN = 27
       KCHN = 54
       MCHN = 55
       ICHER = 56
C
C.... WAIT FOR PRODUCT ID AND FILE NAME TO BE PASSED FROM DATACOL
C
     5 IF( PRODCT(11).EQ.1 ) GO TO 6
       CALL FDELY(1)
       GO TO 5
C
C.... WRITE MESSAGE TO CONSOLE
C
     6 WRITE(10,7) PRODCT(1)
     7 FORMAT('*** PARSHEF DECODING ',S9,' ***')
C
C.... OPEN THE FILES
C
       OPEN LCHN,FNAME,ATT='BL'
       CALL DFILW('SHEFOUT',IER)
       OPEN JCHN,'SHEFOUT'
       OPEN KCHN,'SHEFPARM',LEN=8
       OPEN MCHN,'DOTBTEMP'
       CALL DFILW('AFSDEC.ER',IER)
       OPEN ICHER,'AFSDEC.ER'
       WRITE(ICHER,10) PRODCT(1)
    10 FORMAT(/35('*')/'SHEF DECODE ERROR LOG FOR ',S9/35("*") )
C
C.... CALL THE PARSING DRIVER ROUTINE
C
       CALL SHDRIVE
C
C.... CLOSE THE FILES
C
       CLOSE LCHN
       CLOSE JCHN
```

```
      CLOSE KCHN
      CLOSE MCHN
      CLOSE ICHER
C
C.... THATS ALL
C
      IGCM = NERROR
      CALL WRCMN(IGCM,0,1,IER)
      CALL EXIT
      END
```

```
      SUBROUTINE SHERR(IER)
C
C-----------------------------------------------------------------------C
C   VERSION 1.0 APRIL 1982    GEOFFREY M BONNIN  MBRFC                  C
C-----------------------------------------------------------------------C
C
      PARAMETER ICH=5
      DIMENSION IFILE(3),LINE(80)
      COMMON /CODES/ ICHA,ICHB,ICHC,ICHD,ICHE,ICHF,ICHG,ICHH,ICHI,
     1               ICHJ,ICHK,ICHL,ICHM,ICHN,ICHO,ICHP,ICHQ,ICHR,
     2               ICHS,ICHT,ICHU,ICHV,ICHW,ICHX,ICHY,ICHZ,ICHO,
     3               ICH1,ICH2,ICH3,ICH4,ICH5,ICH6,ICH7,ICH8,ICH9,
     4               IBLNK,ISLASH,ICOLON,IPLUS,IMINUS,IDOT,IARROW,
     5               ICOMMA
      COMMON /BUFFER/ IBUF(80),IP,NBLNK
      COMMON /ERROR/ NERROR
      COMMON /LUNS/    LCHN,JCHN,KCHN,MCHN,MREC,ICHER
      DATA IFILE/'SF',2*'<000><000>'/
C
      IF( (IP.LE.0).OR.(IP.GE.81) ) GO TO 30
      DO 10 I=1,IP
         LINE(I) = IBLNK
   10 CONTINUE
      LINE(IP) = IARROW
      WRITE(ICHER,20) (LINE(I),I=1,IP)
   20 FORMAT(80R1)
C
   30 NERROR = NERROR + 1
C
C.... READ THE ERROR MESSAGE
C
      CALL FMT2D(IER,IFILE(2))
      OPEN ICH,IFILE,ERR=900
      CALL RLINE(ICH,LINE,NBYTE,JER)
      CALL CLOSE(ICH,JER)
C
C.... WRITE IT OUT
C
      CALL WLINE(ICHER,LINE,NBYTE,JER)
C
  900 RETURN
      END
```

(continued from front inside cover)

NWS CR 48 Manual of Great Lakes Ice Forecasting. C. Robert Snider December 1971 (COM-72-10143)

NWS CR 49 A Preliminary Transport Wind and Mixing Height Climatology, St. Louis, Missouri. Donald E. Wuerch, Albert J. Courtois, Carl Ewald, Gary Ernst. June 1972 (COM-72-10859)

NWS CR 50 An Objective Forecast Technique for Colorado Downslope Winds Wayne E. Sangster. December 1972 (COM-73-10280)

NWS CR 51 Effect on Temperature and Precipitation of Observation Site Change at Columbia, Missouri. Warren M. Wisner. March 1973 (COM-73-10734)

NWS CR 52 Cold Air Funnels. Jack R. Cooley and Marshall E. Soderberg September 1973. (COM-73-11753)

NWS CR 53 The Frequency of Potentially Unfavorable Temperature Conditions in St. Louis, Missouri. Warren M. Wisner. October 1973

NWS CR 54 Objective Probabilities of Severe Thunderstorms Using Predictors from FOUS and Observed Surface Data. Clarence A. David. May 1974 (COM-74-11258)

NWS CR 55 Detecting and Predicting Severe Thunderstorms Using Radar and Sferics. John V. Graff and Duane C. O'Malley. June 1974 (COM-74-11335)

NWS CR 56 The Prediction of Daily Drying Rates. Jerry D. Hill. Nov. 1974. (COM-74-11806)

NWS CR 57 Summer Radar Echo Distribution Around Limon, Colorado. Thomas D. Karr and Ronald L. Wooten. Nov. 1974. (COM-75-10076)

NWS CR 58 Guidelines for Flash Flood and Small Tributary Flood Prediction. Lawrence A. Hughes and Lawrence L. Longsdorf. October 1975
NWS CR 58rev (PR247569/AS) NWS CR 58(revised) March 1978 (PB281461/AS)

NWS CR 59 Hourly Cumulative Totals of Rainfall - Black Hills Flash Flood June 9-10, 1972. Don K. Halligan and Lawrence L. Longsdorf (PB256087)

NWS CR 60 Meteorological Effects on the Drift of Chemical Sprays. J. D. Hill. July 1976. (PB259593)

NWS CR 61 An Updated Objective Forecast Technique for Colorado Downslope Winds. Wayne E. Sangster. March 1977. (PB266966)

NWS CR 62 Design Weather Conditions for Prescribed Burning. Ronald E. Haug. April 1977. (PB268034)

NWS CR 63 A Program of Chart Analysis (with some diagnostic and forecast implications). Lawrence A. Hughes. March 1978 (PB279866)

NWS CR 64 Warm Season Nocturnal Quantitative Precipitation Forecasting for Eastern Kansas Using the Surface Geostrophic Wind Chart. Wayne E. Sangster. April 1979 (PB295982)

NWS CR 65 The Utilization of Long Term Temperature Data in the Description of Forecast Temperatures. Nov. 1981. Arno Perlow (PB8263064)

NWS CR 66 The Effect of Diurnal Heating on the Movement of Cold Fronts Through Eastern Colorado. August 1982. James L. Wiesmueller (PB83118463)

NWS CR 67 An Explanation of the Standard Hydrologic Exchange Format (SHEF) and its Implementation in the Central Region. April 1983 Geoffrey M. Bonnin.