U.S. DEPARTMENT OF COMMERCE
NATIONAL OCEANIC AND ATMOSPHERIC ADMINISTRATION
NATIONAL WEATHER SERVICE
NATIONAL METEOROLOGICAL CENTER

OFFICE NOTE 319

CIRCULAR FILES

FORMAT DOCUMENTATION AND USERS GUIDE

RAY CRAYTON
AUTOMATION DIVISION

APRIL 1986

This is an unreviewed manuscript, primarily intended for the informal exchange of information among NMC staff members.

CIRCULAR FILES

FORMAT DOCUMENTATION AND USERS GUIDE

CONTENTS

1. INTRODUCTION

Selected data, received by NMC's IBM 4341 computer from communications lines, and needed for NMC and other units' operations, are reformatted and transferred to files on a disk volume that is accessible by both the IBM 4341 and NAS 9000 computer systems. These files are commonly referred to as the circular files. Pointers (record numbers and byte numbers) to data in the circular files are kept in a file of control records.

This Office Note documents the formats of the control records and the circular files. It shows examples using the W3FK14 subroutine to retrieve data from the circular files. Finally, it provides the documentation for the W3FK14 routine.

ACCESS TO THE CONTROL RECORDS FILE AND THE CIRCULAR FILES IS RESTRICTED TO OPERATIONALLY SCHEDULED PROGRAMS.

2. DEFINITIONS

2.1 'Record' is synonymous with 'physical record' or 'block'.

2.2 'Logical record' or 'data prefix/bulletin' is used when referring to a data prefix and its associated data bulletin or data field.

2.3 'File' is synonymous with 'data set'.

3. CONTROL RECORDS FILE

Data set 'NMC.PROD.RGTR.RAWCTLS' contains the bookkeeping for the programs that write the circular files. The records in this file are the so-called control records. Record number 1 contains a log of the circular data sets writing activities. There is a control record for each circular data file. The record numbers are listed in Table A, column C. Each record contains pointers (record numbers and byte numbers) to data in the relevant circular file and also contains the number of records in the circular file, the byte size of each record, and the date/time data were transferred to the circular file.

3.1 FILE ATTRIBUTES

RECORD FORMAT	- Fixed Length
RECORD SIZE	- 2176 Bytes
FILE ORGANIZATION	- Physical Sequential

3.2 RECORD 1 FORMAT (the logging record)

BYTE NUMBER(S)	DATA REPRESENTATION	CONTENTS
1-4	EBCDIC	Record identification 'LOGƀ'.
5-8	EBCDIC	Blanks.
9-2176	Binary	542 4-byte entries (see below).

4-BYTE ENTRY FORMAT (for bytes 9-2176)

1	Binary	Control record number to which entry applies. See Table, A column B.
2	Binary	Day / Date and time that / the data were
3	Binary	Hour / transferred to the circular file.
4	Binary	Minute /

(CONTINUED ON NEXT PAGE)

3.3 RECORDS 2,3,...n FORMAT (the control records)

BYTE NUMBER(S)	DATA REPRESENTATION	CONTENTS
1-6	EBCDIC	Control record identification See Table A, column C.
7-8	EBCDIC	Blanks.
9-14	Binary	Zero fill.
15-16	Binary	Number of records in the circular file.
17-18	Binary	Number of bytes in each record of the circular file.
19-20	Binary	Record number of the next write in the circular file.
21-22	Binary	Byte number of the next write at which data will start in the record specified by bytes 19-20.
23	Binary	Year / Date and time that the
24	Binary	Month / data were transferred
25	Binary	Day / to the circular file.
26	Binary	Hour /
27	Binary	Minute/
28	Binary	Zero fill.
29-2170	Binary	153 14-byte entries formatted the same as bytes 15-28.
2171-2176	Binary	Zero fill.

A new entry is made on each data transfer providing the most recent entry in bytes 15-28 and the oldest entry in bytes 2157-2170.

4. CIRCULAR FILES

Each record in a circular file consists of continuous variable length logical records (data prefixes/bulletins) except for the last 8 bytes. These 8 bytes contain the pointer to where the first complete data prefix starts in the record and the date/time it was transferred to the circular file.

A 20-byte end-of-data entry is appended to the last logical record of a data transfer. This entry is overwritten by the first logical record of the next data transfer. Logical records span records when necessary.

Pointers (record numbers and byte numbers) relevant to processing the circular files are kept in the control records file (data set 'NMC.PROD.RGTR.RAWCTLS').

