

QC  
874.3  
.U63  
no.3

NOAA Eastern Region Computer Programs  
and Problems NWS ERCP - No. 3



---

PUPPY (AFOS HYDROLOGIC DATA REPORTING PROGRAM)

Scientific Services Division  
Eastern Region Headquarters  
December 1981

---

**U.S. DEPARTMENT OF  
COMMERCE**

/ National Oceanic and  
Atmospheric Administration

/ National Weather  
Service



NOAA TECHNICAL MEMORANDUM  
National Weather Service, Eastern Region Computer Programs and Problems

The Eastern Region Computer Programs and Problems (ERCP) series is a subset of the Eastern Region Technical Memorandum series. It will serve as the vehicle for the transfer of information about fully documented AFOS application programs. The format of ERCP - No. 1 will serve as the model for future issuances in this series.

- 1 An AFOS Version of the Flash Flood Checklist. Cynthia M. Scott, March 1981. (PB81 211252).
- 2 An AFOS Applications Program to Compute Three-Hourly Stream Stages. Alan P. Blackburn, September 1981. (PB82 156886).



82 00982

H  
QC  
874.3  
U63  
no. 3

NOAA Eastern Region Computer Programs and Problems - No. 3

" PUPPY (AFOS hydrologic data reporting program)

Daniel P. Provost  
NWS River Forecast Center  
Cincinnati, Ohio

Scientific Services Division  
Eastern Region Headquarters  
December 1981

CENTRAL  
LIBRARY  
  
APR 26 1982  
  
N.O.A.A.  
U. S. Dept. of Commerce

UNITED STATES  
DEPARTMENT OF COMMERCE  
Malcolm Baldrige, Secretary

National Oceanic and  
Atmospheric Administration  
John V. Byrne, Administrator

National Weather  
Service  
Richard E. Hallgren, Director



82 00982



TABLE OF CONTENTS

	<u>Page</u>
I. GENERAL INFORMATION . . . . .	1
II. APPLICATION . . . . .	1
III. PROCEDURES . . . . .	9
APPENDIX 1 . . . . .	A1-1
APPENDIX 2 . . . . .	A2-1



# PUPPY (AFOS HYDROLOGIC DATA REPORTING PROGRAM)

Daniel P. Provost  
NWS River Forecast Center, Cincinnati, Ohio

## I. General Information

### A. Summary:

PUPPY is designed to automate the handling of hydrologic data in the interim before the Office of Hydrology implements the FHYDO program. Once the data (precipitation observations, river stages, etc.) is entered on AFOS via PUPPY, it can be passed to other computer systems without manual intervention. PUPPY also produces hydrologic weather wire products and plotfiles for transformation into graphics.

### B. Environment:

PUPPY runs on the Data General ECLIPSE S/230 and interacts with the AFOS data base. The language used is Data General's FORTRAN IV.

## II. Application

### A. Complete Program Description:

This software package is divided into three programs. The first program which must be used is called NETWORK.SV. This interactive program provides the user with a menu. This menu allows the user to select options such as adding, deleting, or changing a station. Also, the user may choose to redefine the map window (portion of map background to be plotted) or move a station up or down on the AFOS GDM map position. Since the program is interactive, errors can be aborted by continuing to enter data. The program in most cases will give the user a chance to review the entries made and abort if desired.

Before attempting to run this program please review this documentation. You will have to gather specific information before you begin building this network file.

Here is a brief description of the NETWORK program and its subroutines:

NETWORK.FR: The main program displays a menu and loads the overlay area depending on user menu selection. On exit from the program the NETBAS.SV program is loaded via chaining if the user is a WSFO requiring a downstream file for the PUPPY program.

NET1.FR: Subroutine DESCRIBE - This subroutine merely lists the files of the program and a brief description of their contents. (Overlay OV1)

NET2.FR: Subroutine ADDITION - Allows the user to add a station to the network (the items required are listed in the Procedures section). (Overlay OV2)



- NET3.FR: Subroutine DLTDATA - Allows the user to delete a station from the network. (Overlay OV3)
- NET4.FR: Subroutine CHANGE - Allows the user to change information about a station such as latitude, longitude, etc. Again as with all other portions of NETWORK.SV, the user is given choices by the program. (Overlay OV4)
- NET5.FR: Subroutine ADJUST - Allows the user to change the pixel values of any station. The Lat-Lon entered generates the pixel values. However, if two stations are too close, the user can move these stations. (Overlay OV5)
- NET6.FR: Subroutine NEWMAP - Allows the user to redefine the map window previously selected. This could be useful for new map projections. (Polar stereographic only!)
- NET7.FR: Subroutine LOADNT - Loads the call signs of the entire network into an array for binary search purposes.
- NET8.FR: Subroutine PIXELS - Generates polar stereographic pixels from a Western Region program (provided by Don Laurine SLC RFC).
- NET9.FR: Subroutine BINSR - Is a binary search subroutine converted from a subroutine written by Tom McPhillips CIN RFC.

The second portion of the package is the program chained from NETWORK called NETBAS.SV. This program is initially chained because the additions and deletions made with the network program are monitored into a file called MONITOR.DT. This file becomes the input to NETBAS.SV. The user is prompted with the call sign and the request to store this call sign into a station number. The current downstream file (if any) is stored in AFOS. Before any of the questions are asked, however, NETBAS.SV looks for a file called PUP.FL. This contains all the CCCNNNXXXs where you wish to store your products. If NETBAS does not find this file it will ask you for a CCCNNNXXX to store a skeleton PUP.FL (Figure 1). You are then instructed to edit this CCCNNNXXX and reexecute NETBAS. (The purpose of the use of AFOS for file storage along with RDOS is to facilitate listing any of your files. Even though NETBAS and PUPPY use the RDOS version listing these RDOS files on the Dasher can be avoided.) The same procedure is repeated for the basin file, PUP.BX (Figure 2).

The following is a brief description of the NETBAS program:

- NETBAS.FR: The main program allows the user to determine a downstream order to the hydrologic service area. This is an example of how the files will be set up. Your basin file will contain up to 25 basin names, and your downstream file will contain stations in sequence corresponding to your basin file. If there are any changes to be made to the basin file you must edit the CCCNNNXXX containing these names and follow the instructions in that file. NETBAS helps you do this step by step, plus it allows you to suppress weather wire fields.



