NOAA, NATIONAL WEATHER SERVICE, CENTRAL REGION

COMPUTER PROGRAMS AND PROBLEMS

NWS CRCP - NO. 8

A MULTIPURPOSE WEATHER ROUNDUP PROGRAM

Warren E. Sunkel
National Weather Service Forecast Office
Topeka, Kansas

May 1983

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

NOAA, National Weather Service
Central Region Computer Programs and Problems Series

NWS CRCP No. 1 A Computer Program for Determining and Displaying the Geostrophic Winds on Lakes Superior and Michigan. R. Somrek. June 1980 (PB80 225675)

NWS CRCP No. 2 Program ALEMBIC. W. Sunkel. March 1982 (PB83 114488)

NWS CRCP No. 3 A Subroutine to Find and Extract Data from an AFOS Database Product. R. Somrek. May 1982 (PB83 118828)

NWS CRCP No. 4 Programs SAVALL and STORALL. W. Sunkel. July 1982. (PB83 118448)

NWS CRCP No. 5 Program ANALYZ. T. Schwein. Sept. 1982

NWS CRCP No. 6 A Regional Weather Depiction Plot. W. Sunkel. October 1982.

NWS CRCP No. 7 The Topeka Library (TOP.LB). W. Sunkel. March 1983

NOAA, National Weather Service, Central Region
Computer Programs and Problems

NWS CRCP - No. 8

A MULTIPURPOSE WEATHER ROUNDUP PROGRAM

Warren E. Sunkel
National Weather Service Forecast Office
Topeka, Kansas

May 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

A MULTIPURPOSE WEATHER ROUNDUP PROGRAM

Warren E. Sunkel
National Weather Service Forecast Office
Topeka, Kansas

I. GENERAL INFORMATION

A. Introduction

Program RWR9 reads Surface Airways Observations (SAO's) from the local AFOS database and reformats them into plain language collectives for Hourly, State, and Regional Weather Roundups. Output is suitable for direct transmission on the state weather wire. Program options also allow the creation of a locally stored Broadcast Text, which contains observations to be broadcast on the weather radio.

B. Program Environment

RWR9 is designed to operate in the background partition of the AFOS minicomputer at either a Weather Service Forecast Office (WSFO) or a Weather Service Office (WSO). The bulk of the program is written in Data General's FORTRAN IV although two subroutines are written in assembly language. The program will execute in 9K of core memory.

Approximately 100 RDOS blocks of disk storage is required to operate the program. The program save (.SV) and overlay (.OL) files require 90 blocks and may reside on either DP0 or DP0F. Several blocks are required on DP0 for directory files (.DT), and several blocks are required on DP0F for the temporary output file (.SC).

Execution time is variable and depends on the number of observations to be processed. In general, short messages, such as the Hourly Roundup, require less than one minute for execution. Longer messages, such as the Regional Roundup, may require two or three minutes.

Three input/output channels are required. One channel reads the overlays, one channel reads the directory information, and one channel is used for output.

C. References

Program RWR9 evolved from an original regional weather program written by the author in August, 1980, using SAO decoder routines from Richard Thomas, now with AFOS Operations Division, National Weather Service Headquarters. Several Central Region forecasters and programmers made suggestions and changes during the development of the program. Major contributions came from John Hughes, WSFO Ann Arbor, and Pete Browning, WSFO St. Louis. Parts of the SAO decoder were updated to prevent error returns on correctly coded observations. Bob van Haaren, Scientific Services Division, Central Region Headquarters, coordinated the efforts and assisted in the preparation of this document.

II. APPLICATION PROGRAM DESCRIPTION

A. Functional Description

Program RWR9 converts coded SAO's into plain language variables which are output in tabular form. Input is from individual SAO's in the local database. For data to be output, a record observation (SA or RS) for the current hour must exist within the latest three versions available in the database. In order to compute the pressure tendency (e.g. rising, steady, falling), the program searches back a maximum of five additional versions for the previous hour's record observation. The current hour is determined from the system clock. From H+50 to H+59 the system time is rounded to the next whole hour. From H+00 to H+49 the system time is truncated to the current whole hour. For observations to be decoded, the WMO header and the time reported in the body of the observation must be within 30 minutes of the current hour.

Output variables are computed in the following manner:

1. STATION NAME is read from the program directory file. A station name may contain up to 14 characters.

2. PRESENT WEATHER is determined on a priority basis using the scheme described in Appendix 1.

3. CURRENT TEMPERATURE in degrees Fahrenheit is output as reported. Negative values are preceded with the letter B.

4. RELATIVE HUMIDITY in percent is computed from the temperature and dew point using the Goff-Gratch formulation from the Smithsonian Meteorological Tables.

5. WIND DIRECTION to eight compass points is determined from reported wind direction in tens of degrees using conversion table A3-5 from Federal Meteorological Handbook #1.

6. WIND SPEED in miles per hour is computed from reported wind speed in knots by multiplying by 1.15.

7. WIND GUSTS (if reported) are converted to miles per hour by multiplying by 1.15. Gusts are preceded by the letter G.

8. BAROMETRIC PRESSURE in inches of mercury is output from the reported altimeter setting.

9. PRESSURE TENDENCY is determined by comparing the altimeter setting from one hour ago to the current value. Tendency, if available, is output as a single letter (R, S, or F). If it is missing, the space remains blank.

10. WEATHER REMARKS include thunder and obstructions to

vision not reported as present weather. Thunder is output anytime it is occurring. Obstructions to vision are reported only if present weather is a cloud amount and visibility is three miles or less. A maximum of one weather remark is output.

11. WIND CHILL INDEX is output if the temperature is 35 degrees or less and the wind speed is 4 miles per hour or greater. The wind chill index is computed using the method described by Siple and Passel, Proceedings of the American Philosophical Society, Volume 89: 177-179, 1945. Negative values are are followed by the letter B.

12. DEW POINT temperature in degrees Fahrenheit as reported is output if the dew point option is selected. Negative values are followed by the letter B.

If the directory indicates that a station must always be output, and a current observation cannot be found; the station name and the phrase NOT AVBL are output.

One line of output is used for each station. The maximum length of the line is 71 characters plus a carriage return.

B. Program Structure

Program execution is controlled by a directory file which contains message headings, names of stations, and lines of text. Any of several directory files may be used. The proper directory file and other options are selected with global switches in the program command line.

The header information for the output file is determined as follows:

1. The product key of the output message is copied from the product key in which the directory file was created.

2. The message addressee defaults to 000. If the transmit option (/X) is selected, the message address becomes the regional or local addressee which has been octal edited into RWR9.SV.

3. Binary date/time is set to missing (-1). Thus, the current system time is used as the message creation time.

4. Message priority is determined from the default priority stored in the key record of the output product. To change this priority, use the KEY: command at an AFOS console.

5. Message type is fixed as 0.

6. The WMO header of the message is extracted from the WMO header of the directory file. The date and time are updated

to the current hour.

7. One blank line and the first two lines of the mass media header are read from the directory file. The date/time line of the mass media header is supplied by the program based on the current hour.

The program then steps through the station keys and lines of text. If a line begins with a plus (+), an observation is decoded and output only if it is current. If a line begins with a minus (-), the station name is always output; and if a current observation cannot be found or decoded, NOT AVBL is printed. A line beginning with any other character is assumed to be a line of text and is output verbatim. Processing halts upon reading the end of text character of the directory file.

The initialization, processing, and output routines of program RWR9 are contained in individual subroutines. These subroutines are loaded into the program overlay area as they are needed. Loading of the overlays and execution of the subroutines is diagramed in Appendix 2.

Program input comes from the directory file and from SAO's stored in the database. The directory file is created with AFOS message composition using the format described in Appendix 3. The finished directory is copied to disk with the AFOS SAVE: command and the appropriate filename.

Program output is written to file DPOF:RWR9.SC. This file is in a suitable format for direct storage into the database with subroutine FSTORE.

The program save and overlay files are created with the following load line:

RLDR RWR9 [RWR1 MMHDR CSUB IDAWK JONES,RWR2 FSTOR FORK, RWR3 LTR,RWR4 RWR5 RWR6 ALG EXP,AIRXA,AIRXB,AIRXC] NUMBR ORGO ANDEQ ANDGO UTIL.LB TOP.LB FORT.LB

Subroutines RWR9, RWR1, RWR2, RWR3, RWR4, RWR5, and RWR6 are main modules of the RWR9 program. Subroutines JONES and LTR perform auxillary operations.

Subroutines AIRXA, AIRXB, AIRXC, NUMBR, ORGO, ANDEQ, and ANDGO compose the SAO decoder.

To minimize core requirements, subroutines used in only one overlay were extracted from their parent libraries and are loaded as discrete .RB's. Subroutines MMHDR, CSUB, IDAWK, FSTOR, and FORK were extracted from TOP.LB. ALG and EXP were extracted from FORT.LB.

The program load map is presented in Appendix 5.

III. PROGRAM TESTING AND IMPLEMENTATION

A. Procedures for Program Implementation

At a WSFO the program save and overlay files should reside on DPOF. Move RWR9.SV and RWR9.OL to DPOF and create links to these files on DPO. At a WSO it may be preferable to move the files directly to DPO.

Create the necessary directory file(s) using AFOS message composition and the product key that is to receive the program output. Follow the format described in Appendix 3.

Copy the directory to an RDOS file on DPO using the AFOS SAVE: command and one of the following filenames:

HRWX.DT	- for the Hourly Weather Roundup
HRWXN.DT	- for the abbreviated, nighttime Hourly Weather Roundup
RWX.DT	- for the State or Regional Weather Roundup
RWXN.DT	- for the abbreviated, nighttime State or Regional Weather Roundup

The following example illustrates the proper SAVE command for the Regional Weather Roundup issued by WSFO Topeka:

```
SAVE:TOPRWRTOP RWX.DT
```

Once a directory file has been created, its associated product key may be changed with the AFOS STORE: command. Suppose RWX.DT was created with the product key TOPTPTTOP and the key needs to be changed to TOPRWRTOP. Merely enter the following command at an ADM:

```
STORE:RWX.DT TOPRWRTOP
```

If the WMO header must also be changed, enter the above STORE: command, edit TOPRWRTOP, and SAVE:TOPRWRTOP RWX.DT.

A MACRO simplifies the execution command required from the operator or issued by a PROCEDURE. It can tailor the program to the requirements of a particular office. Usual practice is to name the MACRO file NNN.MC, where NNN is the product category of the desired output (HRR for Hourly Weather Roundup, RWR for Regional Weather Roundup, etc.). Create MACROs at the Keyboard/Printer (Dasher) with the text editor or at an ADM with the M:F/ command.