4.1 FILE ATTRIBUTES

RECORD FORMAT	- Fixed Length
RECORD SIZE	- See Table A, column E
FILE ORGANIZATION	- Physical Sequential

4.2 RECORD FORMAT

BYTE NUMBER(S)	DATA REPRESENTATION	CONTENT
1 ... n-8	As specified by byte 9 of data prefixes.	A continuous string of data organized into data prefixes/bulletins.
n-7 & n-6	Binary	The byte number beginning the first complete data prefix in the record. It is set to zero when there is no complete prefix in the record.
n-5	Binary	Year / The same entry as in
n-4	Binary	Month / bytes 13-17 of the
n-3	Binary	Day / first data prefix in
n-2	Binary	Hour / the record. It is zero
n-1	Binary	Minute/ when bytes n-7 & n-6 above are zero.
n	Binary	Zero fill.

4.3 LOGICAL RECORD FORMAT

4.3.1 DATA PREFIX

BYTE NUMBER(S)	DATA REPRESENTATION	CONTENTS
1	Binary	Data type(s). See table B.
2	Binary	Communications line number that data were received on.
3-4	Binary	Number of bytes in the data prefix (P).
5-8	Binary	Number of bytes in the logical record (L).
9	Binary	Data representation. See Table C.
10	Binary	Day / Date and time that
11	Binary	Hour / bulletin was received
12	Binary	Minute/ at NMC.
13	Binary	Year / Date and time that the
14	Binary	Month / data were transferred to
15	Binary	Day / the circular file.
16	Binary	Hour /
17	Binary	Minute/
18-20	Binary	Zero fill

NOTE: THE PREFIX WILL BE EXPANDED AS NEEDED.

4.3.2 DATA BULLETIN OR DATA FIELD

BYTE NUMBER(S)	DATA REPRESENTATION	CONTENTS
P+1...L-2	As specified by byte 9 of the prefix.	One data bulletin of type(s) specified by byte 1 of the prefix.
L-1...L	EBCDIC	The last two characters of the data bulletin are '**' which is hexadecimal 5C5C.

4.4 END-OF-DATA ENTRY FORMAT

BYTE	DATA	
NUMBER(S)	REPRESENTATION	CONTENTS
1-20	EBCDIC	'ENDOFDATAb̸b̸ENDOFDATA' is appended to the last logical record of a data transfer. This entry is overwritten by the first logical record of the next data transfer.

5. REFERENCE TABLES

5.1 TABLE A CIRCULAR FILE SUMMARY

A	B	C	D	E	F
NMC.PROD.RGTR.HYGOES	3	HYGCTL	405	11616	31
NMC.PROD.RGTR.MTYBUL	4	MTYCTL	102	11616	33
NMC.PROD.RGTR.SHPRAW	5	SHPCTL	135	11476	10
NMC.PROD.RGTR.GEOSAT	6	GEOCTL	180	11476	37
NMC.PROD.RGTR.MISC01	7	MS1CTL	180	11476	06,30,38,39,45
NMC.PROD.BUOY.RPORTS	-	-	15	11476	40,42

NOTE: Only six circular files in the Office Note 319 format are active as of April 10, 1986. Files will be added to this summary when they are activated.

KEY FOR COLUMN HEADINGS:

A --- File Name.
B --- Relevant Record Number in the Control Records File.
C --- Record Identification in the Control Records File.
D --- Number of Records in the File.
E --- Record Size in Bytes.
F --- Data Type(s). See Table B.

5.2 TABLE B DATA TYPE CODE

TYPE CODE	DATA SET NAME	BULLETIN HEADING(S)
06	NMC.PROD.RGTR.MISC01	ADMN67
10	NMC.PROD.RGTR.SHPRAW	SHIP, SHPS, SHVa, SIVa, SIWa, SMVa, SMWa, SNVa, SNWa, SSVX, SSVD, SSNT, SSWX, SXUS, YEXX, YHXX, YOXX
30	NMC.PROD.RGTR.MISC01	SXUS70
31	NMC.PROD.RGTR.HYGOES	PLHY11, RRaa, SRUS75
33	NMC.PROD.RGTR.MTYBUL	NQaa
37	NMC.PROD.RGTR.GEOSAT	TSXX
38	NMC.PROD.RGTR.MISC01	SRaa
39	NMC.PROD.RGTR.MISC01	UXUS
40	NMC.PROD.BUOY.RPORTS	BUOY AND C-MAN REPORTS
42	NMC.PROD.BUOY.RPORTS	BATHYTHERMOGRAPH REPORTS
45	NMC.PROD.RGTR.MISC01	FXXT10, FXXN10, FXXS10
197	NONE	RESERVED

NOTE:

1. Only six circular files in Office Note 319 format are active as of April 10, 1986. More codes will be added when files are activated.

2. The 'a' or 'aa' means that the bulletin heading contains a one or two-letter area designator.

3. This data-type code is the code referred to by byte 1 of the data prefix in section 4.3.1.

(THIS PAGE INTENTIONALLY LEFT BLANK)

5.3 TABLE C DATA REPRESENTATION CODE

CODE	DATA REPRESENTATION
00	UNSPECIFIED
01	EBCDIC
02	ASCII
03	BINARY
04	EBCDIC HEADING, ASCII CONTENT
05	ASCII HEADING, BINARY CONTENT

NOTE: The data representation code is the code referred to by byte 9 of the data prefix in section 4.3.1.