Figure 1. PUP.FL Skeleton.

```

***** PUPPY CONTROL FILE *****
CODE 77=START DO NOT CHANGE!!!
CODE 78=WEATHER WIRE MESSAGE MAX 9 FILES
79=PLT FORMAT(NATIONAL) MAX 1 FILE
80=BASIN LISTING IN AFOS MAX 1 FILE
81=DOWNSTREAM STATION LISTING IN AFOS MAX 1 FILE
82=CONTROL FILE LISTING IN AFOS MAX 1 FILE
83=STRANGER FILE (IE. NO CALL SIGN AVAIL.)
84=RFC MESSAGE (ALL DATA) MAX 1 FILE
86=RAWARC HEADING(IE. SRUS23 RWRA)
87=PLT FORMAT(LOCAL)
99=END DO NOT CHANGE!!!
THE NNN CATEGORIES LISTED BELOW ARE ONLY RECOMMENDATIONS
EDIT THIS FILE IN EXACT COLUMNS THEN EXECUTE THE FOLLOWING COMMANDS
SAVE:CCCNNNXXX DPX:PUPPY.TS
RUN:DPX:NETBAS
77 CONTROL FILES:
78 CCCRR1XXX 99,00,00,00,00,00,00,00,00,00,00,00,00,00,00
78 CCCRR2XXX 01,02,03,04
79 CCCPLTXXX
80 CCCWRKXXX
81 CCCWRKXXX
82 CCCWRKXXX
83 CCCNNNXXX
86 SRUS RWRA
87 CCCPLTXXX
99 END

```

Figure 2. PUB.BS Skeleton.

```

BASIN LISTING FILE
TO GENERATE OR CHANGE THIS FILE, EDIT THIS PRODUCT,
ENTER YOUR BASIN NAMES AND WSFO ID. DO NOT SHIFT
COLUMNS AND DO NOT CHANGE ANY 88,89 OR 99 CODES!!!! IF YOU WISH
TO ADD A BASIN SIMPLY INSERT A LINE BUT LEAVE THE NUMBER OF THE
BASIN BLANK. THE PROGRAM WILL RENUMBER BOTH THIS FILE AND
YOUR STATION LISTING FILE IF ANY. DO NOT EDIT ABOVE THIS LINE!!!
ALSO DO NOT DELETE ANY NUMBERS OTHER THAN THOSE NOT NEEDED.
-----
88 BASIN FILE.
89 WSFO IDENTIFICATION (MAX 60 LETTERS)
01 BASIN NAME
02 BASIN NAME
03 BASIN NAME
04 BASIN NAME
05 BASIN NAME
99 END
R

```



- NETBAS1.FR: Subroutine STATUS - Determines whether your file control table exists and your basin file exists. If not, NETBAS gives you instructions on how to do this.
- NETBAS2.FR: Subroutine CORREC - Determines if there is a PUPPY.TS correction file present. If the user makes any changes to the file control table or the basin file stored in AFOS, that CCCNNNXXX is stored in PUPPY.TS, and the NETBAS program is executed. The corrections made in PUPPY.TS will also be made in the RDOS version of those same files.

The last portion of this software package is the PUPPY program. This program consists of 14 subroutines. Here is a brief description of the program and the subroutines:

- PUPPY.FR: The main program loads the overlay area and calls all necessary subroutines to execute loading and decoding the incoming data.
- PUP1.FR: Subroutine ADMCK - Checks the input file called PUPPY.IN for the correct data and determines the correct date which should be yesterday's data in PUP.YS. The program also checks the program option to see if the option agrees with the date in the data files already on disk (if any).
- PUP2.FR: Subroutine BINSR - Performs a binary search whenever the network location of a station is required. This subroutine is used extensively along with the call FSEEK (FORTRAN Random Access).
- PUP3.FR: Subroutine LOADST - Loads the downstream file station listing and stores its file location in ILCTN(ILOC,2) array.
- PUP4.FR: Subroutine LOADNT - Loads the data and determines the valid call sign or the various other call sign options which can be used.
- PUP5.FR: Subroutine LOADYS - Loads yesterday's data location from the file PUP.YS, into array ILCTN(ILOC,3); used in computing 24-hour river changes.
- PUP6.FR: Subroutine FORMT - Determines the record format by counting characters and spaces; also determines whether the input is free (untagged) or tagged by sensing data separators , / =
- PUP7.FR: Subroutine LOADNT - Loads the NETWORK.DT file into array INET(1500,3)-3A2 format; this array is used extensively for binary searches and contains a maximum of 1500 stations.
- PUP8.FR: Writes to a stranger file from any entry with the call sign: XXXXX
- PUP9.FR: Subroutine ENDREC - Rereads the previous records according to the format determined by subroutine FORMT; also reads the default element numbers from PUP.ST if the data was untagged.



- PUP10.FR: Subroutine PLTFM - Reads the PUP.DT file containing the data. The location was stored in ILCTN(ILOC,1). This subroutine generates 2 PLT files: one with national pixels and one with local pixels obtained from NETWORK.DT.
- PUP11.FR: Subroutine WEATHR - Reads the PUP.DT file and enters the latest station data by basin into Weather Wire style files. Basin order is in PUP.BS; 99=all basins or maximum 15 separate basin numbers in random order. Element fields may be suppressed at local option by running the NETBAS program.
- PUP12.FR: Subroutine JULN - Determines the Julian date.
- PUP13.FR: Subroutine RFC - Generates a message containing all data which is sent to the RFC.
- PUP16.FR: Subroutine WXHDG - Determines the Weather Wire heading required based on options selected by the user when running the NETBAS program. Snow and water equivalent are suppressed if no data is reported (i.e., summer).
- TRNS.FR: Subroutine PUP - Is a modification of subroutine FSTORE (Programming Note 90) and allows RDOS files to be stored in a CCCNNNXXX. This version will wait up to 3 minutes at 2-second intervals for AFOS to answer its request to store, otherwise the program will stop.
- PUPPY.OL Overlay file containing subroutines PUP7, PUP1, PUP3, PUP4, PUP10, PUP13.

Before running the PUPPY program the following files must have been generated:

NETWORK.DT	Alphabetical listing
PUP.ST	Downstream and element default listing
PUP.FL	File control table (all CCCNNNXXXs)
PUP.BS	Basin name listing

The instructions for generating these files are in the procedures section.

It is assumed that the PUPPY pre-format is used for all data entry into an AFOS product of the user's choice. Once the data is entered, the file is saved in RDOS as PUPPY.IN (SAVE:CCNXXX DPX:PUPPY.IN). If PUPPY.IN does not exist, or PUPPY.IN contains the wrong date, the program will abort and tell the user the cause. The program reads your input and writes your data into the PUP.DT file, while remembering its PUP.DT file location for later use. PUPPY will also read PUP.YS and store yesterday's file location (if any) for later use in computing 24-hour river changes.

PUPPY will generate a stranger file if any data is reported with the call sign as XXXXX. This is used to send a special file to the RFC of any miscellaneous public reports with no valid call sign.

PUPPY then generates 2 files in PLT format. These files are similar to the NMCPLTOA file containing surface weather observations transmitted by NMC. The first file generated contains national pixels



which match the NMCGPHB02 map background. The other file contains local pixels which match a local map of your choice (possibly containing river and county outlines). Only the national file needs to be transmitted on the RDC for RFCs and WSFOs.