The following program options are available through global switches used in the command line. The appropriate switches are included during the creation of a MACRO.

/A Generate a console alert upon program completion

/E	Eastern time zone
/D	Output dew point as a remark regardless of temperature
/H	Use the letter H as a prefix to the directory filename
/M	Mountain time zone
/N	Use the letter N as a suffix to the directory filename between the hours of 04Z and 11Z
/P	Pacific time zone
/R	Omit pressure tendencies (for regional roundups)
/S	Suppress "sunny" in sky condition terms
/T	Force standard time year-round
/X	Xmit the product on the RDC

If no options are selected, the program defaults to Central time, directory RWX.DT, and addressee 000.

The following examples illustrate the use of MACRO files.

```
RWR9/E/H/D/X
```

will execute RWR9 in the Eastern time zone, using the HRWX.DT directory, outputting dew points, and transmitting the product on the RDC.

```
RWR9/M/X/R
```

will execute RWR9 in the Mountain time zone with transmission on the RDC and no pressure tendencies.

If more than one Hourly Weather Roundup and one State or Regional Weather Roundup is required, link an appropriate directory to the required filename before initiating execution. In the following example, RUN:RWR will create a regional roundup and transmit it on the RDC. RUN:BRT will create a broadcast text for the weather radio, store the product locally, and generate a console alert upon completion.

RWR.MC contains

```
UNLINK RWX.DT
LINK RGNLWX.DT RWX.DT
RWR9/R/X
```

BRT.MC contains

```
UNLINK RWX.DT
LINK RADIO.DT RWX.DT
RWR9/A
```

An AFOS PROCEDURE may be used to execute the RWR9 program at a specified time. The following PROCEDURE, named "HRWX" will execute an Hourly Weather Roundup at H+58 and a Regional Weather Roundup at H+15:

```
PAUSE 01
WTIME 58
RUN:HRR
WTIME 15
RUN:RWR
"HRWX"
```

This PROCEDURE requires MACRO files named HRR.MC and RWR.MC. Although the necessary global switches may be used in the PROCEDURE, the above method simplifies manual execution in the event of a PROCEDURE failure.

The AFOS addressee used with the /X option and the limits of the nighttime directory used with the /N option may be changed by octal editing the following locations of RWR9.SV:

434	First two characters of addressee
435	Third character of addressee plus a null
436	Low limit of nighttime directory
437	High limit of nighttime directory

Remember to place a decimal point after decimal numbers.

B. Sample Run for Testing

After creating the necessary directory file(s) described above, the program should be run several times without the /X option. Output can be checked for possible errors in the directory. Fatal directory errors cause a message to be typed at the KB/P and processing to halt. Non-fatal errors may produce undesired output. If a given SAO is not routinely stored in the local database, the phrase NOT STORED is output after the station name. This key will need to be PIL edited into the database.

Sample directories and output are shown in Appendix 4.

C. Error Codes and Restrictions

All fatal errors are identified with a typeout on the KB/P. If the program was run from an ADM, only the program message will be typed. Rerun the program from the KB/P to receive the complete error message including that part typed by RDOS.

DIRECTORY ERROR	Program message
FILE DOES NOT EXIST: RWR9.SV	RDOS message

The program has no known restrictions.

LIST OF APPENDICES

APPENDIX 1

The following scheme is used to determine present weather based on the coded weather and cloud groups from the airways observation. The weather terms are listed in priority order with the highest priority terms first.

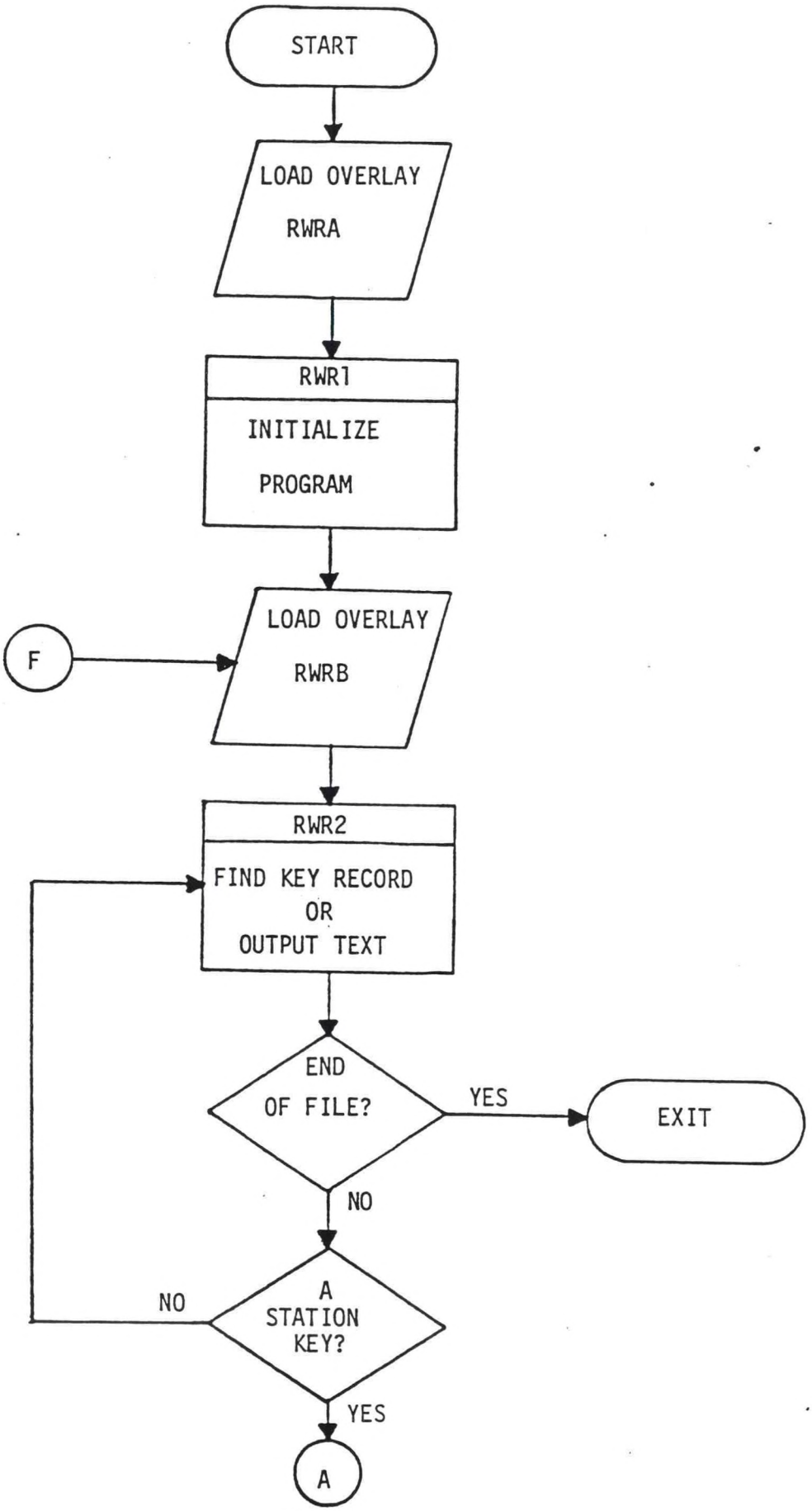
CODED SYMBOLS	PRESENT WEATHER
A	HAIL
T (W,IP,S,R, or L) -	TSHWR
T (W,IP,S,R, or L)	TSTORM
Z R	FRZGRAIN
Z L	FRZGDRZL
IP W	SLEET
R S W	RAINSNOW
S W	SNOWSHWR (Note 2)
R W	SHWR
IP	SLEET
R S	RAINSNOW
S (but not BS)	SNOW (Note 2)
R	RAIN
L	DRZL
BS	BLWGSNOW
BD	BLWGDUST
F (Note 1)	FOG
D (Note 1)	DUST
H (Note 1)	HAZE
K (Note 1)	SMOKE
OVC or X (but not -X)	CLDY
BKN or -OVC	MOCLDY or PTSUNNY
SCT or -BKN	PTCLDY or MOSUNNY
CLR, -SCT or -X	CLEAR or SUNNY
?	UNKNOWN

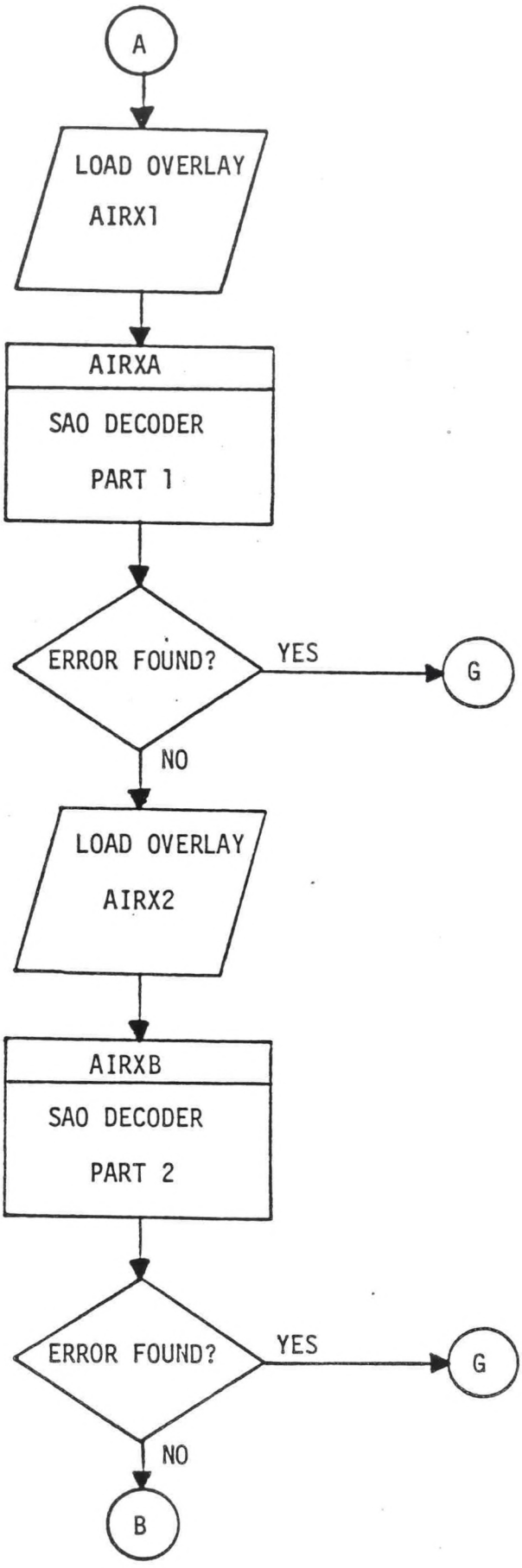
NOTES:

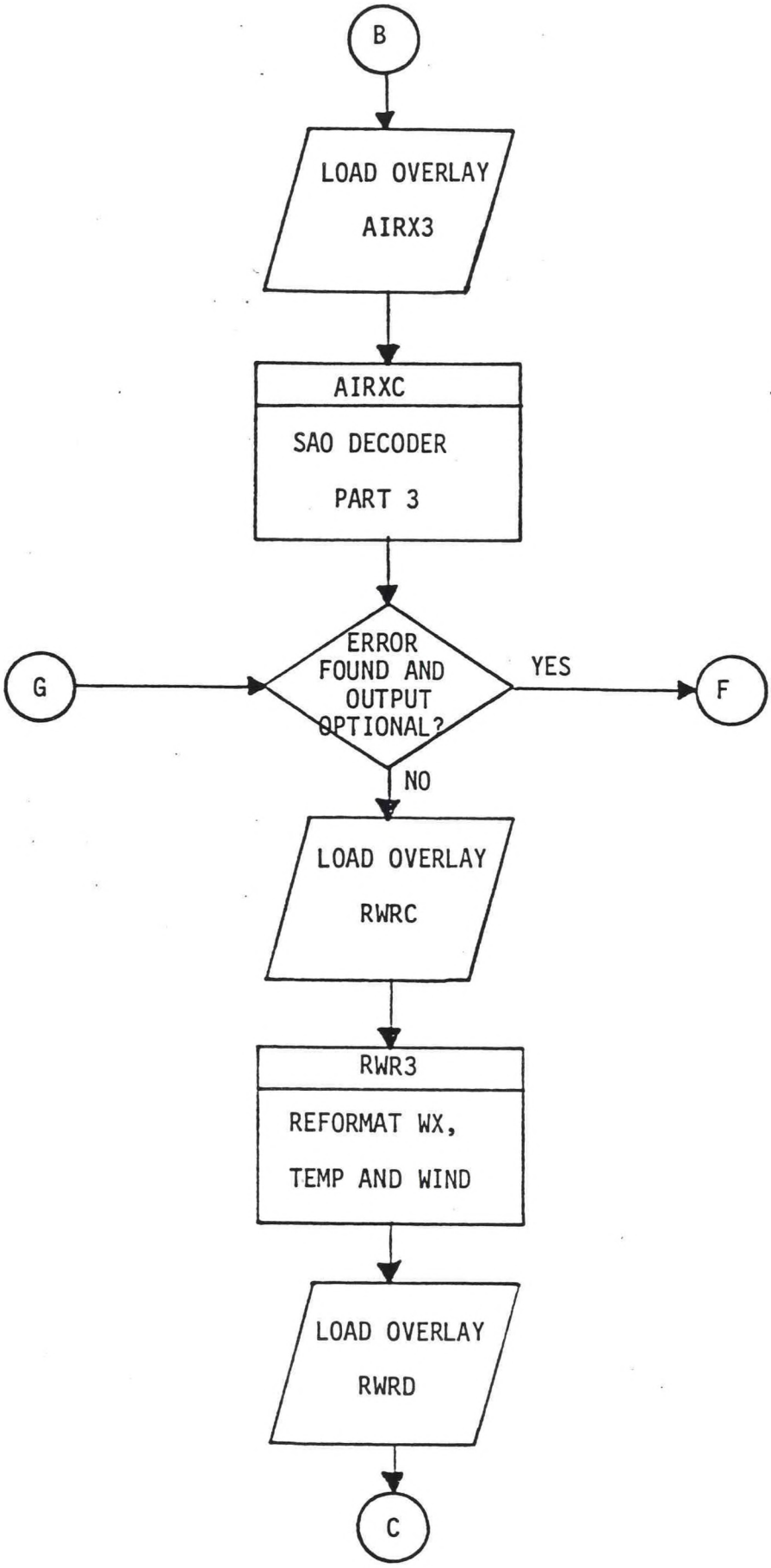
1. Obstructions to vision are reported as present weather only if visibility is one mile or less. The first letter reported (i.e., the predominant obstruction) is used as the present weather.

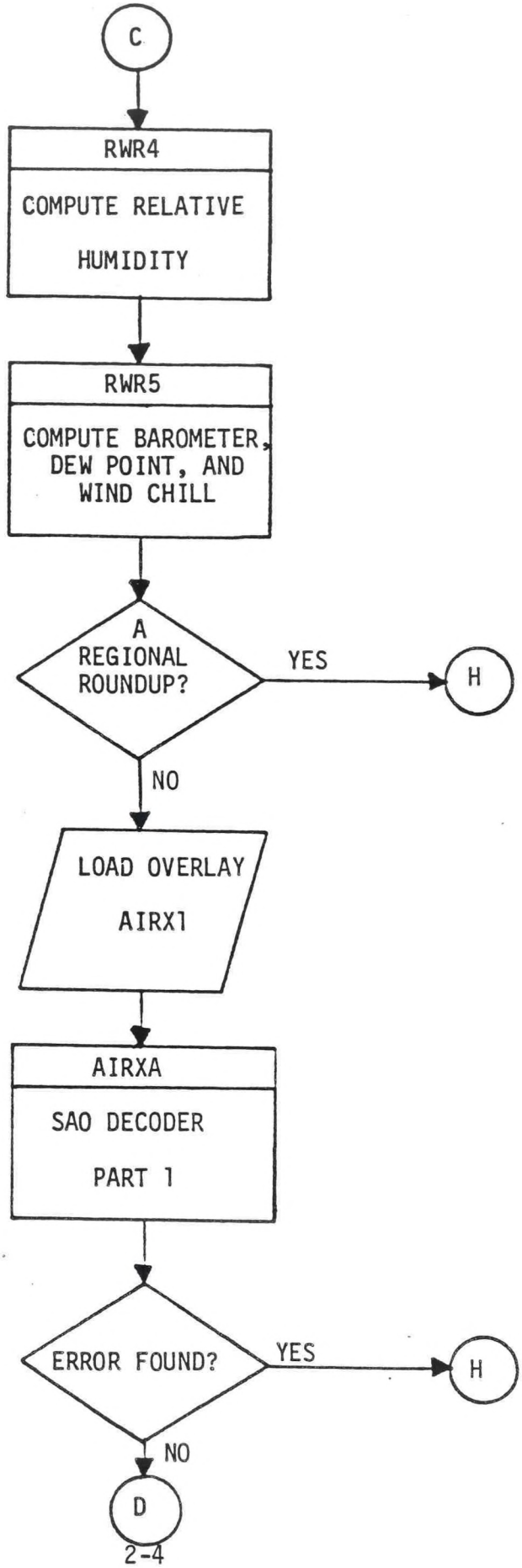
2. SNOW and SNOWSHWR are changed to FLURRIES if visibility is seven miles or greater.

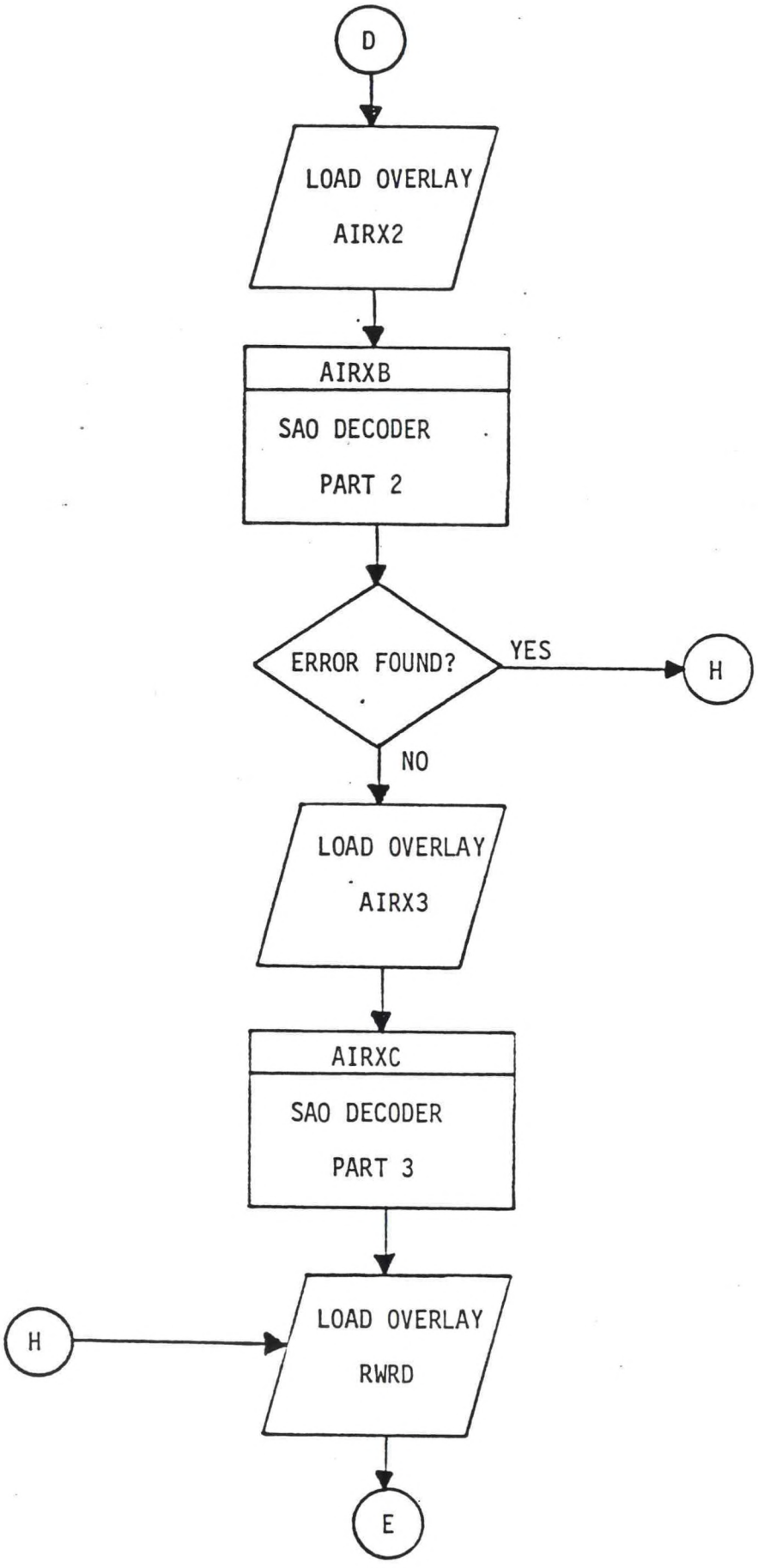
APPENDIX 2

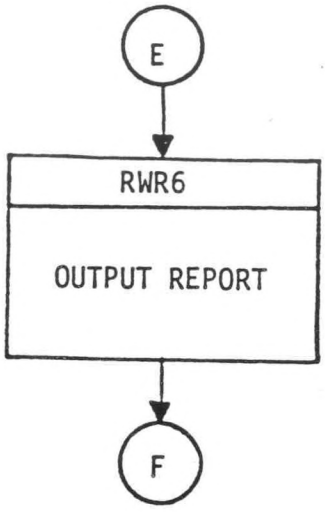

START
LOAD OVERLAY
RWRA
RWR1
INITIALIZE
PROGRAM
LOAD OVERLAY
RWRB
F
RWR2
FIND KEY RECORD
OR
OUTPUT TEXT
END
OF FILE?
YES
EXIT
NO
A
STATION
KEY?
NO
YES
A