6. HOW TO RETRIEVE DATA FROM A CIRCULAR FILE

Data can be retrieved from a circular file by using subroutine W3FK14. The documentation for this routine is found in APPENDIX A and the program load module is in data set 'NWS.NMC.MVSW3LIB.LOAD'. After opening a circular file with a call to W3FK14 one needs to make calls to entry point W3FK15 to retrieve data bulletins, W3FK16 for error recovery, and W3FK17 to close the file. The following are examples on using the W3FK14 routine and its entry points, and the control records file to retrieve data from a circular file.

6.1 EXAMPLE 1 -

a. Single call to W3FK14.
b. Retrieves two types of data from one file.
c. Attempts to recover from a read error.

```
C
      LOGICAL*1 IDTTME(5), IPREF(100), IBULL(5000)
      INTEGER*2 IFILL(3) , NUMREC    , LENREC
      INTEGER*2 IEREC    , IEBYTE
      INTEGER*2 ISREC    , ISBYTE
      INTEGER*4 NUMTYP/2/, ITYPE(2)/02,11/
      INTEGER*4 LENPFX/100/, LENBUL/5000/
      REAL*8    DDNAME/'FT08F001'/
      REAL*8    IDENR    , IDENRW/'MS2CTL  '/,
      REAL*8    IDENU
C*
C* DEFINE THE CONTROL RECORDS FILE FOR DIRECT ACCESS
      DEFINE FILE 1(100,2176,L,KVAR)
C*
C* READ THE REVELANT CIRCULAR FILE CONTROL RECORD TO GET POINTERS
C* TO WHERE THE MOST RECENT DATA ENDS IN THE CIRCULAR FILE.
C* IN THIS EXAMPLE THE POINTERS ARE IN CONTROL RECORD 2.
C*
      READ(1'2,ERR=800)IDENR,IFILL,NUMREC,LENREC,IEREC,IEBYTE,
     1                 IDTTME
      IF(IDENR.NE.IDENRW)GO TO 875
C*
C* READ USER CONTROLS TO GET POINTERS TO WHERE THE PROCESSING
C* SHOULD START IN THE CIRCULAR FILE.  USERS SHOULD KEEP DATA
C* POINTERS ACCORDING TO THEIR OWN REQUIREMENTS.  THIS IS ONLY
C* ONE EXAMPLE OF USING AND SAVING USER DATA CONTROLS.
C*
      READ(2,100,END=850,ERR=800)IDENU,ISREC,ISBYTE
  100 FORMAT(A8,2I5,5I2)
      IF(IDENU.NE.IDENRW)GO TO 875
      REWIND 2
```

(CONTINUED ON NEXT PAGE)

```
C*
C* SAVE THE POINTERS FROM THE CIRCULAR FILE CONTROL RECORD
C* IN THE USER CONTROL FILE FOR THE NEXT RUN.
C*
        WRITE(2,100)IDENU,IEREC,IEBYTE,IDTTME
C*
C* OPEN THE CIRCULAR FILE.
C*
        CALL W3FK14(DDNAME,ITYPE,NUMTYP,LENPFX,LENBUL,NUMREC,
     1            LENREC,IEREC,IEBYTE,ISREC,ISBYTE,ISTAT)
        IF(ISTAT.EQ.0)GO TO 200
        IF(ISTAT.EQ.1)GO TO 650
C    SOME KIND OF ERROR.
        GO TO 800
C*
C*
C* RETRIEVE DATA BULLETINS.
C*
  200   CALL W3FK15(IPREF,LP,IBULL,LB,IRS,ISTAT)
        IF(ISTAT.EQ.0)GO TO 400
        IF(ISTAT.EQ.1)GO TO 600
C    NOT END OF DATA.
        IF(ISTAT.EQ.2)GO TO 450
C    WAS NOT A BULLETIN FRAGMENT.  MUST BE SOME KIND OF ERROR.
        IF(ISTAT.EQ.21)GO TO 500    (You may want to terminate
C                                    on a read error.)
C    NOT A READ ERROR.  GO TERMINATE.
        GO TO 800
C*
C* GOT A BULLETIN (OR BULLETIN FRAGMENT WHEN ISTAT=2) PROCESS IT.
C*
 400    .
        .
        .     (Your data bulletin processing code.)
 450    .
        .
        GO TO 200
C*
C* COMES HERE IF THERE IS A READ ERROR ON THE CIRCULAR FILE.
C*
 500    .
        .     (Process read error.)
        .     (Call W3FK16 if you want to go to next record
        .     (and continue processing.)
        CALL W3FK16(1,ISTAT)
        IF(ISTAT.EQ.0)GO TO 200
        IF(ISTAT.EQ.1)GO TO 600
C    ERROR HERE, GO TERMINATE.
        GO TO 800  (Or check for another read error and attempt
C                   to continue processing by calling W3FK16
C                   again.  Your choice.)
```

```
C*
C* NORMAL END.
C*
 600    .
        .            (Normal end of run code.)
 650    .
        .
        STOP
C*
C* ABNORMAL END.
C*
 800    .
        .            (Abnormal end of run code.)
 850    .
        .
 875    .
        STOP
        END
```