The next step is for PUPPY to generate your Weather Wire files. You may generate multiple Weather Wire files, each containing up to 15 different basins in any order you selected or all basins in one file by using 99 as your basin routing code. Example: assume you have 20 basins in your hydrologic service area and you decide to generate 4 different Weather Wire messages. In message 1 you want basins 5, 4, 3, 2 and 1; in message 2 you want basins 20, 17, 16, 19 and 18; in message 3 you want basin B; and in message 4 you want all basins (99); in your AFOS file control table code 78 designates a Weather Wire file; therefore, you enter 4 code 78 lines containing AFOS keys (i.e., RR1 RR2 RR3 RR4 - totally optional). Then you enter your basin numbers using the format displayed on AFOS. Do not misalign the basin numbers, and remember that the more files you generate the longer the program execution time. When executing the NETBAS program you may suppress certain element fields from your Weather Wire products.

The last step for PUPPY is to generate an RFC file. The RFC will need to receive all data in downstream order in a readable format compared to the PLT formatted file which is used for plotting purposes. All data are transmitted including call sign and comments to the RFC and Weather Wire (RRRRR is call sign for comment line to RFC; CCCCC is call sign for comment to Weather Wire and RFC). Since you have the option of suppressing fields in your Weather Wire files, these files may not be suitable for the RFC.

#### B. Machine Requirements

To build, maintain or update the hydrologic database using NETWORK and NETBAS uses 15K and 13K, respectively. PUPPY requires 24K.

PUPPY.SV and .OL require 153 blocks of disk space. The size of the auxiliary and output files will vary with the number of stations and quantity of data. NETWORK.SV and .OL and NETBAS.SV take up an additional 131 blocks, but these programs can be stored on a floppy and only moved to a hard disk when database operations are planned. PUPPY should run from a hard disk. (This allows more reliable interfacing with AFOS.)

NETWORK and NETBAS are interactive, so their run times will vary. PUPPY runs in 4 to 5 minutes at RFCs and WSFOs (RFCs slightly longer than WSFOs).

The programs require approximately 12 channels, but 17 have been allotted to prevent any problems.



### C. Structure of Software

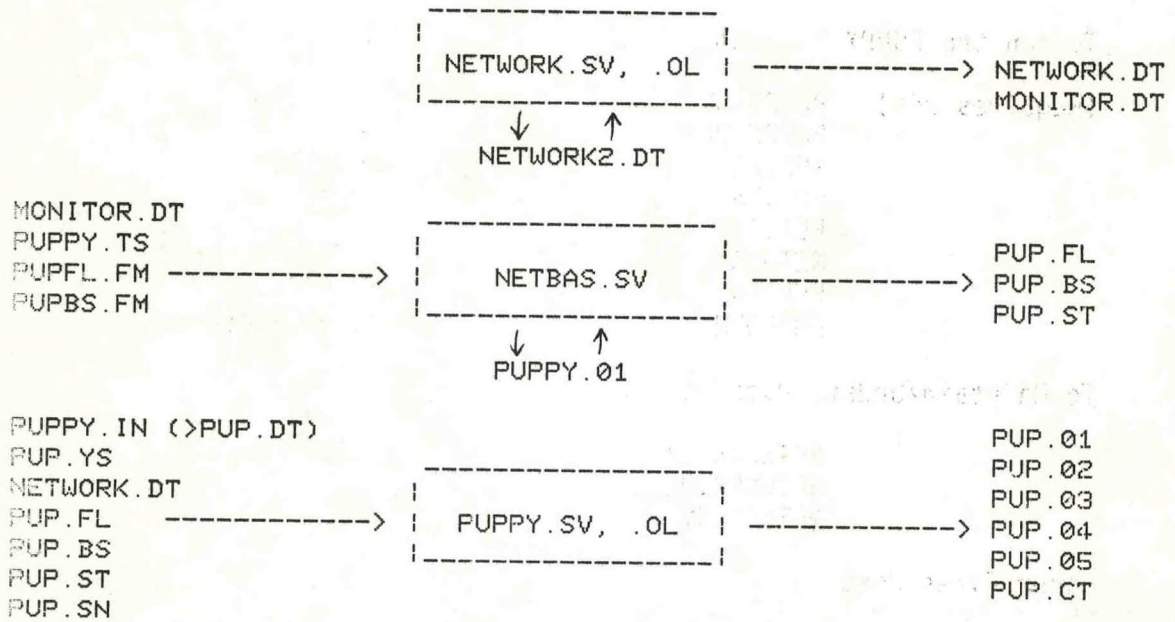


Figure 3. Basic Software Structure



## D. Files Needed

### To Build Hydro Database from Scratch:

(Requires 15K)	NETWORK.SV	
	NETWORK.OL	
(Requires 13K)	NETBAS.SV	
	PUPBS.FM	Skeleton Basin File
	PUPFL.FM	Skeleton File Control Table

### Above Programs will Generate:

PUP.FL	File Control Table
PUP.BS	Basin File
PUP.ST	Downstream Network Listing
NETWORK.DT	Alphabetical Network Listing
MONITOR.DT	Input for NETBAS Program

### To run the PUPPY Program:

(Requires 24K)	PUPPY.SV	
	PUPPY.OL	
	PUP.FL	Previously Described
	PUP.BS	" "
	PUP.ST	" "
	NETWORK.DT	" "
	PUP.SN	Synoptic Weather File
	PUPFORM	Stored in AFOS as Preformat

### To Maintain/Update Database:

NETWORK.SV  
NETWORK.OL  
NETBAS.SV

### Other Files Used:

FORMAT.DT	Runtime Formatting Work File
PUPPY.TS	Transaction File for NETBAS.SV
PUPPY.01	Work File for NETBAS.SV
NETWORK2.DT	NETWORK Work File
PUPPY.IN	PUPPY Input Data File
PUP.YS	Yesterday's PUP.DT
PUP.CT	PUPPY Comment Storage File
PUP.01	RDOS Stranger File
PUP.02	RDOS National PLT File
PUP.03	RDOS Local PLT File
PUP.04	RDOS Weather Wire File
PUP.05	RDOS RFC File



### III. Procedures

#### A. Creating the hydrologic database

##### Steps:

1. Obtain your local map window (Degrees Decimal).
2. Obtain or create a list of major basin names in your hydrologic service area (maximum 25 basins; maximum 60 character names-- see Figure 2).
3. Obtain or create a list of all stations which report data within each of the above basins. This list should be in downstream order for each basin.

##### (a) Each station should have the following items:

Call sign (maximum 5 characters)  
Station name (maximum 20 characters, 1-12 for Weather Wire)  
Latitude and longitude (degrees, minutes)  
AFOS zoom ratio (0 for 1st order; 1 substations)  
Basin number corresponding to the step 1 basin number  
Flood stage  
Station elevation  
Maximum number of elements observed at this station

- (b) Obtain the code number of each element from the PUPPY element list provided (Figure 4). Determine the reporting order for these elements. This order will become the default reporting order. Example: Station Z reports 2 (precip) and 8 (stage) on a daily basis. If you enter the maximum number of elements as 2 and enter the element numbers 2, 8 in the downstream file, the data entries in PUPPY.IN can appear as:

	Z	1.05	12.4	(untagged)
or	Z	2=1.05	8=12.4	(tagged)

Any additional data received from Station Z would require that all the data be tagged. If tagged, the order you enter the data is irrelevant. If untagged, the PUPPY program will access this downstream file you are building now to determine the tags for each element.