A
LOAD OVERLAY
AIRX1
AIRXA
SAO DECODER
PART 1
ERROR FOUND?
YES
G
NO
LOAD OVERLAY
AIRX2
AIRXB
SAO DECODER
PART 2
ERROR FOUND?
YES
G
NO
B


B
LOAD OVERLAY
AIRX3
AIRXC
SAO DECODER
PART 3
ERROR
FOUND AND
OUTPUT
OPTIONAL?
YES
G
F
NO
LOAD OVERLAY
RWRC
RWR3
REFORMAT WX,
TEMP AND WIND
LOAD OVERLAY
RWRD
C


C
RWR4
COMPUTE RELATIVE HUMIDITY
RWR5
COMPUTE BAROMETER, DEW POINT, AND WIND CHILL
A REGIONAL ROUNDUP?
YES
H
NO
LOAD OVERLAY AIRX1
AIRXA
SAO DECODER PART 1
ERROR FOUND?
YES
H
NO
D


D
LOAD OVERLAY
AIRX2
AIRXB
SAO DECODER
PART 2
ERROR FOUND?
YES
H
NO
LOAD OVERLAY
AIRX3
AIRXC
SAO DECODER
PART 3
H
LOAD OVERLAY
RWRD
E


E
RWR6
OUTPUT REPORT
F

APPENDIX 3

Directory files for program RWR9 should be created according to the following format.

The first line of the directory contains the WMO header to be used in the output message. Only the first two groups are used; the date/time group is updated during execution.

The next three lines will be written to the output message verbatim. These lines usually contain a blank line and the first two lines of the mass media header.

The program will insert the date/time line of the mass media header, one blank line, and the column headings at this point. No provision is needed in the directory for these lines.

The remaining lines of the directory contain AFOS product keys of observations to be decoded or lines of text to be written verbatim. A product key is identified by a plus or a minus in column one. Any other character indicates a line of text.

A minus preceding the product key means that the station is always output, regardless of whether a current observation is found. A plus means that the station is output only if a current observation is available. The plus option is convenient for part-time stations.

A 9-character product key is followed by one space and up to 14 characters of city name.

There is no special ending for the directory information. The program terminates upon finding the AFOS-supplied end of text character.

Proper WMO headers and mass media headers for the various weather roundup products are listed in chapter C-21 of the Weather Service Operations Manual.

```
SXUS2 KTOP 010200

HOURLY WEATHER ROUNDUP
NATIONAL WEATHER SERVICE TOPEKA KS

-TOPSAOICT WICHITA
-TOPSAOTOP TOPEKA
-TOPSAOCNK CONCORDIA
-TOPSAOGLD GOODLAND
-TOPSAODDC DODGE CITY
-STLSAOMCI K.C. INTNL
```

Sample directory HRWX.DT for Hourly Weather Roundup

```
SXUS2 KTOP 010100

HOURLY WEATHER ROUNDUP
NATIONAL WEATHER SERVICE TOPEKA KS
700PM CST FRI DEC 31 1982

 CITY           SKY/WX    TEMP/RH  WIND     PRES     REMARKS

WICHITA         CLEAR     29  88   SW8      30.27S   WCI 17
TOPEKA          CLEAR     31  63   SW8      30.22F   WCI 20
CONCORDIA       CLEAR     26  78   SW12     30.15F   WCI 8
GOODLAND        CLEAR     24  65   SW12     30.16R   WCI 5
DODGE CITY      CLEAR     26  68   SW13     30.23S   WCI 6
K.C. INTNL      CLEAR     32  69   SW15     30.20F   WCI 11
```

Sample output for Hourly Weather Roundup

```
ABUS22 KTOP 010200

KANSAS WEATHER ROUNDUP
NATIONAL WEATHER SERVICE TOPEKA KS

-TOPSAOCNK CONCORDIA
+TOPSAOCNU CHANUTE
-TOPSAODDC DODGE CITY
+TOPSAO1K5 ELKHART
+TOPSAOEMP EMPORIA
-TOPSAOGCK GARDEN CITY
-TOPSAOGLD GOODLAND
+TOPSAOHLC HILL CITY
+TOPSAOMHK MANHATTAN
+TOPSAOP28 MEDICINE LODGE
-TOPSAORSL RUSSELL
-TOPSAOSLN SALINA
-TOPSAOICT WICHITA
```

Sample directory RWX.DT for State Weather Roundup

```
ABUS22 KTOP 010100

KANSAS WEATHER ROUNDUP
NATIONAL WEATHER SERVICE TOPEKA KS
700PM CST FRI DEC 31 1982

 CITY            SKY/WX     TEMP/RH  WIND     PRES     REMARKS

CONCORDIA        CLEAR      26  78   SW12     30.15F   WCI 8
CHANUTE          CLEAR      30  72   S6       30.24S   WCI 23
DODGE CITY       CLEAR      26  68   SW13     30.23S   WCI 6
ELKHART          UNKNOWN    23  62   SW8      30.23R   WCI 10
GARDEN CITY      CLEAR      14  88   SW8      30.21R   WCI 0
GOODLAND         CLEAR      24  65   SW12     30.16R   WCI 5
MANHATTAN        CLEAR      27  71   W10      30.20F   WCI 11
MEDICINE LODGE   UNKNOWN    33  56   SW6      30.25R   WCI 27
RUSSELL          CLEAR      27  85   SW13     30.18S   WCI 7
SALINA           NOT AVBL
WICHITA          CLEAR      29  88   SW8      30.27S   WCI 17
```

Sample output for State Weather Roundup

```
ABUS32 KTOP 010200

REGIONAL WEATHER ROUNDUP
NATIONAL WEATHER SERVICE TOPEKA KS

...KANSAS...
+TOPSAOCNU CHANUTE
+TOPSAO1K5 ELKHART
+TOPSAOEMP EMPORIA
-TOPSAOGCK GARDEN CITY
+TOPSAOHLC HILL CITY
+TOPSAOMHK MANHATTAN
+TOPSAOP28 MEDICINE LODGE
-TOPSAORSL RUSSELL
-TOPSAOSLN SALINA

...WYOMING...
-CYSSAOCYS CHEYENNE
-CYSSAOLAR LARAMIE

...NEBRASKA...
+OMASAOBIE BEATRICE
+OMASAOFNB FALLS CITY
-OMASAOGRI GRAND ISLAND
+OMASAOHSI HASTINGS
+OMASAOIML IMPERIAL
-OMASAOLNK LINCOLN
+OMASAOMCK MCCOOK
-OMASAOLBF NORTH PLATTE
-OMASAOOMA OMAHA
-OMASAOBFF SCOTTSBLUFF
+OMASAOSNY SIDNEY

...COLORADO...
-DENSAOCOS COLO SPRINGS
-DENSAODEN DENVER
+DENSAOLHX LA JUNTA
-DENSAOPUB PUEBLO

...MISSOURI...
-STLSAOJLN JOPLIN
-STLSAOMCI KANSAS CITY
+STLSAOSTJ ST. JOSEPH
-STLSAOSGF SPRINGFIELD

...TEXAS...
-LBBSAOAMA AMARILLO
-LBBSAODHT DALHART

...OKLAHOMA...
-OKCSAOGAG GAGE
-OKCSAOOKC OKLAHOMA CITY
-OKCSAOPNC PONCA CITY
-OKCSAOTUL TULSA

...ARKANSAS...
-LITSAOFYV FAYETTEVILLE
-LITSAOLIT LITTLE ROCK
```

Sample directory RWX.DT for Regional Weather Roundup

ABUS32 KTOP 010100

REGIONAL WEATHER ROUNDUP
NATIONAL WEATHER SERVICE TOPEKA KS
700PM CST FRI DEC 31 1982

CITY	SKY/WX	TEMP	RH	WIND	PRES	REMARKS
...KANSAS...						
CHANUTE	CLEAR	30	72	S6	30.24	WCI 23
ELKHART	UNKNOWN	26	55	SW9	30.22	WCI 11
GARDEN CITY	CLEAR	14	88	SW8	30.21	WCI 0
MEDICINE LODGE	UNKNOWN	34	59	SW6	30.24	WCI 28
RUSSELL	NOT AVBL					
SALINA	NOT AVBL					
...WYOMING...						
CHEYENNE	CLEAR	21	71	NW15	30.14	WCI 38
LARAMIE	NOT AVBL					
...NEBRASKA...						
BEATRICE	CLEAR	19	84	S6	30.15	WCI 11
FALLS CITY	CLEAR	26	88	SW9	30.20	WCI 11
GRAND ISLAND	CLEAR	23	74	SW14	30.08	WCI 0
HASTINGS	CLEAR	19	88	SW9	30.12	WCI 3
LINCOLN	CLEAR	22	84	S5	30.13	WCI 18
MCCOOK	CLEAR	24	71	W8	30.16	WCI 11
NORTH PLATTE	CLEAR	24	78	W8	30.09	WCI 11
OMAHA	CLEAR	20	84	S6	30.12	WCI 12
SCOTTSBLUFF	CLEAR	29	48	W15	30.10	WCI 7
SIDNEY	UNKNOWN	27	66	W10	30.10	WCI 11
...COLORADO...						
COLO SPRINGS	CLEAR	20	54	N7	30.17	WCI 9
DENVER	PTCLDY	20	68	S7	30.13	WCI 9
LA JUNTA	CLEAR	21	62	SW6	30.19	WCI 14
PUEBLO	CLEAR	18	84	N6	30.18	WCI 10
...MISSOURI...						
JOPLIN	CLEAR	32	66	SW5	30.27	WCI 29
KANSAS CITY	CLEAR	32	69	SW15	30.20	WCI 11
SPRINGFIELD	CLEAR	32	66	SW5	30.26	WCI 29
...TEXAS...						
AMARILLO	CLDY	26	60	SW6	30.24	WCI 19
DALHART	NOT AVBL					
...OKLAHOMA...						
GAGE	NOT AVBL					
OKLAHOMA CITY	CLEAR	33	69	NW7	30.30	WCI 24
PONCA CITY	NOT AVBL					
TULSA	CLEAR	39	55	W3	30.31	
...ARKANSAS...						
FAYETTEVILLE	NOT AVBL					
LITTLE ROCK	MOCLDY	38	67	CALM	30.33	

Sample output for Regional Weather Roundup

WOUS00 KICT 020000

BROADCAST TEXT
NATIONAL WEATHER SERVICE WICHITA KS
600PM CST SAT JAN 1 1983

CITY	SKY/WX	TEMP	RH	WIND	PRES	REMARKS
WICHITA	CLEAR	34	78	N6	30.24S	WCI 28
CHANUTE	CLEAR	34	78	N6	30.24S	WCI 28
DODGE CITY	CLEAR	29	69	NW8	30.23S	WCI 17
CONCORDIA	CLEAR	30	78	N10	30.23F	WCI 14
TOPEKA	CLEAR	31	66	S5	30.28R	WCI 28
JOPLIN	CLDY	38	62	S6	30.23R	
PONCA CITY	CLDY	33	75	NW7	30.23F	WCI 24

Sample output of a Broadcast Text (TOPBRTICT)

Double spacing is achieved through the use of blank lines in the directory file.