6.2 EXAMPLE 2 - a. Multiple calls to W3FK14.

b. Retrieve all data types.

```
C
         LOGICAL*1 IDTTME(5)     ,IPREF(100)      ,IBULL(5000)
         INTEGER*2 IFILL(3)      ,IRAWC(4)
         INTEGER*2 ISREC         ,ISBYTE
         INTEGER*4 INUSER(2)/3,4/
         INTEGER*4 INREC (2)/2,3/
         INTEGER*4 NUMTYP/1/     ,ITYPE/0/
         INTEGER*4 LENPFX/100/   ,LENBUL(5000)
         REAL*8    IDENR         ,IDENRW(2)/'SFCCTL  ','ACFCTL  '/
         REAL*8    IDENU
         REAL*8    DDNAME(2)/'FT08F001','FT09F001'/
C*
C* DEFINE THE CIRCULAR CONTROL RECORDS FILE AND THE USER CONTROLS
C* FOR DIRECT ACCESS.
C*
        DEFINE FILE 1(100,2176,L,KVAR)
        DEFINE FILE 2(10 ,  28,L,LVAR)
        IX   = 0
```

(CONTINUED ON NEXT PAGE)

```
C*
C* READ THE CIRCULAR FILE CONTROL RECORD TO GET POINTERS
C* TO WHERE THE LATEST DATA ENDS IN THE CIRCULAR FILE.  IN
C* THIS EXAMPLE THE POINTERS ARE IN RECORDS 2 AND 3.
C*
 50     IX    = IX+1
        INR   = INREC(IX)
        READ(1'INR,ERR=800)IDENR,IFILL,IRAWC,IDTTME
        IF(IDENR.NE.IDENRW(IX))GO TO 850
C*
C* READ USER CONTROLS TO GET POINTERS TO WHERE THE PROCESSING
C* SHOULD START IN THE CIRCULAR FILE.  USER CONTROLS ARE IN
C* RECORDS 3 AND 4.  USERS SHOULD KEEP DATA POINTERS ACCORDING
C* TO THEIR OWN REQUIREMENTS.  THIS EXAMPLE DOES NOT IMPLY
C* THAT IT NEED BE DONE EXACTLY THIS WAY.
C*
        IOUC = INUSER(IX)
        READ(2'IOUC,100,ERR=800)IDENU,ISREC,ISBYTE
 100    FORMAT(A8,2I5,5I2)
        IF(IDENU.NE.IDENRW(IX))GO TO 850
C*
C* SAVE CIRCULAR FILE POINTERS IN THE USER CONTROL FILE
C* FOR THE NEXT RUN.
C*
        WRITE(2'IOUC,100)IDENU,IRAWC(3),IRAWC(4),IDTTME
C*
C* OPEN THE CIRCULAR FILE.
C*
        CALL W3FK14(DDNAME(IX),ITYPE,NUMTYP,LENPFX,LENBUL,
     1  IRAWC(1),IRAWC(2),IRAWC(3),IRAWC(4),ISREC,ISBYTE,ISTAT)
C
        IF(ISTAT.EQ.0)GO TO 200
        IF(ISTAT.EQ.1)GO TO 600
C    SOME KIND OF ERROR.  TERMINATE PROCESSING.
        GO TO 800
C*
C*
C* RETRIEVE DATA BULLETINS.
C*
  200   CALL W3FK15(IPREF,LP,IBULL,LB,IRS,ISTAT)
        IF(ISTAT.EQ.0)GO TO 400
        IF(ISTAT.EQ.1)GO TO 600
C    NOT END OF DATA.
        IF(ISTAT.EQ.2)GO TO 450
C    NOT A BULLETIN FRAGMENT.  SOME KIND OF ERROR.
        GO TO 800 (You may want to try an error recovery using
C                  W3FK16.  Your choice.  SEE EXAMPLE 1.)
C
C* GOT A BULLETIN (OR BULLETIN FRAGMENT WHEN ISTAT=2) PROCESS IT.
C*
  400   .
        .
  450   .                 (Your data bulletin processing code.)
        .
        GO TO 200
```

```
C*
C* NORMAL END.
C*
 600     IF(IX.EQ.2)GO TO 675
         CALL W3FK17(ISTAT)    (If additional calls are to be made
C                               to W3FK14, the circular file just
         GO TO 50               processed must be closed.)
 675     .
         .                     (Normal end of run code.)
         .
         STOP
C*
C* ABNORMAL END.
C*
 800     .
         .                     (Abnormal end of run code.)