4. Obtain from your AFOS focal point 3 available reference keys (CCCREXXXX) which can be used to store your file control table, basin names and your downstream file. More of these AFOS CCCNNXXXXs will be needed later for operational use to be determined by your office requirement and your RFC (see Figure 1).



Figure 4: PUPPY Program Element List

Element Number	Element Type
1	Reserved (Station Circle)
2	Precipitation
3	Unassigned
4	Previous 7 AM precipitation
5	Present weather (Synoptic Codes 1-99)
6	New Snow
7	Stage (Tailwater)
8	Stage (Headwater)
9	Flow
10	Dam gates
11	Water temperature
12	Current air temperature
13	Maximum air temperature
14	Minimum air temperature
15	Snow depth
16	Water equivalent
17	Flow 1 PM
18	Flow 7 PM
19	Flow 1 AM
20	Flow 7 AM
21	Stage (Tailwater Previous 7 PM)
22	Stage (Headwater Previous 7 PM)
23	Stage (Tailwater Previous 7 AM)
24	Stage (Headwater Previous 7 AM)
25-29	Unassigned



5. When you have obtained all the information in steps 1-4, you may begin by executing NETWORK.SV.

- (a) Add a station (NETWORK will sort alphabetically).
- (b) Enter your map window.
- (c) Enter all your data (program updates disk after each 20 stations)
- (d) NETBAS will check for PUP.EL and PUP.BS.. These will not exist, but NETBAS will store a skeleton version in AFOS for you. Follow its instructions.
- (e) After entering your station downstream file, you must refer to your new station numbered list in AFOS after each 20 stations. (AFOS file renumbered)
- (f) When completed with NETBAS, backup the following files on a floppy disk: all save and overlay files; PUPBS.FM; PUPFL.FM; PUP.FL, PUP.BS, PUP.ST and their AFOS versions; PUP.SN; NETWORK.DT; PUPFORM.DT.
- (g) Store the PUPPY pre-format supplied as PUPFORM.DT into any CCCMCPXXX.
- (h) You are now ready to begin running the PUPPY program.

## B. Running PUPPY

### 1. Input

Hydro data is entered using the PUPPY preformat at an ADM. First the GMT time and date, and the local time, day and date (for Weather Wire) must be entered. Then one of the program options listed below must be selected and entered:

- 1 Indicates a new day for 12Z data and the first program execution.
- 2 Indicates additional 12Z data with the output containing all previous data entered for 12Z. Also the words ...additional data... are transmitted with the message headings.
- 3 Indicates additional 12Z data with the output containing only the most recent data entries for that execution. The words ...additional data... are also included with the message headings.
- 4 Indicates off-time data such as 18Z, 00Z, 06Z which need to be transmitted. Off-time data can only be processed similar to option 3 without appending previous transmissions.
- 5 Data correction. This option inserts the words ...corrected data... and will process only 12Z errors. Off-time errors must be processed as option 4 using AFOS message-comp edit to insert ...corrected data... manually.



Each row of data entered must have a valid location identifier in columns 2 through 6. (Column 1 = [ and column 7 = ] in the preformat.) These are listed below:

- (-----) Any valid station identifier known to the WSFO and the RFC with a maximum of 5 characters (preferably NWS Comms Handbook #5).
- (-----) Blank block indicates a data continuation line and must follow a valid call sign. More than one extra line may be added if necessary, but a blank line with no data should not be added.
- RRRRR This call sign is to be used when a comment is to be sent along with the data to the RFC only! This comment line is not intended to be disseminated to the public, i.e., a comment reflecting the possible discrepancy of the report. This option must follow a valid call sign. If selected, the text on the remainder of that line follows the data in the RFC file.
- CCCCC This call sign is similar to the RRRRR entry and is to be used when a comment is to be sent to both the public Weather Wire files and the RFC files.(i.e., record stage).
- XXXXX This call sign is used to report data or comments from stations with no valid identifier. When PUPPY detects this call sign, the remaining text on that line is written to the stranger file. It need not follow a valid station identifier and as many XXXXX lines can appear in a row thus allowing a paragraph to be written to the stranger file.
- IIIII This call sign indicates the whole line is ignored.
- NNNN Designates end of data.

The data in each row may be entered in either tagged or untagged (free) format. (Note: 99.99 indicates missing data, while 88.88 indicates trace.)

Tagged: In reporting tagged data you select the element number from the PUPPY program element list and you enter that number followed by a separator and the observed value. Valid separators are , / =  
Example: Cincinnati has a data value of 1.26 inches of precipitation and 5 inches of snow depth. You must enter the station identifier (CVG) after entering message-comp. From the element list we obtain the number 2 for precipitation and the number 15 for snow. This may be coded as follows:

CVG 2=1.26 15,5 or CVG 2/ 1.26 15= 5

You may have spaces between the separator and the data, but you must have at least one space between the data and the next element number:

CVG 2=1.2615/5 is illegal; a space must be after 1.26



The order of the data is irrelevant when tagged:

CVG 2/1.26 15,5 or CVG 15=5 2/1.26

If you have too much data for one line, leave the next line identifier blank and continue tagging data. The end of the station data is the next nonblank ID.

Untagged: You have previously defined default reporting elements in your downstream file. This feature now becomes used by the PUPPY program if you elect to report data in untagged mode. Whenever the PUPPY program detects any valid separator it uses the tagged system to route your data. If no separator is found, the default system is used. This allows the user to free-format data without tags. Example: as with the previous example, we have 2 observations to report (precip and snow). In defining your downstream file you knew on a routine morning CVG would report these two values. Therefore, you entered maximum values reported as 2 and element types as 2,15. Since these became your default values you may enter:

CVG 1.26 5 PUPPY will tag this data (2=1.26 15=5)

CVG 5 1.26 is invalid; default is (2=5 15=1.26)

If you had 3 observations from station CVG to report, i.e., stage 26.5:

CVG 1.26 5 26.5 would be invalid because the default for the 3rd number is 0 since only 2,15 were defined

Whenever extra data is to be reported all data must be tagged:

CVG 2=1.26 15/5 8/26.5

If only 1st observation is available you may enter:

CVG 1.26 first default is 2; remaining are unused

If only 2nd observation is available you must enter:

CVG 99.99 5 CVG precip would be missing

CVG 5 would be incorrect; default is 2=5

Figure 5 shows the PUPPY preformat. Once you have filled it in (Figure 6) and stored the product, execute on AFOS the command

SAVE:CCCNXX DPX:PUPPY.IN



Figure 3 is an example of the input file PUPPY.IN. Once the input file has been saved, execute

```
RUN:DPX:PUPPY
```

to initiate the PUPPY program.