APPENDIX 5

```
RWR9.SV         LOADED BY RLDR REV 06.00 AT 11:53:16 03/24/83
 .MAIN    000452
          001413
          000,000  RWR1      001413
                   MMHDR     004311
                   CSUB      005121
                   IDAWK     005241
                   JONES     005373 003770
          000,001  RWR2      001413
                   FSTOR     002413
                   FORK      002477 001204
          000,002  RWR3      001413
                   LTR       005743 004367
          000,003  RWR4      001413
                   RWR5      002401
                   RWR6      003707
                   ALG       004204
                   EXP       004411 003160
          000,004  AIRXA     001413 003036
          000,005  AIRXB     001413 004342
          000,006  AIRXC     001413 004320
          006013
NUMBR     006013
ORGO      006052
ANDEQ     006106
ANDGO     006140
TIME      006172
DATE      006211
PACK      006234
UNPAC     006261
COMCM     006316
FCOM      006401
ERROR     006426
SPCHR     006457
CRAND     006521
DELET     006532
EXIT      006543
GCHN      006553
KLOSE     006564
OPENR     006574
OPENN     006607
OVLOD     006622
RDL       006635
RDS       006647
WRL       006662
WRS       006674
RTN2      006707
RTN4      006715
BGDR      006726
EBG       007677
ATUMO     010757
APAGE     011023
I         011023
RESET     011344
CLOSE     011367
IOPRE     011442
BSTRI     012063
FREDI     012121
FALOC     012226
FSBR      012246
CGT       012437
MOD       012463
IABS      012474
SMPY      012505
SDVD      012526
INT       012546
FPWER     012557
SQRT      012623
PLY1      012765
BREAK     013015
FLIP      013045
AFRTN     013071
FL        013076
STREG     014022
LDREG     014046
MVF       014077
MOVE      014115
LD0       014160
STOP      014177
FINIT     014247
ISARE     014272
FLINK     014303
RTER      014446
WRCH      014657
BDASC     014674
BASC      014755
COUT      015011
LDSTB     015034
CPYAR     015044
MAD       015106
FARG0     015131
ARGUM     015164
FPZER     015234
FPTRS     015234
DUMMY     015234
ARDUM     015234
EMPYD     015235
TMIN      015246
NSAC3     015342

NMAX      015342
ZMAX      000306
CSZE      000703
 EST      000000
 SST      000000
```

```
C       PROGRAM RWR9, HOURLY AND REGIONAL WX ROUNDUP IN 9K
C
C       CONTAINS CONTRIBUTIONS BY RICH THOMAS, WARREN SUNKEL,
C       HUMPHREY PLANER, JOHN HUGHES, AND PETE BROWNING
C
C       DEDICATED TO THE MET TECHS WHO PERFORMED THESE TASKS
C       MANUALLY FOR SO MANY YEARS
C
        EXTERNAL RWRA,RWRB,RWRC,RWRD,AIRX1,AIRX2,AIRX3
        COMMON/AAA/IADR(2),NLIMS(2),IOF(10)
        COMMON ICP,ICP1,J1,IY,IHOUR,IBUF(128),IDATA(256),JER,ISTN,MINCP,
     1  ITMIN,ICAN,ISW(2),ICH,JCH,JSTN(8),IKEY(5),KSAV,IDPH,ITPH,IPPH,JBUF(36)
        DIMENSION KEY(5)

        DATA IADR/"C  "/; ADDRESSEE TO USE FOR THE XMIT OPTION
        DATA NLIMS/4,11/; LIMITS FOR NIGHTTIME DIRECTORY (Z)
        DATA IOF/"DP0F:RWR9.SC",0/; OUTPUT FILENAME
        ISA=1; SKIP SPECIALS
        CALL GCHN(IOCH,IER)
        CALL OPENR(IOCH,"RWR9.OL",0,IER); OPEN THE OVERLAY FILE
   10   CALL ERROR(IER,"OVERLAY ERROR")
        CALL OVLOD(IOCH,RWRA,-1,IER)
        IF(IER.NE.1)GO TO 10
        CALL RWR1(IDAY,ITIME); INITIALIZE PROGRAM AND BUILD HEADERS
   20   CALL OVLOD(IOCH,RWRB,-1,IER)
        IF(IER.NE.1)GO TO 10
   24   CALL RWR2(KEY,NVER,IER); FIND KEY RECORD OR OUTPUT TEXT
        IF(IER.NE.1)GO TO 24; NOT A STATION
C
C       READ CURRENT HOUR'S OBSERVATION
C       SEARCHING THROUGH A MAXIMUM OF 3 VERSIONS
C
        CALL OVLOD(IOCH,AIRX1,-1,IER)
        IF(IER.NE.1)GO TO 10
        CALL AIRXA(KEY,IDAY,ITIME,ISA,3)
        IF(JER.NE.0)GO TO 40
        CALL OVLOD(IOCH,AIRX2,-1,IER)
        IF(IER.NE.1)GO TO 10
        CALL AIRXB
        IF(JER.NE.0)GO TO 40
        CALL OVLOD(IOCH,AIRX3,-1,IER)
        IF(IER.NE.1)GO TO 10
        CALL AIRXC
   40   IF(JER.NE.1.AND.KSAV.EQ.53K)GO TO 20; SKIP OUTPUT
        CALL OVLOD(IOCH,RWRC,-1,IER)
        IF(IER.NE.1)GO TO 10
        CALL RWR3; REFORMAT WX, TEMP, WIND
        CALL OVLOD(IOCH,RWRD,-1,IER)
        IF(IER.NE.1)GO TO 10
        CALL RWR4; COMPUTE RELATIVE HUMIDITY
        CALL RWR5; COMPUTE BAROMETER, DEW POINT AND WIND CHILL
        IBUF(41)=-1
        IF((ISW(2).AND.40000K).NE.0)GO TO 50; REGIONAL ROUNDUP
C
C       READ PREVOUS HOUR'S ALTIMETER SETTING
```

```
C       SEARCHING THROUGH A MAXIMUM OF 5 ADDITIONAL VERSIONS
C
        CALL OVLOD(IOCH,AIRX1,-1,IER)
        IF(IER.NE.1)GO TO 10
        CALL AIRXA(KEY,IDPH,ITPH,ISA,5)
        IF(JER.NE.0)GO TO 50
        CALL OVLOD(IOCH,AIRX2,-1,IER)
        IF(IER.NE.1)GO TO 10
        CALL AIRXB
        IF(JER.NE.0)GO TO 50
        CALL OVLOD(IOCH,AIRX3,-1,IER)
        IF(IER.NE.1)GO TO 10
        CALL AIRXC
   50   CALL OVLOD(IOCH,RWRD,0,IER)
        IF(IER.NE.1)GO TO 10
        CALL RWR6; OUTPUT
        GO TO 20; PROGRAM EXITS FROM SUBROUTINE RWR2
        END
```

```
      OVERLAY RWRA
      SUBROUTINE RWR1(IDAY,ITIME); INITIALIZE PROGRAM AND BUILD HEADERS

      COMMON IDUM(5),IHDR(128),IDATA(256),JDUM(5),ISW(2),ICH,JCH,
     1 ISTN(8),IKEY(5),KSAV,IDPH,ITPH,IPPH,JBUF(36)
      COMMON/AAA/IADR(2),NLIMS(2),IOF(10)
      COMMON/R91/ISR(12),ISS(12),IHDG(30)

      DATA ISR/10,9,9,8,8,7,7,8,8,9,9,10/; LATEST SUNRISE +2
      DATA ISS/15,16,16,17,17,18,18,17,16,15,15,15/; EARLIEST SUNSET -2
      DATA IHDG/" CITY           SKY/WX   TEMP/RH  WIND     PRES   REMARKS ",
     1 6412K/
C
C     FETCH THE GLOBAL SWITCHES AND SYSTEM DTG
C
      CALL FCOM(ICH,IER)
      CALL COMCM(ICH,IDATA,NB,ISW,IER)
      CALL ERROR(IER,"COM.CM ERROR")
      CALL KLOSE(ICH,IER)
      CALL DATE(IDUM,IER)
      CALL TIME(JDUM,IER)
      CALL DATE(IDATA,IER)
      IF(JDUM(2).LT.50)GO TO 14
C
C     ROUND TIME TO NEXT HOUR FOR HOURLY ROUNDUP
C
      ITPH=JDUM(1)
      JDUM(1)=JDUM(1)+1
      IF(JDUM(1).LT.24)GO TO 16
      JDUM(1)=0
      CALL TMRW(IDUM)
      GO TO 16
C
C     CALCULATE PREVIOUS HOUR FOR FINDING PRESSURE TENDENCY
C
   14 ITPH=JDUM(1)-1
      IF(ITPH.GE.0)GO TO 16
      ITPH=23
      CALL YDA(IDATA)
   16 IDPH=IDATA(2)
      ITPH=ITPH*100
C
C     STORE DAY AND TIME OF CURRENT OBSERVATIONS
C
      IDAY=IDUM(2)
      ITIME=JDUM(1)*100
C
C     COMPUTE MASS MEDIA HEADER AND ANSWER DAY/NIGHT QUESTION
C
      IF((ISW(2).AND.10000K).NE.0)CALL JONES; T TO FORCE STANDARD TIME
      IZ=2; DEFAULT TO CENTRAL TIME
      IF((ISW(1).AND.4000K).NE.0)IZ=1; E FOR EASTERN
      IF((ISW(1).AND.10K).NE.0)IZ=3; M FOR MOUNTAIN
      IF((ISW(1).AND.1).NE.0)IZ=4; P FOR PACIFIC
      IDUM(3)=MOD(IDUM(3),100)
```

```
      DO 203 I=1,3
  203 IHDR(I)=IDUM(I)
      IHDR(4)=JDUM(1)
      IHDR(5)=0
      ITEMP=IHDR(4)-(IZ+4); ZULU TO LOCAL TIME
      IF(ITEMP.LT.0)ITEMP=ITEMP+24
      IF(ITEMP.GE.ISR(IHDR(1)).AND.ITEMP.LE.ISS(IHDR(1)))GO TO 22
      ISW(2)=ISW(2).OR.20000K; SUPRESS 'SUNNY' WITH S BIT
   22 CALL MMHDR(-IZ,IHDR,IER)
C
C     BUILD FILENAME OF DIRECTORY [H]RWX[N].DT
C
      N=1
      IDATA(N)=110K; H
      IF((ISW(1).AND.400K).NE.0)N=N+1
      IDATA(N)=122K; R
      IDATA(N+1)=127K; W
      IDATA(N+2)=130K; X
      N=N+3
      IDATA(N)=116K; N
      IF((ISW(1).AND.4).EQ.0)GO TO 30; NOT N
      ITEMP=ITIME/100
      IF(ITEMP.GE.NLIMS(1).AND.ITEMP.LE.NLIMS(2))N=N+1
   30 IDATA(N)=56K;
      IDATA(N+1)=104K; D
      IDATA(N+2)=124K; T
      IDATA(N+3)=0
      N=N+3
      CALL PACK(IDATA,N,IDATA)
C
C     OPEN DIRECTORY AND FETCH PRODUCT KEY
C
      CALL GCHN(ICH,IER)
      CALL OPENR(ICH,IDATA,0,IER)
   32 CALL ERROR(IER,"DIRECTORY ERROR")
      NB=20
      CALL RDS(ICH,IDATA,NB,IER)
      IF(IER.NE.1)GO TO 32
      CALL UNPACK(IDATA,9,IHDR(100))
      IHDR(109)=0
      CALL PACK(IHDR(100),10,IKEY); OUTPUT KEY
C
C     FETCH MESSAGE PRIORITY
C
      CALL KSRCF(IKEY,IDATA,IER)
      IF(IER.NE.1)IER=4; ILLEGAL FILENAME
      CALL ERROR(IER,"OUTPUT KEY")
      IPRI=ISHFT((IDATA(5).AND.70K),-3)
C
C     OPEN THE OUTPUT FILE
C
      CALL DELETE(IOF,IER)
      CALL CRAND(IOF,IER)
      CALL GCHN(JCH,IER)
      CALL OPENN(JCH,IOF,0,IER)
   34 CALL ERROR(IER,"OUTPUT ERROR")
C
C     BUILD HEADER BLOCK
C
      CALL UNPACK(IKEY,9,IDATA)
```

```
      IDATA(10)=60K; 0
      IDATA(11)=60K; 0
      IDATA(12)=60K; 0
      IF((ISW(2).AND.400K).NE.0)CALL UNPACK(IADR,3,IDATA(10))
      CALL PACK(IDATA,12,IDATA)
      IDATA(7)=-1
      IDATA(8)=-1
      IDATA(9)=ISHFT(IPRI,8)
      IDATA(10)=142600K