 850     .
         .
         STOP
         END
```

```
C$$$                    THE CIRCULAR FILE READER
C
C SUBPROGRAM:  W3FK14            OPENS A CIRCULAR FILE
C   AUTHOR: RAY CRAYTON          ORG: W/NMC441      DATE: 86-04-10
C
C
C ABSTRACT: CHECKS INPUT ARGUMENTS FOR VALIDITY, OPENS AN OFFICE
C   NOTE 319 CIRCULAR FILE AND READS THE BEGINNING RECORD.  THIS
C   ROUTINE HAS ENTRY POINTS FOR RETRIEVING DATA (W3FK15), ERROR
C   RECOVERY (W3FK16), AND FILE CLOSING (W3FK17).
C
C
C
C USAGE:   CALL W3FK14(DDNAME,ITYPE,NUMTYP,LENPFX,LENBUL,NUMREC,
C                    LENREC,IEREC,IEBYTE,ISREC,ISBYTE,ISTAT)
C
C   INPUT VARIABLES:
C     NAMES  INTERFACE DESCRIPTION OF VARIABLES AND TYPES
C     ------ --------- -----------------------------------------------
C
C    DDNAME  ARG LIST  8-CHARACTER EBCDIC DDNAME LEFT-ALIGNED
C                      WITH BLANK FILL.
C
C    ITYPE   ARG LIST  INTEGER*4 ARRAY CONTAINING THE TYPES OF DATA TO
C                      BE RETRIEVED OR AN INTEGER*4 WITH 'ITYPE' = 0
C                      TO CAUSE ALL DATA TYPES TO BE RETRIEVED.
C                      THE 'ITYPE' VARIABLE REFERS TO THE FIRST BYTE
C                      OF THE DATA PREFIX.
C
C    NUMTYP  ARG LIST  INTEGER*4 NUMBER OF DATA TYPES TO BE RETRIEVED.
C                      SET 'NUMTYP' = 1 FOR ALL TYPES WHEN 'ITYPE' = 0.
C
C    LENPFX  ARG LIST  INTEGER*4 BYTE LENGTH OF THE USER'S DATA PREFIX
C                      AREA.  MUST BE AT LEAST 20 BYTES.
C
C    LENBUL  ARG LIST  INTEGER*4 BYTE LENGTH OF THE USER'S DATA BULLETIN
C                      AREA.  MUST BE AT LEAST 80 BYTES.
C
C    NUMREC  ARG LIST  INTEGER*2 NUMBER OF RECORDS IN THE CIRCULAR
C                      FILE (THE ENTRY IN BYTES 15-16 OF THE
C                      CIRCULAR FILE'S CONTROL RECORD).
C
C    LENREC  ARG LIST  INTEGER*2 BYTE LENGTH OF THE DATA RECORDS IN
C                      THE CIRCULAR FILE.  IT CAN NOT EXCEED 11616
C                      BYTES (THE ENTRY IN BYTES 17-18 OF THE CIRCULAR
C                      FILE'S CONTROL RECORD).
C
C    IEREC   ARG LIST  INTEGER*2 RECORD NUMBER TO BE USED AS THE
C                      'END-OF-DATA' RECORD (USUALLY THE ENTRY IN
C                      BYTES 19-20 OF THE CIRCULAR FILE'S CONTROL
C                      RECORD).
```

```
C    IEBYTE  ARG LIST  INTEGER*2 BYTE NUMBER IN RECORD 'IEREC' TO BE
C                      USED AS THE END-OF-DATA BYTE (USUALLY THE
C                      ENTRY IN BYTES 21-22 OF THE CIRCULAR FILE'S
C                      CONTROL RECORD).
C
C    ISREC   ARG LIST  INTEGER*2 RECORD NUMBER TO BE USED AS THE
C                      'START-OF-DATA' RECORD.
C
C    ISBYTE  ARG LIST  INTEGER*2 BYTE NUMBER IN RECORD 'ISREC' TO BE
C                      USED AS THE 'START-OF-DATA' BYTE.  IF YOU WANT
C                      TO START AT THE FIRST DATA PREFIX/BULLETIN IN
C                      RECORD 'ISREC', SET ISBYTE = 0 (SEE REMARK 2).
C
C   OUTPUT VARIABLES:
C     NAMES  INTERFACE DESCRIPTION OF VARIABLES AND TYPES
C     ------ --------- -----------------------------------------------
C     ISTAT  ARG LIST  INTEGER*4 RETURN CODE:
C
C                      =  0  SUCCESSFUL FILE OPEN.
C
C                      =  1  NO NEW DATA TO READ. FILE NOT OPENED.
C
C                      =  5  FORMAT ERROR.  THE POINTER TO THE FIRST
C                            DATA PREFIX/BULLETIN IN THE RECORD IS OUT
C                            OF RANGE (SEE REMARK 2).  FILE OPEN.
C
C                      =  6  USER'S SPECIFIED DATA BULLETIN AREA SIZE IS
C                            LESS THAN 80 BYTES.  FILE NOT OPENED.
C
C                      =  7  USER'S PREFIX AREA IS LESS THAN 20 BYTES.
C                            FILE NOT OPENED.
C
C                      =  9  RECORD BYTE SIZE IS LESS THAN 80 OR GREATER
C                            THAN 11616 BYTES.  FILE NOT OPENED.  CHECK
C                            INPUT ARG. 'LENREC', IT MUST BE TYPE
C                            INTEGER*2.
C
C                      = 10  A CIRCULAR FILE IS ALREADY OPEN.  IT MUST
C                            BE CLOSED USING W3FK17 BEFORE OPENING
C                            ANOTHER (SEE REMARK 1).
C
C                      = 11  THIS ROUTINE USES THE W3AK43 I/O PACKAGE.
C                            THE MAXIMUM NUMBER OF FILES THAT CAN BE
C                            OPENED WITH W3AK43 IS THREE.  A FILE MUST
C                            BE CLOSED USING W3AK46 BEFORE USING THE
C                            W3FK14 ROUTINE.
C
C                      = 12  FILE NOT OPENED.  PROBABLY WRONG DDNAME.
C
C                      = 21  INPUT READ ERROR.  FILE OPEN.
C
C                      = 22  WRONG LENGTH RECORD SPECIFIED.  FILE OPEN.
C                            CHECK INPUT ARG. 'LENREC', IT MUST BE TYPE
C                            INTEGER*2.
```

```
C                         = 23  AT LEAST ONE OF THE FOLLOWING INPUT ARGS.
C                               IS OUT OF RANGE: 'NUMREC', 'ISREC',
C                               'ISBYTE', 'IEREC', 'IEBYTE'.  THEY MUST BE
C                               TYPE INTEGER*2.  FILE OPEN.
C
C                         = 24  NO FILE OPENED FOR DDNAME SPECIFIED.
C
C   SUBPROGRAMS CALLED:
C     NAMES                                                       LIBRARY
C     --------------------------------------------------------   --------
C     CIRFIL                                                      UNIQUE
C     W3AK43                                                      W3LIB
C
C   REMARKS:
C
C     1.  ONLY ONE FILE CAN BE OPEN AT ANY ONE TIME.  ADDITIONAL
C     FILES CAN BE OPENED BUT PREVIOUSLY OPENED FILE MUST BE CLOSED
C     USING ENTRY POINT W3FK17.
C
C     2.  'ISBYTE = 0' FORCES THE USE OF A DATA POINTER FOUND IN THE
C     LAST 8 BYTES OF EACH RECORD.  THIS POINTER GIVES THE BYTE NUMBER
C     TO THE FIRST DATA PREFIX/BULLETIN IN THE RECORD SPECIFIED BY