## 2. Output

PUPPY stores its output in the keys that you specified in PUPPY.TS (Figure 1). After PUPPY stores a product the alert light will flash and the product name will appear on your screen. You may then issue the commands:

```
XMIT:CCCNNNXXX      For RDC transmission
```

or

```
ACOMMS:XMIT L CCCNNNXXX  For Async transmission
```

Plotting the PUPPY data is a function of the Hydro version of the PMOD program and will be treated separately.







Figure 6. Filled-in preformat for sample input.

```

CLEWRKHYD      EWOUS00 KCLE 291500
IIIII / PUPPY PROGRAM PRE-FORMAT / CALL SIGNS ARE 5 LTR ID OR:
IIIII XXXXX=STRANGER STATION ; IIIII=IGNORED LINE
IIIII RRRRR=COMMENT LINE TO RFC ; CCCCC=COMMENT LINE(WX WIRE+RFC)
IIIII =BLANK IS ADDITIONAL DATA ; NNNN =END OF ALL DATA
IIIII TIME<OF OB> DATE GMT / LOCAL TIME LOCAL DATA / PROGRAM OPTION
IIIII 1200 Z 12 / 29 / 81 7AM EST TUE 12 / 29 / 81 1
CROH 00.08 05.06
CHOH 99.99 04.71
RRRRR PRECIP QUESTIONABLE FROM PIKETON
PKOH 01.55 06.76
BOOH 00.00 02.56
CMHOH 00.10 689.3
HUROH 00.08 877.99
DDOH 00.19 910.09 881.08
ACDOH 99.99 885.50 5.61
DCDOH 00.09 796.85 770.58
PCOH 00.03 787.93 3.01
TJE 00.11 721.17 681.87
ENOH 99.99 01.81
ATOH 00.14 99.99 06.76
MILOH 00.12 06.44 06.4
EFDOH 00.08 730.81 616.58
CJBOH 00.14 1009.15 18.00 2
SPGOH 00.16 02.21 2
HUFMN 03.96
DAYOH 00.10 726.51
CMDOH 99.99
HAM 99.99 99.99 59.59
CCDOH 00.13 846.25
DFOH 99.99 99.99 03.25
FDY 00.16 02.30 6
WULOH 05.44
NUOH 03.87
PMH 00.00 21.14 23.70
MLDH 99.99 99.99 99.99
CUG 00.10 99.99
FRNOH 00.05 32.95
MRAOH 00.22 18.9 18.4
YNG 00.09 03.19 1
LEVOH 04.88
OH01 03.11
OH02 04.25
OH03 02.68
OH04 00.20
OH05 00.20
OH06 00.10
OH07 00.20
OH08 00.20
OH09 99.99
OH11 00.10
OH12 04.97
MLPC7 00.10
WCOH 00.00
LNOH 00.10
AKOH 05.06
MATOH 99.99 01.72 1
EATOH 00.20 5 1.00
CEL01 00.34 6

```



Note: The PUPPY program interface with AFOS is a modification of subroutines in AFOS Programming Note 90. An added feature allows PUPPY to wait for AFOS to respond for 3 minutes at 2-second intervals with 1-, 2-, and 3-minute Dasher PUPPY warnings. If any problem occurs, restart AFOS and run PUPPY again.

Figures 7 through 10 are examples of PUPPY's output. Appendix 1 contains explanations of PUPPY's console messages.

C. Cautions/Restrictions

This software package was designed in 3 components interdependent on each other. Any changes, however slight, could seriously affect the hydrologic network database or other vital functions of the program. Therefore, the user assumes full responsibility for the complete software package if any changes are made on site.

Furthermore, the use of any text editor is not recommended for any file; however, the use of AFOS message-composition remains a useful tool for editing final AFOS products.

D. Source Files

Source files appear in Appendix 2.



Figure 7. Sample plotfile with national pixels.

CLEPLT001 ECLEPLT0010010200020481536285014250097501688142912811423  
 1224,0275,10000Z,12Z 12/29/81;  
 1224,0259,10000Z,001 HYDROLOGIC OBSERVATIONS ;  
 1385, 675,200006,,ACDOH,8,99.99,,,,,5.61,885.50,;  
 1400, 710,100006,,AKOH,8,,,,,,5.06,;  
 1408, 700,200006,,ATIC6,8,0.06,,,,,894.04,922.83,;  
 1405, 667,000006,,ATOH,8,0.14,,,,,6.76,99.99,;  
 1416, 689,000006,,BARF6,8,0.08,;  
 1402, 701,200006,,BCSC5,8,0.08,,,,,929.96,949.34,;  
 1387, 658,200006,,BOOH,8,0.00,,,,,2.56,;  
 1404, 702,200006,,BOSC6,8,0.10,,,,,889.70,900.58,;  
 1402, 708,000006,,CAK,8,0.09,,,,,1.00,;  
 1408, 687,100006,,CAME5,8,2.74,;  
 1371, 661,100006,,CCDOH,8,0.13,,,,,846.25,;  
 1351, 678,100006,,CEL01,8,0.34,,,,,6.00,;  
 1390, 661,000006,,CHOH,8,99.99,,,,,4.71,;  
 1368, 668,200006,,CJBOH,8,0.14,,,,,18.00,1009.15,,,,,2.00,;  
 1411, 695,200006,,CLBD6,8,0.02,,,,,856.55,893.07,;  
 1389, 716,000006,,CLE,8,0.29,,,,,4.00,;  
 1387, 697,200006,,CMBC4,8,0.13,,,,,985.04,997.78,;  
 1363, 657,200006,,CMDOH,8,,,,,99.99,;  
 1384, 673,000006,,CMHOH,8,0.10,,,,,689.30,;  
 1401, 690,200006,,COMD5,8,0.10,,,,,9.12,;  
 1388, 667,200006,,CROH,8,0.08,,,,,5.06,;  
 1364, 645,000006,,CUG,8,0.10,,,,,99.99,;  
 1363, 665,000006,,DAY,8,0.18,,,,,2.00,;  
 1364, 662,200006,,DAYOH,8,0.10,,,,,726.51,;  
 1385, 675,100006,,DCDOH,8,0.09,,,,,770.58,796.85,;  
 1379, 684,100006,,DDOH,8,0.19,,,,,881.08,910.09,;  
 1348, 696,100006,,DFOH,8,99.99,,,,,3.25,99.99,;  
 1399, 682,200006,,DILF4,8,0.09,,,,,705.07,734.27,;  
 1405, 700,200006,,DOTC6,8,0.06,,,,,865.07,869.47,;  
 1400, 686,200006,,DRME4,8,,,,,9.64,;  
 1357, 658,200006,,EATOH,8,0.20,,,,,5.00,1.00,;  
 1371, 646,200006,,EFDH,8,0.08,,,,,616.58,730.81,;  
 1396, 670,200006,,ENOH,8,99.99,,,,,1.81,;  
 1373, 650,200006,,FAYOH,8,0.09,;  
 1362, 695,200006,,FDY,8,0.16,,,,,2.30,;  
 1364, 645,200006,,FRNOH,8,0.05,,,,,32.95,;  
 1361, 651,100006,,HAM,8,99.99,,,,,59.59,99.99,;  
 1361, 661,200006,,HUFMN,8,3.96,;  
 1385, 679,100006,,HUROH,8,0.08,,,,,877.99,;  
 1397, 694,200006,,KIKC5,8,,,,,10.63,;  
 1410, 700,200006,,LEMD6,8,0.06,,,,,919.10,957.56,;  
 1408, 720,200006,,LEVOH,8,,,,,4.88,;  
 1393, 672,000006,,LNOH,8,0.10,;  
 1401, 704,200006,,MATOH,8,99.99,,,,,1.72,,,,,1.00,;  
 1407, 676,200006,,MCMG5,8,,,,,4.64,;  
 1388, 696,200006,,MELC4,8,6.54,;  
 1367, 648,200006,,MILOH,8,0.12,,,,,6.40,6.44,;  
 1396, 690,200006,,MKWD4,8,0.08,,,,,797.54,806.84,;  
 1373, 640,200006,,MLOH,8,99.99,,,,,99.99,99.99,;  
 1413, 708,000006,,MLPC7,8,0.10,;  
 1391, 699,200006,,MOLC4,8,0.11,,,,,934.67,935.51,;  
 1416, 674,100006,,MRAOH,8,0.22,,,,,18.40,18.90,;  
 1341, 701,000006,,MTOH,8,0.19,,,,,5.30,,,,,8.00,;