      NB=20
      CALL WRS(JCH,IDATA,NB,IER)
      IF(IER.NE.1)GO TO 34
C
C     MOVE THE WMO HEADER FROM THE DIRECTORY FILE
C
      CALL RDL(ICH,IHDR(40),NB,IER)
      IF(IER.NE.1)GO TO 32
      CALL UNPACK(IHDR(40),NB,IDATA)
      N=0
      DO 205 I=1,2
   40 N=N+1
      IF(IDATA(N).NE.40K)GO TO 40
  205 CONTINUE
      N=N+1
      DO 206 I=1,6
  206 IDATA(N+I-1)=IHDR(I)+60K
      N=N+6
      IDATA(N)=15K
      IDATA(N+1)=12K
      NB=N+1
      CALL PACK(IDATA,NB,IDATA)
      CALL WRS(JCH,IDATA,NB,IER)
      IF(IER.NE.1)GO TO 34
C
C     MOVE 3 LINES OF MEDIA HEADER FROM THE DIRECTORY
C
      DO 202 I=1,3
      CALL RDL(ICH,IDATA,NB,IER)
      IF(IER.NE.1)GO TO 32
      CALL WRL(JCH,IDATA,NB,IER)
      IF(IER.NE.1)GO TO 34
      NB=1
      CALL WRS(JCH,5000K,NB,IER); LF
      IF(IER.NE.1)GO TO 34
  202 CONTINUE
C
C     WRITE THE DATE/TIME LINE AND THE LEGEND
C
      CALL UNPACK(IHDR(7),28,IDATA)
      DO 204 I=25,28
      IF(IDATA(I).NE.0)GO TO 204 .
      IDATA(I)=15K
      IDATA(I+1)=12K
      GO TO 44
  204 CONTINUE
   44 NB=I+1
      CALL PACK(IDATA,NB,IDATA)
      CALL WRS(JCH,IDATA,NB,IER)
      IF(IER.NE.1)GO TO 34
      NB=2
```

```
CALL WRS(JCH,6412K,NB,IER); CR LF
IF(IER.NE.1)GO TO 34
NB=60
CALL WRS(JCH,IHDG,NB,IER)
IF(IER.NE.1)GO TO 34
RETURN
END
```

```
      OVERLAY RWRB
      SUBROUTINE RWR2(KEY,NVER,JER); FIND KEY RECORD OR OUTPUT TEXT

      COMMON IDUM(5),IBUF(128),IDATA(256),JDUM(5),ISW(2),ICH,JCH,ISTN(8),
     1 IKEY(5),KSAV,IDPH,ITPH,IPPH,JBUF(36)
      COMMON/AAA/IADR(2),NLIMS(2),IOF(10)
      COMMON/R2/IMSG(11),JMSG(6),NOT(6)
      DIMENSION KEY(1)
      DATA IMSG/"DIRECTORY FILE ERROR "/JMSG/"WRITE ERROR"/
      DATA NOT/"NOT STORED<15>"/
C
C     READ NEXT LINE OF DIRECTORY INFO
C
      NB=1
      CALL RDS(ICH,ITEMP,NB,IER); LF
      CALL ERROR(IER,IMSG)
      CALL RDL(ICH,IDATA,NB,IER)
      IF(IER.EQ.9)GO TO 90
      CALL ERROR(IER,IMSG)
      ITEMP=ISHFT(IDATA(1),-8)
      IF(ITEMP.EQ.3)GO TO 90; ETX MINUS PARITY BIT
      IF(ITEMP.EQ.53K.OR.ITEMP.EQ.55K)GO TO 10; + OR - MEANS A PRODUCT KEY
C
C     WRITE DIRECTORY LINE VERBATIM
C
      CALL WRL(JCH,IDATA,NB,IER)
      CALL ERROR(IER,JMSG)
      NB=1
      CALL WRS(JCH,5000K,NB,IER)
      CALL ERROR(IER,JMSG)
      JER=4; ILLEGAL FILENAME
      RETURN
C
C     FIND KEY RECORD OF CURRENT STATION
C
   10 KSAV=ITEMP
      CALL UNPACK(IDATA,80,IBUF)
      IBUF(11)=0
      CALL PACK(IBUF(2),10,KEY)
      CALL KSRCF(KEY,IDATA,IER)
      IF(IER.EQ.1)GO TO 20
      JER=13; FILE DOES NOT EXIST
   22 DO 201 I=12,27
      IF(IBUF(I).EQ.15K)GO TO 12
  201 CONTINUE
      I=27
   12 DO 202 J=I,27
  202 IBUF(J)=40K
      IF(JER.EQ.1)GO TO 30
      CALL UNPACK(NOT,11,IBUF(27))
      CALL PACK(IBUF(12),26,IBUF)
      CALL WRL(JCH,IBUF,NB,IER)
      CALL ERROR(IER,JMSG)
      NB=1
      CALL WRS(JCH,5000K,NB,IER); LF
```

```
      CALL ERROR(IER,JMSG)
      RETURN
C
C     SAVE STATION NAME AND GO TO DECODER
C
   20 JER=1
      NVER=IDATA(14).AND.177K
      GO TO 22
   30 CALL PACK(IBUF(12),16,ISTN)
      RETURN
C
C     END OF DIRECTORY, CLOSE THE PROGRAM
C
   90 NB=1
      CALL WRS(JCH,101400K,NB,IER); ETX
      CALL ERROR(IER,JMSG)
      CALL ERROR(IER,JMSG)
      CALL RESET
      CALL FSTOR(IOF,0,IER)
      IF(ISW(1).LT.0)CALL FORKP("ROUNDUP",IKEY,IER); A
      CALL EXIT
      END
```

```
      OVERLAY RWRC
      SUBROUTINE RWR3; REFORMAT WX, TEMP, WIND

      COMMON IDUM(5),IBUF(128),IDATA(256),JER,JDUM(4),ISW(2),ICH,JCH,ISTN(8),
     1 IKEY(5),KSAV,IDPH,ITPH,IPPH,JBUF(36)
      DIMENSION IWX(12),ICODE(3)
      EQUIVALENCE (IWX,IBUF(24))
      COMMON/RW/NWX(4,30),NWND(36)
      DATA NWND/"N N NENENENEE E E E E SESESESES S S S S SWSWSWSWW ",
     1  "W W W W NWNWNWNWN N N "/
      DATA NWX/"HAIL     TSTM     TSHWR    FRZGRAINFRZGDRZLSLEET    ",
     1  "RAINSNOWSNOWSHWRSHWR     SLEET    RAINSNOWSNOW     ",
     2  "RAIN     DRZL     BLWGSNOWBLWGDUSTFOG      DUST     ",
     3  "HAZE     SMOKE    CLDY     MOCLDY  PTCLDY  CLEAR    ",
     4  "UNKNOWN PTSUNNY MOSUNNY SUNNY    THUNDER FLURRIES"/
      IWX(11)=0; FOR INDEX OVERFLOW (OVERWRITES SLP)
      DO 201 I=1,72
  201 IDATA(I)=40K
      IDATA(50)=15K
      IDATA(58)=15K
      IDATA(66)=15K
      IDATA(72)=15K
      KSAV=50
      CALL UNPACK(ISTN,14,IDATA)
      IF(JER.EQ.1)GO TO 10
      IDATA(16)=116K; N
      IDATA(17)=117K; O
      IDATA(18)=124K; T
      IDATA(19)=40K
      IDATA(20)=101K; A
      IDATA(21)=126K; V
      IDATA(22)=102K; B
      IDATA(23)=114K; L
      IDATA(24)=15K
      RETURN
   10 IVSB=IBUF(22)
      ITT=IBUF(35)
      IDD=IBUF(38)
      IFF=IBUF(39)
      IF(IFF.GT.0)IFF=IFF*1.15+.51
      IGG=IBUF(40)
      IF(IGG.GT.0)IGG=IGG*1.15+.51
      IF(IVSB.EQ.-99)GO TO 85; NO VSB MEANS WX ERROR
      IF(IWX(10).EQ.40K)GO TO 85; NO WX REPORTED
C
C     SEARCH WX GROUP AND SET CORRESPONDING BITS
C
      MASK=0
      IF(LTR(101K,IWX,I))MASK=MASK.OR.100000K; A
      IF(LTR(124K,IWX,I))MASK=MASK.OR.40000K; T
      IF(LTR(132K,IWX,I))MASK=MASK.OR.20000K; Z
      IF(LTR(127K,IWX,I))MASK=MASK.OR.10000K; W
      IF(LTR(111K,IWX,I).EQ.0)GO TO 56
      IF(IWX(I+1).EQ.120K)MASK=MASK.OR.4000K; IP
   56 IF(LTR(123K,IWX,I).EQ.0)GO TO 52
```

```
      IF(I.EQ.1)GO TO 50
      IF(IWX(I-1).EQ.102K)GO TO 52
   50 MASK=MASK.OR.2000K; S (BUT NOT BS)
   52 IF(LTR(122K,IWX,I))MASK=MASK.OR.1000K; R
      IF(LTR(114K,IWX,I))MASK=MASK.OR.400K; L
C
C     DETERMINE PRESENT WX ON A PRIORITY BASIS
C
      M=1; HAIL
      IF((MASK.AND.100000K).NE.0)GO TO 90
      IF((MASK.AND.40000K).EQ.0)GO TO 54; NO THUNDER
      IF((MASK.AND.17400K).EQ.0)GO TO 54; NO PCPN
      M=2; TSTORM
      IF(LTR(55K,IWX,I))M=3; - FOR TSHWR
      GO TO 90
   54 M=4; FREEZING RAIN
      IF((MASK.AND.21000K).EQ.21000K)GO TO 90
      M=5; FREEZING DRIZZLE
      IF((MASK.AND.20400K).EQ.20400K)GO TO 90
      M=6; SLEET SHOWERS
      IF((MASK.AND.14000K).EQ.14000K)GO TO 90
      M=7; RAIN AND SNOW SHOWERS
      IF((MASK.AND.13000K).EQ.13000K)GO TO 90
      M=8; SNOW SHOWERS
      IF((MASK.AND.12000K).EQ.12000K)GO TO 90
      M=9; RAIN SHOWERS
      IF((MASK.AND.11000K).EQ.11000K)GO TO 90
      M=10; SLEET
      IF((MASK.AND.4000K).NE.0)GO TO 90
      M=11; RAIN AND SNOW
      IF((MASK.AND.3000K).EQ.3000K)GO TO 90
      M=12; SNOW
      IF((MASK.AND.2000K).NE.0)GO TO 90
      M=13; RAIN
      IF((MASK.AND.1000K).NE.0)GO TO 90
      M=14; DRIZZLE
      IF((MASK.AND.400K).NE.0)GO TO 90
      IF(LTR(102K,IWX,I).EQ.0)GO TO 58
      M=15; BLOWING SNOW
      IF(IWX(I+1).EQ.123K)GO TO 90
      M=16; BLOWING DUST
      IF(IWX(I+1).EQ.104K)GO TO 90
   58 VSBY=IVSB
      IF(VSBY.LT.0)VSBY=-VSBY/1000.
      IF(VSBY.GT.1)GO TO 96
      DO 206 I=1,10
      M=17; FOG
      IF(IWX(I).EQ.106K)GO TO 90; F
      M=18; DUST
      IF(IWX(I).EQ.104K)GO TO 90; D
      M=19; HAZE
      IF(IWX(I).EQ.110K)GO TO 90; H
      M=20; SMOKE
      IF(IWX(I).EQ.113K)GO TO 90; K
  206 CONTINUE
C
C     CHECK FOR OBSTRUCTION TO VISION IN REMARKS
C
   96 IF(IVSB.GT.3)GO TO 85
      DO 207 I=1,10
```

```
          M1=17
          IF(IWX(I).EQ.106K)GO TO 97
          M1=18
          IF(IWX(I).EQ.104K)GO TO 97
          M1=19
          IF(IWX(I).EQ.110K)GO TO 97
          M1=20
          IF(IWX(I).EQ.113K)GO TO 97
   207    CONTINUE
          GO TO 85
    97    CALL UNPACK(NWX(1,M1),8,IDATA(50))
          KSAV=58
C
C         OUTPUT CLOUDS IF THERE IS NO WX
C
    85    DO 211 I=1,3; NO OF CLOUD LAYERS
          K=I*4+6
          ICODE(I)=4
          IF(IBUF(K).EQ.117K.OR.IBUF(K).EQ.130K)GO TO 120; O OR X
          ICODE(I)=3
          IF(IBUF(K).EQ.102K)GO TO 120; B
          ICODE(I)=2
          IF(IBUF(K).EQ.123K)GO TO 120; S
          ICODE(I)=1
          IF(IBUF(K).EQ.103K)GO TO 120; C
          ICODE(I)=0
          GO TO 211
   120    IF(IBUF(K+1).NE.1.AND.IBUF(K+1).NE.51)GO TO 211
          ICODE(I)=ICODE(I)-1; THIN
          IF(IBUF(K).EQ.130K)ICODE(I)=1; -X = CLEAR
   211    CONTINUE
          ITEMP=0
          DO 203 I=1,3
          IF(ICODE(I).GT.ITEMP)ITEMP=ICODE(I)
   203    CONTINUE
          M=25-ITEMP; SKY COVER
C
C         MAKE LAST-MINUTE ADJUSTMENTS
C
    90    IF((ISW(2).AND.20000K).NE.0)GO TO 92; SKIP SUNNY
          IF(M.GE.22.AND.M.LE.24)M=M+4
    92    IF((M.EQ.7.OR.M.EQ.11).AND.IVSB.GE.7)M=30; FLURRIES
          CALL UNPACK(NWX(1,M),8,IDATA(16))
          IF((MASK.AND.40000K).EQ.0.OR.M.LT.4)GO TO 94
          CALL UNPACK(NWX(1,29),8,IDATA(50)); PUT THUNDER IN REMARKS