C     'ISREC'.  THERE ARE CONDITIONS THAT CAN OCCUR WHERE THE DATA
C     POINTER IS SET TO ZERO.  HERE ARE A FEW RULES TO FOLLOW TO STAY
C     CLEAR OF THESE PITFALLS:
C
C        A.  DO NOT USE 'ISBYTE = 0' WITH 'ISREC = IEREC'.
C        B.  DO OUT USE 'ISBYTE = 0' WITH 'ISREC = IEREC+1'.
C
C        A RETURN OF 'ISTAT=5' CAN OCCUR IF THESE RULES ARE NOT
C     FOLLOWED.
C
C     3. HERE ARE A FEW MORE CAVEATS:
C
C        A.  BEWARE OF A 'RACE TRACK' CONDITION.  DO NOT START WITH
C        A RECORD 'ISREC' THAT MIGHT BE USED ON THE NEXT DATA
C        TRANSFER TO THE CIRCULAR FILE.  THE CIRCULAR FILE WRITER
C        MAY BE TRANSFERRING DATA TO THE CIRCULAR FILE AT THE SAME
C        TIME YOUR RETRIEVE PROGRAM IS EXECUTING.  THEREFORE IT IS
C        POSSIBLE FOR THE DATA WRITER TO PASS YOUR DATA RETRIEVE
C        AREA.
C
C            YOU NEED TO BE FAMILIAR WITH THE 'WRAP AROUND' TIME OF
C        THE CIRCULAR FILES YOU ARE USING.
C
C        B.  THE LATEST AND THE OLDEST DATA USUALLY RESIDE IN RECORD
C        'IEREC' WHEN 'IEREC' IS OBTAINED FROM THE CONTROL RECORDS
C        FILE.  ATTEMPTS TO RETRIEVE DATA ACROSS THE TRANSITION FROM
C        LATEST TO OLDEST DATA MAY HAVE UNDESIRABLE RESULTS.  ALWAYS
C        RETRIEVE DATA STARTING WITH THE OLDER DATA AND STOPPING
C        WITH THE MOST RECENT DATA TO AVOID THIS.
C
C ATTRIBUTES:
C   LANGUAGE: FORTRAN H EXTENDED
C   SOURCE STATEMENTS: 288
C************************************************************************
```

```
************************************************************************
C
C   ENTRY: W3FK15               RETRIEVES CIRCULAR FILE DATA
C
C ABSTRACT: RETRIEVES ONE DATA PREFIX/BULLETIN.  DATA BULLETINS
C   TOO LARGE FOR THE AREA RESERVED WILL BE RETURNED IN PARTS.
C
C   W3FK15 IS AN ENTRY POINT IN SUBROUTINE W3FK14.
C
C USAGE:  CALL W3FK15(IPREF,LP,IBULL,LB,IRS,ISTAT)
C
C   INPUT VARIABLES:
C     NAMES  INTERFACE DESCRIPTION OF VARIABLES AND TYPES
C     ------ --------- -----------------------------------------------
C     NONE
C
C   OUTPUT VARIABLES:
C     NAMES  INTERFACE DESCRIPTION OF VARIABLES AND TYPES
C     ------ --------- -----------------------------------------------
C
C     IPREF  ARG LIST  AN ARRAY OF LENGTH 'LENPFX' (W3FK14 ARG)
C                      CONTAINING THE DATA PREFIX.
C                      SEE DOCUMENTATION ON THE DATA PREFIX IN
C                      NMC OFFICE NOTE 319 (SEC. 4.3.1).
C
C     LP     ARG LIST  INTEGER*4 ACTUAL NUMBER OF BYTES TRANSFERRED
C                      TO 'IPREF'.
C
C     IBULL  ARG LIST  AN ARRAY OF LENGTH 'LENBUL' (W3FK14 ARG)
C                      CONTAINING THE DATA BULLETIN INCLUDING THE
C                      TRAILER '**' WHICH IS HEXADECIMAL 5C5C.
C
C     LB     ARG LIST  INTEGER*4 NUMBER OF BYTES TRANSFERRED TO
C                      'IBULL' INCLUDING THE TRAILER '**'.
C
C     IRS    ARG LIST  INTEGER*4 RECORD NUMBER OF THE LAST RECORD READ.
C
```