Figure 8. Sample plotfile with local pixels.

```

CLEPLT002          ECLEPLT0020010200020481536285014250097501688142912811423
1224,0275,100000Z,12Z 12/29/81;
1224,0259,100000Z,00Z HYDROLOGIC OBSERVATIONS ;
1679,1291,200006,,ACDOH,8,99.99,,,,,5.61,885.50,,;
2413,2029,100006,,AKOH,8,,,,,,5.06,,;
2534,1703,200006,,ATIC6,8,0.06,,,,,894.04,922.83,,;
2106, 883,000006,,ATOH,8,0.14,,,,,6.76,99.99,,;
2615,1341,000006,,BARF6,8,0.08,,;
2385,1779,200006,,BCSC5,8,0.08,,,,,929.96,949.34,,;
1555, 834,200006,,BOOH,8,0.00,,,,,2.56,,;
2454,1792,200006,,BOSC6,8,0.10,,,,,889.70,900.58,,;
2449,1973,000006,,CAK,8,0.09,,,,,1.00,,;
2387,1358,100006,,CAME5,8,2.74,,;
1168,1078,100006,,CCDOH,8,0.13,,,,,846.25,,;
 851,1735,100006,,CEL01,8,0.34,,,,,6.00,,;
1662, 879,000006,,CHOH,8,99.99,,,,,4.71,,;
1172,1307,200006,,CJBOH,8,0.14,,,,,18.00,1009.15,,,,,2.00,,;
2539,1544,200006,,CLBD6,8,0.02,,,,,856.55,893.07,,;
2220,2307,000006,,CLE,8,0.29,,,,,4.00,,;
1971,1839,200006,,CMBC4,8,0.13,,,,,985.04,997.78,,;
 930,1071,200006,,CMDOH,8,,,,,99.99,,;
1644,1269,000006,,CMH0H,8,0.10,,,,,689.30,,;
2235,1514,200006,,COMD5,8,0.10,,,,,9.12,,;
1671,1074,200006,,CROH,8,0.08,,,,,5.06,,;
 834, 764,000006,,CVG,8,0.10,,,,,99.99,,;
1014,1275,000006,,DAY,8,0.18,,,,,2.00,,;
1020,1184,200006,,DAYOH,8,0.10,,,,,726.51,,;
1679,1291,100006,,DCDOH,8,0.09,,,,,770.58,796.85,,;
1610,1588,100006,,DDOH,8,0.19,,,,,881.08,910.09,,;
 940,2220,100006,,DFOH,8,99.99,,,,,3.25,99.99,,;
2125,1341,200006,,DILF4,8,0.09,,,,,705.07,734.27,,;
2464,1735,200006,,DOTC6,8,0.06,,,,,865.07,869.47,,;
2176,1422,200006,,DRME4,8,,,,,9.64,,;
 783,1167,200006,,EATOH,8,0.20,,,,,5.00,1.00,,;
1019, 690,200006,,EFDOH,8,0.08,,,,,616.58,730.81,,;
1918,1041,200006,,ENDH,8,99.99,,,,,1.81,,;
1119, 781,200006,,FAYOH,8,0.09,,;
1303,2055,200006,,FDY,8,0.16,,,,,2.30,,;
 824, 741,200006,,FRNOH,8,0.05,,,,,32.95,,;
 812, 937,100006,,HAM,8,99.99,,,,,59.59,99.99,,;
 924,1174,200006,,HUFMN,8,3.96,,;
1714,1406,100006,,HURDH,8,0.08,,,,,877.99,,;
2172,1661,200006,,KIKC5,8,,,,,10.63,,;
2578,1671,200006,,LEMD6,8,0.06,,,,,919.10,957.56,,;
2732,2197,200006,,LEV0H,8,,,,,4.88,,;
1855,1144,000006,,LNOH,8,0.10,,;
2400,1870,200006,,MATOH,8,99.99,,,,,1.72,,,,,1.00,,;
2252,1103,200006,,MCMG5,8,,,,,4.64,,;
1980,1805,200006,,MELC4,8,6.54,,;
 950, 784,200006,,MILOH,8,0.12,,,,,6.40,6.44,,;
2121,1580,200006,,MKWD4,8,0.08,,,,,797.54,806.84,,;
1016, 518,200006,,MLOH,8,99.99,,,,,99.99,99.99,,;
2727,1846,000006,,MLPC7,8,0.10,,;
2083,1841,200006,,MOLC4,8,0.11,,,,,934.67,935.51,,;
2468, 948,100006,,MRAOH,8,0.22,,,,,18.40,18.90,,;
 827,2426,000006,,MTOH,8,0.19,,,,,5.30,,,,,8.00,,;

```



Figure 9. Sample weather wire output.