          KSAV=58
C
C         DECODE TEMPERATURE AND WIND
C
    94    IF(ITT.NE.-99)GO TO 20
          IDATA(26)=115K; M
          IDATA(27)=115K; M
          GO TO 30
    20    NEG=.FALSE.
          IF(ITT.GE.0)GO TO 22
          ITT=-ITT
          NEG=.TRUE.
    22    K=27
          IDATA(K)=MOD(ITT,10)+60K; RIGHT JUSTIFY
          K=K-1
```

```
      ITT=ITT/10
      IF(ITT.EQ.0)GO TO 24
      IDATA(K)=MOD(ITT,10)+60K
      K=K-1
      ITT=ITT/10
      IF(ITT.EQ.0)GO TO 24
      IDATA(K)=ITT+60K
      K=K-1
24    IF(NEG)IDATA(K)=102K; B
30    IF(IDD.GE.0.AND.IDD.LE.360)GO TO 40
32    IDATA(34)=115K; M
      IDATA(35)=111K; I
      IDATA(36)=123K; S
      IDATA(37)=107K; G
      IDATA(38)=40K
      GO TO 48
40    IF(IDD.GT.0)GO TO 42
      IDATA(34)=103K; C
      IDATA(35)=101K; A
      IDATA(36)=114K; L
      IDATA(37)=115K; M
      IDATA(38)=40K
      GO TO 48
42    IF(IFF.EQ.-99)GO TO 32
      CALL UNPACK(NWND(IDD/10),2,IDATA(34))
      K=36
      IF(IDATA(35).EQ.40K)K=35
      IF(IFF/10.EQ.0)GO TO 44
      IDATA(K)=MOD(IFF/10,10)+60K
      K=K+1
44    IDATA(K)=MOD(IFF,10)+60K
      K=K+1
      IF(IGG.EQ.-99)GO TO 48
      IDATA(K)=107K
      K=K+1
      IF(IGG/10.EQ.0)GO TO 46
      IDATA(K)=MOD(IGG/10,10)+60K
      K=K+1
46    IDATA(K)=MOD(IGG,10)+60K
48    RETURN
      END
```

```
      OVERLAY RWRD
      SUBROUTINE RWR4; COMPUTE RELATIVE HUMIDITY

     COMMON IDUM(5),IBUF(128),IDATA(256),JER,JDUM(4),ISW(2),ICH,JCH,
    1 ISTN(8),IKEY(5),KSAV,IDPH,ITPH,IPPH,JBUF(36)
      DIMENSION T(4)
      IDATA(72)=15K
      IF(IDATA(26).EQ.115K.OR.IDATA(16).EQ.116K)RETURN;T MISG OR STN NA?
      IF(IBUF(36).EQ.-99)GO TO 75;DP MISG?
      T(1)=IBUF(35)
      T(2)=IBUF(36)
      IF(T(2).GE.T(1))GO TO 70;DP.GE.TEMP?
      RL=ALOG(10.)
      A=ALOG(1013.25)/RL
      DO 201 I=1,2
      T(I)=5.*(T(I)-32.)/9.+273.16
      TEMP=373.16/T(I)
      A1=A-7.903*(TEMP-1)
      A1=A1+5.03*ALOG(TEMP)/RL
      A1=A1-1.382E-7*(10**(11.334*(1-T(I)/373.16))-1)
201   T(I+2)=A1+8.133E-3*(10**(-3.49*(TEMP-1))-1)
      TEMP=(T(4)-T(3))*RL
      RH=100*EXP(TEMP)+.51
      IRH=RH
     K=30
      GO TO 88
70     IDATA(29)=61K
     IDATA(30)=60K
     IDATA(31)=60K
     RETURN
75     IDATA(30)=115K; M
      IDATA(31)=115K; M
      RETURN
88    IF(IRH/10.EQ.0)GO TO  89
     IDATA(K)=MOD(IRH/10,10)+60K
     K=K+1
89    IDATA(K)=MOD(IRH,10)+60K
     RETURN
     END
```

```
      SUBROUTINE RWR5
C
C     COMPUTE AND OUTPUT WIND CHILL INDEX WHEN TEMP IS
C           35 OR LESS AND WIND SPEED IS 4 MPH OR GREATER.
C     OUTPUT DEW POINT IF GLOBAL SWITCH /D IS SET
C     OUTPUT BAROMETRIC PRESSURE
C
      COMMON IDUM(5),IBUF(128),IDATA(256),JER,JDUM(4),ISW(2),ICH,JCH,
     1 ISTN(8),IKEY(5),KSAV,IDPH,ITPH,IPPH,JBUF(36)
C
      IDONE=0    ;FLAG FOR DEWPOINT PROCESSED 0=NO
      IF(IBUF(35).EQ.-99)GO TO 115;TEMP MISG?
      IF(IDATA(34).EQ.115K.OR.IDATA(36).EQ.114K)GO TO 88;WND MISG OR CALM?
      IF(IBUF(35).GT.35.OR.IBUF(39).LT.4)GO TO 88;TEMP.GT.35 OR SPD.LT.4?
      V=IBUF(39)*1.15+.51
      T=IBUF(35)
      A=6.686*SQRT(V)-.447*V+10.45
      WCI=91.4+A*(.04538*T-4.148)
      IWC=WCI+.51
      NEG=.FALSE.
      IF(IWC.GE.0)GO TO 51
      IWC=-WCI+.5
      NEG=.TRUE.
   51 K=KSAV+4
      IDATA(KSAV)=127K ;W
      IDATA(KSAV+1)=103K ;C
      IDATA(KSAV+2)=111K ; I
      IDATA(KSAV+3)=40K ; SPACE
      KSAV=KSAV+8
   55 IF(IWC.LT.10)GO TO 66
      IF(IWC.LT.100)GO TO 64
      IDATA(K)=IWC/100+60K
      IWC=MOD(IWC,100)
      K=K+1
   64 IDATA(K)=IWC/10+60K
      K=K+1
   66 IDATA(K)=MOD(IWC,10)+60K
      IF(NEG)IDATA(K+1)=102K; B
   88 CONTINUE
      IF(IDONE.EQ.1)  GO TO 115
      IF((ISW(1).AND.10000K).EQ.0)GO TO 115; NOT D
      IF(IBUF(36).EQ.-99) GO TO 115
      IF(IBUF(36).GT.IBUF(35))GO TO 115;DP.GT.TEMP?
      IDATA(KSAV)=104K;  D
      IDATA(KSAV+1)=120K  ;P
      IDATA(KSAV+2)=40K;  SPACE
      K=KSAV+3
      IWC=IBUF(36)
      NEG=.FALSE.
      IDONE=1    ;DEWPOINT PROCESSED  1=YES
      IF(IWC.GE.0) GO TO 55
      IWC=-IWC
      NEG=.TRUE.
      GO TO 55
  115 IT1=IBUF(41)/1000
```

```
IDATA(43)=IT1+60K
IT1=IBUF(41)-(1000*IT1)
IT2=IT1
IT1=IT1/100
IDATA(44)=IT1+60K
IDATA(45)=56K
IT1=IT2-(100*IT1)
IT2=IT1
IT1=IT1/10
IDATA(46)=IT1+60K
IDATA(47)=(IT2-(10*IT1))+60K
 IPPH=IBUF(41); SAVE CURRENT ALSTG
 CALL PACK(IDATA,72,JBUF); SAVE OUTPUT LINE
 RETURN
 END
```

```
      SUBROUTINE RWR6; COMPUTE PRESSURE TENDENCY AND OUTPUT

      COMMON IDUM(5),IBUF(128),IDATA(256),JDUM(5),ISW(2),ICH,JCH,
     1  ISTN(8),IKEY(5),KSAV,IDPH,ITPH,IPPH,JBUF(36)

      IF(IPPH.LT.0)GO TO 10; NO CURRENT VALUE
      IF(IBUF(41).LT.0)GO TO 10; NO PREVIOUS VALUE
      IDP=IPPH-IBUF(41)
      ITEMP=123K; S
      IF(IDP.GT.0)ITEMP=122K; R
      IF(IDP.LT.0)ITEMP=106K; F
      JBUF(24)=(JBUF(24).AND.177400K)+ITEMP
   10 CALL WRL(JCH,JBUF,NB,IER)
   14 CALL ERROR(IER,"WRITE ERROR")
      NB=1
      CALL WRS(JCH,5000K,NB,IER)
      IF(IER.NE.1)GO TO 14
      RETURN
      END
```

```
 0001 JONES     MACRO REV 06.70          12:08:16 03/24/83
                      .TITL   JONES           ;RAIDER OF THE LOST SUBROUTINE
02                    .ENT    JONES
03                    .EXTN   MMHDR
04                    .EXTD   .FRET,.CPYL
05                ;
06                ;       SUBROUTINE JONES
07                ;
08                ;       MODIFY SUBROUTINE MMHDR TO USE STANDARD TIME YEAR-ROUND
09                ;       (AS IN INDIANA)
10                ;
11                ;       PROGRAMMER:  WARREN E. SUNKEL, WSFO TOPEKA
12
13                      .NREL
14 00000'000000         0
15                JONES:
16 00001'006002$        JSR     @.CPYL
17 00002'020404         LDA     0,J1
18 00003'030404         LDA     2,J2
19 00004'041114         STA     0,114,2
20 00005'006001$        JSR     @.FRET
21
22 00006'000507 J1:     507
23 00007'077777 J2:     MMHDR
24
25                      .END

**00000 TOTAL ERRORS, 00000 PASS 1 ERRORS
```

```
  0002 JONES

J1     000006'        1/17    1/22
J2     000007'        1/18    1/23
JONES  000001' EN     1/02    1/15
MMHDR  000007' XN     1/03    1/23
.CPYL  000002$ XD     1/04    1/16
.FRET  000001$ XD     1/04    1/20
```

```
 0001 LTR       MACRO REV 06.70          12:07:27 03/24/83
                        .TITL   LTR     ;SEARCH FOR A SPECIFIED LETTER
02                      .ENT    LTR
03                      .EXTD   .FRET,.CPYL
04                ;
05                ;
06                ;       FUNCTION LTR(ICHAR,IAR,I)
07                ;
08                ;       SEARCH THE UNPACKED 10-CHARACTER ARRAY IAR FOR THE CHARACTER ICHAR.
09                ;       IF FOUND, RETURN WITH LTR=.TRUE. AND I=ARRAY INDEX.
10                ;       IF NOT FOUND, RETURN WITH LTR=.FALSE.
11                ;
12                ;
13                      .NREL
14 00000'000004         4
15                LTR:
16 00001'006002$        JSR     @.CPYL          ;GET STACK LINKAGE
17 00002'021611         LDA     0,-167,3
18 00003'040430         STA     0,.LTR          ;ADDRESS OF FUNCTION VALUE
19 00004'023612         LDA     0,@-166,3               ;CHARACTER TO AC0
20 00005'031613         LDA     2,-165,3                ;ADDRESS OF ARRAY
21 00006'025614         LDA     1,-164,3
22 00007'044426         STA     1,.I            ;ADDRESS OF INDEX RETURN
23 00010'024426         LDA     1,D10
24 00011'044423         STA     1,COUNT
25                L1:
26 00012'025000         LDA     1,0,2
27 00013'106405         SUB     0,1,SNR
28 00014'000407         JMP     L2              ;EQUAL
29 00015'151400         INC     2,2
30 00016'014416         DSZ     COUNT
31 00017'000773         JMP     L1              ;LOOP
32 00020'102400         SUB     0,0             ;NOT FOUND
33 00021'042412         STA     0,@.LTR         ;FALSE
34 00022'006001$        JSR     @.FRET
35                L2:
36 00023'020413         LDA     0,D10           ;COMPUTE INDEX
37 00024'024410         LDA     1,COUNT
38 00025'122400         SUB     1,0
39 00026'101400         INC     0,0             ;BIAS FOR FORTRAN
40 00027'042406         STA     0,@.I
41 00030'102000         ADC     0,0             ;-1 FOR TRUE
42 00031'042402         STA     0,@.LTR
43 00032'006001$        JSR     @.FRET
44
45 00033'000000 .LTR:   0
46 00034'000000 COUNT:  0
47 00035'000000 .I:     0
48 00036'000012 D10:    10.
49                      .END

**00000 TOTAL ERRORS, 00000 PASS 1 ERRORS
```