(CONTINUED ON NEXT PAGE)

```
C      ISTAT  ARG LIST  INTEGER*4 RETURN CODE:
C
C                          =  0  SUCCESSFUL.
C
C                          =  1  END OF DATA.
C
C                          =  2  THE DATA RETURNED IN 'IBULL' IS NOT A
C                                COMPLETE BULLETIN.  THE REMAINDER WILL BE
C                                RETURNED ON THE NEXT CALL TO W3FK15.
C
C                          =  3  RECORD FORMAT OR DATA POINTER ERROR.
C                                THE EXPECTED DATA TRAILER '**' WAS NOT
C                                DETECTED.
C
C                          =  7  RECORD FORMAT OR DATA POINTER ERROR.
C                                THE DATA PREFIX IS LESS THAN 20 OR GREATER
C                                THAN 100 BYTES, OR THE PREFIX/BULLETIN IS
C                                LESS THAN 22 OR GREATER THAN 32768 BYTES.
C
C                          = 21  INPUT READ ERROR.
C
C                          = 22  WRONG LENGTH RECORD SPECIFIED.
C
C                          = 23  RECORD NUMBER SPECIFIED IS OUT OF RANGE.
C
C                          = 24  INPUT FILE WAS NOT OPENED OR THERE WAS
C                                AN ERROR RETURNED FROM THE PREVIOUS CALL
C                                TO W3FK14 OR W3FK16. CORRECT THE ERROR
C                                BEFORE PROCEEDING.
C
C    SUBPROGRAMS CALLED:
C      NAMES                                                  LIBRARY
C      -----------------------------------------------------  --------
C      CIRFIL                                                 UNIQUE
C      XMOVEX                                                 W3LIB
C
C    REMARKS: NONE
C
C
C***********************************************************************
```

```
C***********************************************************************
C
C ENTRY POINT: W3FK16        RECOVERS FROM ERROR
C
C ABSTRACT: SEARCHES FOR THE NEXT DATA PREFIX/BULLETIN OR READS
C   THE NEXT RECORD AND SETS THE DATA POINTER AT THE START OF
C   THE FIRST DATA PREFIX/BULLETIN IN THE RECORD.
C
C   IF THIS OPERATION IS SUCCESSFUL, USE W3FK15 TO CONTINUE THE
C   RETRIEVE PROCESS.
C
C
C   W3FK16 IS AN ENTRY POINT IN THE W3FK14 ROUTINE.
C
C
C USAGE:  CALL W3FK16(IW,ISTAT)
C
C   INPUT VARIABLES:
C     NAMES  INTERFACE DESCRIPTION OF VARIABLES AND TYPES
C     ------ --------- ----------------------------------------------
C
C     IW     ARG LIST  INTEGER*4 SWITCH TO DETERMINE WHICH OPTION TO
C                      SELECT.
C
C                      =  1  SETS DATA POINTER AT THE FIRST BYTE OF THE
C                            FIRST DATA PREFIX/BULLETIN IN THE NEXT
C                            RECORD.
C
C                      =  2  SETS DATA POINTER AT THE FIRST BYTE OF THE
C                            FIRST DATA PREFIX/BULLETIN FOUND (SEE
C                            REMARKS).
C
C   OUTPUT VARIABLES:
C     NAMES  INTERFACE DESCRIPTION OF VARIABLES AND TYPES
C     ------ --------- ----------------------------------------------
C
C     ISTAT  ARG LIST  INTEGER*4 RETURN CODE:
C
C                      =  0  SUCCESSFUL.
C
C                      =  1  END OF DATA.
C
C                      =  5  FORMAT ERROR.  THE POINTER TO THE
C                            FIRST DATA PREFIX/BULLETIN IN THE RECORD
C                            IS OUT OF RANGE.
C
C                      =  6  COULD NOT FIND A DATA TRAILER (**).
C                            SEE REMARKS.
C
C                      = 21  INPUT READ ERROR.
C
C                      = 22  WRONG LENGTH RECORD SPECIFIED.
C
C                      = 23  RECORD NUMBER SPECIFIED IS OUT OF RANGE.
```

```
C                      = 24  INPUT FILE WAS NOT OPENED OR THERE WAS
C                            AN ERROR RETURNED ON THE PREVIOUS CALL
C                            TO W3FK14.  CORRECT THE ERROR BEFORE
C                            PROCEEDING.
C
C
C   SUBPROGRAMS CALLED:
C     NAMES                                                   LIBRARY
C     ------------------------------------------------------- --------
C     CIRFIL                                                  UNIQUE
C
C   REMARKS:
C
C    REFERENCE TO INPUT ARGUMENT IW = 2:
C
C     1.  A SEARCH WILL BE MADE FOR THE NEXT DATA TRAILER '**'.  THE
C     DATA POINTER WILL BE SET AT THE BYTE FOLLOWING THE '**' WHICH
C     WILL BE THE BEGINNING OF A DATA PREFIX/BULLETIN WHEN '**' IS
C     UNIQUE TO THE DATA.
C
C     2.  WHEN A SEQUENCE OF '**' IS NOT UNIQUE TO THE DATA, ADDITIONAL
C     USER PROGRAM CHECKING WILL BE NEEDED TO DETERMINE IF THE DATA
C     POINTER IS POSITIONED AT THE BEGINNING OF A DATA PREFIX/BULLETIN.
C
C
C***********************************************************************
```

```
C***********************************************************************
C
C ENTRY POINT: W3FK17        CLOSES FILE OPENED BY W3FK14
C
C ABSTRACT: CLOSES FILE OPENED BY THE W3FK14 ROUTINE.
C
C     W3FK17 IS AN ENTRY POINT INTO THE W3FK14 ROUTINE.
C
C
C USAGE:  CALL W3FK17(ISTAT)
C
C   INPUT VARIABLES:
C     NAMES  INTERFACE DESCRIPTION OF VARIABLES AND TYPES
C     ------ --------- -----------------------------------------------
C     NONE
C
C   OUTPUT VARIABLES:
C     NAMES  INTERFACE DESCRIPTION OF VARIABLES AND TYPES
C     ------ --------- -----------------------------------------------
C
C     ISTAT  ARG LIST  INTEGER*4 RETURN CODE:
C
C                      =  0  SUCCESSFUL.
C
C                      = 31  FILE FOR DDNAME SPECIFIED WAS NOT OPENED.
C
C   SUBPROGRAMS CALLED:
C     NAMES                                                   LIBRARY
C     ------------------------------------------------------- --------
C     W3AK46                                                  W3LIB
C
C   REMARKS:
C
C       THE CIRCULAR FILE DOES NOT NEED TO BE CLOSED UNLESS
C     ANOTHER CALL TO W3FK14 IS MADE.
C
C
C
C$$$
C$$$
```