CLERUSCLE EWOUS00 KCLE 271500  
 NATIONAL WEATHER SERVICE CLEVELAND OHIO  
 HYDROLOGIC OBSERVATIONS 7AM EST SUNDAY 12/27/81

STATION	FLOOD STAGE	24HR PCPN	RIVER STAGE	24HR CHANGE	SNOW TTL24HR	CURMAXMIN TEMPS	WATER EQUIV PRESENT WX
..... MAUMEE BASIN.....							
NEWVILLE	15		4.81+	1.0			
VAN WERT		0.41					2
FINDLAY	10	0.03	2.52	-0.8			
DEFIANCE	10	MSG	MISSING				
DOWNSTREAM			3.79				
WATERVILLE			5.38	-0.5			
TOLEDO APT		0.13					1
..... VERMILLION BASIN.....							
FITCHVILLE		0.00					
VERMILLION	10		3.39	-0.3			
..... BLACK RIVER BASIN.....							
PENFIELD		0.00					
..... CUYAHOGA RIVER BASIN.....							
BURTON		0.00					
OLD PORTAGE	9		4.73+	0.1			
AKRON			5.10+	0.1			
INDEPENDENCE	16		5.44	-0.3			
CLEVELAND AP		0.00					
..... CHAGRIN RIVER BASIN.....							
WILLOUGHBY	11		MISSING				
..... SCIOTO RIVER BASIN.....							
ALUM CREEK							
DOWNSTREAM			2.39				
HOOVER DAM		0.13	877.84+	0.1			
COLUMBUS USG		0.06	691.20	-1.2			
SEDALIA		0.11					T
CIRCLEVILLE	14	TRACE	6.66	-1.2			
WASHINGTON C		0.00					
CHILLICOTHE	16	0.00	5.80	-0.8			
BOURNEVILLE	10	0.00	2.94	-0.1			
FIKETON	16	TRACE	6.80	-0.9			
..... LITTLE MIAMI RIVER.....							
WILMINGTON		0.10					
MILFORD	17	0.06	6.23	-0.2			
DOWNSTREAM			6.10	-0.2			
FAYETTEVILLE		0.13					
HARSHA		0.15	731.21	-0.3			
DOWNSTREAM			617.65	0.0			
PERINTOWN	19		MISSING				







## APPENDIX 1

## PUPPY Console Messages

Console messages may appear as a result of an error or possible error. The numbers are coded as follows: 001 indicates main program, message number 01; 1202 indicated subroutine PUP12.FR message number 02..., etc.

- 000 -- Start of program.
- 001 -- PUP.ST is empty; 1st record needs to PUP WX Wire routing.  
Solution: run NETBAS.SV program.
- 002 -- End of program.
- 003 -- Code number 84 (RFC file) is missing from PUP.FL.  
Solution: edit AFOS version. Insert code 84; run NETBAS after storing AFOS product in PUPPY.TS.
- 101 -- No input file; save input PIL as PUPPY.IN;  
Solution: enter SAVE command again and check spelling.
- 102 -- Program option less than 1 or greater than 5.  
Solution: edit AFOS input file and correct option.
- 103 -- Wrong date.
- 301 -- Station exists in downstream file but not in master file.
- 401 -- Station does not exist in master file--check spelling!
- 402 -- Comment line was entered; not preceded by a valid station ID;  
Solution: check call sign for accuracy.
- 403 -- Continuation line was entered; not preceded by a valid ID;  
Solution: check call sign for accuracy.
- 404 -- Alphanumeric data was found in data line;  
Solution: only numbers and valid separators can be entered.  
If applicable, send comments as comment lines.
- 405 -- No code 83 found for stranger file.  
Solution: insert code 83 in PUPPY control table in AFOS;  
save as PUPPY.TS; run NETBAS program.
- 406 -- Continuation of 405.
- 501 -- Station reported yesterday but not in NETWORK.DT.  
Solution: check NETWORK.DT if deletion was not intentional.  
Got to backup NETWORK.DT if necessary.
- 601 -- See 404.
- 701 -- No NETWORK.DT.  
Solution: generate NETWORK.DT from scratch or go to backup.
- 702 -- Continuation of 701.
- 801 -- No code 89 found in PUP.BS.  
Solution: edit basin listing in AFOS; insert code 89 and name;  
save as PUPPY.TS; run NETBAS program
- 906 -- PUP.ST file contains error possibly edited by text editor.  
Solution: check file and go to backup.
- 907 -- FORMAT.DT contains new format for input record.  
Solution: check input data; re-execute program.
- 908 -- Error re-reading input file PUPPY.IN.  
Solution: check input data; re-execute program.
- 902 -- Error matching element type.  
Solution: check input data for correct tagging.
- 909 -- Call sign has no data associated with it.  
Solution: insert data or take no action.
- 1001 -- Problem getting a channel.  
Solution: allocate more channels.
- 1002 -- See 1001.
- 1003 -- See 1001.



1004 -- See 1001.  
 1005 -- Problem opening file.  
         Solution: check file; if file in use, clear all files.  
 1006 -- See 1001.  
 1007 -- See 1005.  
 1008 -- Problem reading PUP.DT.  
         Solution: check PUP.DT file; delete if necessary.  
 1009 -- See 1001  
 1010 -- No code 79 found in PUP.FL.  
         Solution: edit AFOS version; insert code 79;  
         save as PUPPY.TS; run NETBAS program.  
 1011 -- No code 87 found in PUP.FL; see 1010.  
 1012 -- No code 79 found in PUP.FL; see 1010.  
 1013 -- No code 87 found in PUP.FL; see 1011  
 1100 -- Start of Weather Wire subroutine.  
 1101 -- See 1001.  
 1104 -- Weather Wire file empty; check data; if true, no action needed.  
 1105 -- Invalid basin number; maximum 25 allowed.  
         Solution: check AFOS version of file control table code 78.  
         check basin numbers for proper columns.  
 1300 -- Start of RFC subroutine.  
 1301 -- No code 89 found; see 801.  
 1302 -- See 1001.  
 1303 -- RFC file empty.  
         Solution: check data; re-execute program.  
 1304 -- Data element either less than 1 or greater than 29.  
         Solution: check data; enter valid tags or compare data with  
         selected default value; see data entry section.  
 1601 -- No code 89 found; see 801.  
 1701 -- AFOS unable to answer PUPPY; possible AFOS crash.  
         Solution: wait for MSG 1703; restart AFOS; re-execute PUPPY.  
 1702 -- See 1701.  
 1703 -- See 1701.

All other messages refer to source code or RDOS Users Manual.