0002 LTR

COUNT	000034'		1/24	1/30	1/37	1/46
D10	000036'		1/23	1/36	1/48	
L1	000012'		1/25	1/31		
L2	000023'		1/28	1/35		
LTR	000001'	EN	1/02	1/15		
.CPYL	000002$	XD	1/03	1/16		
.FRET	000001$	XD	1/03	1/34	1/43	
.I	000035'		1/22	1/40	1/47	
.LTR	000033'		1/18	1/33	1/42	1/45

NOAA SCIENTIFIC AND TECHNICAL PUBLICATIONS

The National Oceanic and Atmospheric Administration was established as part of the Department of Commerce on October 3, 1970. The mission responsibilities of NOAA are to assess the socioeconomic impact of natural and technological changes in the environment and to monitor and predict the state of the solid Earth, the oceans and their living resources, the atmosphere, and the space environment of the Earth.

The major components of NOAA regularly produce various types of scientific and technical information in the following kinds of publications:

PROFESSIONAL PAPERS — Important definitive research results, major techniques, and special investigations.

CONTRACT AND GRANT REPORTS — Reports prepared by contractors or grantees under NOAA sponsorship.

ATLAS — Presentation of analyzed data generally in the form of maps showing distribution of rainfall, chemical and physical conditions of oceans and atmosphere, distribution of fishes and marine mammals, ionospheric conditions, etc.

TECHNICAL SERVICE PUBLICATIONS — Reports containing data, observations, instructions, etc. A partial listing includes data serials; prediction and outlook periodicals; technical manuals, training papers, planning reports, and information serials; and miscellaneous technical publications.

TECHNICAL REPORTS — Journal quality with extensive details, mathematical developments, or data listings.

TECHNICAL MEMORANDUMS — Reports of preliminary, partial, or negative research or technology results, interim instructions, and the like.

Information on availability of NOAA publications can be obtained from:

ENVIRONMENTAL SCIENCE INFORMATION CENTER
ENVIRONMENTAL DATA AND INFORMATION SERVICE
NATIONAL OCEANIC AND ATMOSPHERIC ADMINISTRATION
U.S. DEPARTMENT OF COMMERCE

Rockville, MD 20852