APPENDIX 2  
Source Files

```

-----
C*****
C***** NETWORK PROGRAM: DATABASE MAINT.
C***** DAN PROVOST
C***** NWS RFC CINCINNATI
C***** 5020 FEDERAL OFFICE BLDG
C***** CINCINNATI, OH 45202
C***** FTS 684-2371
C***** VERSION #2
C RLDR NETWORK [NET1,NET2,NET3 NET6,NET4,NET5] NET7 NET9 NET8 FORT.LB
-----
COMMON INET(1500,3),ID(3),IPLACE(10),IREC,ILCTN(20)
COMMON AMXLAT,AMNLAT,AMXLON,AMNLON,MAP,ICHN5
EXTERNAL DESCRIBE,ADDITION,DLTDATA,CHANGE,ADJUST,NEWMAP,OU1
EXTERNAL OU2,OU3,OU4,OU5,OU6
MAP=2 ; NORTH AMERICAN PROJECTION
CALL GCHN(ICHN5,IER)
CALL OPEN(ICHN5,"MONITOR.DT",0,IER)
WRITE(10)"ARE YOU A RIVER FORECAST CENTER? 1=NO , 2=YES "
ACCEPT IRFC
WRITE(10)" "
WRITE(10)" "
WRITE(10)" "
CALL GCHN(ICHN4,IER)
CALL OVLDD(ICHN4,"NETWORK.OL",IER)
WRITE(10,999)
999 FORMAT(3X,'WELCOME TO THE HYDRO DATABASE NETWORK PROGRAM.....',//
1,3X,' MY PROGRAMMING IS DESIGNED TO HELP YOU MAINTAIN YO
2UR',//,3X,'HYDROLOGIC NETWORK FILES. FIRST, I MUST ASK YOU SOME Q
3UESTIONS.',//,3X,'PLEASE BE CAREFUL WITH YOUR ANSWERS.',//)
5 WRITE(10,998)
998 FORMAT(3X,'HERE ARE SEVERAL OPTIONS OF WHAT I CAN DO FOR YOU:',//
1, 5X,'1. DESCRIPTION OF MY FILE STRUCTURE',//, 5X,'2. ADD STATIO
2N TO LISTING',//, 5X,'3. DELETE STATION FROM LISTING',//, 5X,'4.
3 CHANGE STATION IN LISTING ',//, 5X,'5. ADJUST STATION PIXELS',//
4, 5X,'6. REDEFINE LOCAL MAP WINDOW',//, 5X,'7. UPDATE DOWNSTREA
5M FILE, THEN STOP',//,3X,'PLEASE TYPE THE NUMBER OF YOUR CHOICE:
6',//)
READ(11,997)ICHOICE
997 FORMAT(I1)
IF(ICHOICE.GE.6)GO TO 11
CALL LOADNT
11 GO TO (10,20,30,40,50,60,70), ICHOICE
WRITE(10)" "
WRITE(10)"ERROR IN YOUR SELECTION...PLEASE TRY AGAIN"
GO TO 5
10 CALL OVLDD(ICHN4,OU1,1,IER)
CALL DESCRIBE
CALL FOURL(OU1,IER)
GO TO 5
20 CALL OVLDD(ICHN4,OU2,1,IER)
CALL ADDITION
CALL FOURL(OU2,IER)
GO TO 5
30 CALL OVLDD(ICHN4,OU3,1,IER)
CALL DLTDATA
CALL FOURL(OU3,IER)
GO TO 5
40 CALL OVLDD(ICHN4,OU4,-1,IER)
CALL CHANGE

```



```

CALL FOURL(OV4, IER)
GO TO 5
50 CALL OVLDD(ICHN4, OV5, 1, IER)
CALL ADJUST
CALL FOURL(OV5, IER)
GO TO 5
60 CALL OVLDD(ICHN4, OV6, 1, IER)
CALL NEWMAP
CALL FOURL(OV6, IER)
GO TO 5
70 CALL RESET
IF(IRFC.NE.2)CALL FCHAN("NETBAS.SU")
CALL DELETE("MONITOR.DT")
STOP
END

```

NET1.FR

```

OVERLAY OV1
SUBROUTINE DESCRIBE
WRITE(10)"THIS PROGRAM CALLED: NETWORK CONSISTS OF 8 SUBROUTINES."
WRITE(10)"OF THESE, 6 RESIDE IN AN OVERLAY FILE. THE FOLLOWING "
WRITE(10)"LIST BRIEFLY DESCRIBES THE FILES USED BY NETWORK: "
WRITE(10)" "
WRITE(10)" FILE NAME PURPOSE "
WRITE(10)"-----"
WRITE(10)"1. NETWORK.SV EXECUTABLE PROGRAM."
WRITE(10)" "
WRITE(10)"2. NETWORK.OL SUBROUTINE OVERLAY FILE."
WRITE(10)" "
WRITE(10)"3. NETWORK.DT HYDRO DATABASE CONTAINING STATION "
WRITE(10)" INFORMATION: CALL SIGN, STATION NAME, "
WRITE(10)" LATITUDE, LONGITUDE (DEGREES, MINUTES) "
WRITE(10)" COMPUTED POLAR STEREOGRAPHIC PIXELS, "
WRITE(10)" AFOS ZOOM RATIO. THE FIRST RECORD "
WRITE(10)" CONTAINS THE ASSOCIATED MAP BOUNDARY "
WRITE(10)" WITH BASE MAP IN DEGREES-DECIMAL. "
WRITE(10)" "
WRITE(10)"4. NETWORK2.DT TEMPORARY DATABASE WORK FILE WHICH "
WRITE(10)" IS ALWAYS DELETED BY THE PROGRAM. "
WRITE(10)" "
WRITE(10)"5. MONITOR.DT THE CALL SIGN OF EVERY STATION ADDED OR "
WRITE(10)" DELETED IS ENTERED INTO THIS FILE. "
WRITE(10)" "
WRITE(10)"PLEASE MAKE A NEW SELECTION. . ."
RETURN
END

```



NET2.FR

\*\*\*\*\*  
\*\*\*\*\*

NETWORK SUBROUTINE:ADDS STATION  
DAN PROVOST, CINCINNATI RFC

OVERLAY OV2

SUBROUTINE ADDITION

COMMON INET(1500,3),ID(3),IPLACE(10),IREC,ILCTN(20)

COMMON AMXLAT,AMNLAT,AMXLON,AMNLON,MAP,ICHNS

DIMENSION ISAV(20),ELEV(20),ISTRG(20,23),IBUF(36)

KREC=IREC

IJJ=1

\*\*\*\*\*

CALL GCHN(ICHN1,IER)

CALL FOPEN(ICHN1,"NETWORK.DT","B",73)

READ(ICHN1,287,END=300)AMXLAT,AMNLAT,AMXLON,AMNLON

GO TO 214

287 FORMAT(17X,4(F7.2,2X),19X)

300 WRITE(10)"BEGIN BUILDING FILE: "

WRITE(10)"ENTER MAX MAP LATITUDE (DEG. DEC: IE: 38.85 (CR))"

ACCEPT AMXLAT

WRITE(10)"ENTER MIN MAP LATITUDE (DEG. DEC: IE: 35.00 (CR))"

ACCEPT AMNLAT

WRITE(10)"ENTER MAX MAP LONGITUDE (DEG. DEC: IE: 89.20 (CR))"

ACCEPT AMXLON

WRITE(10)"ENTER MIN MAP LONGITUDE (DEG. DEC: IE: 77.00 (CR))"

ACCEPT AMNLON

WRITE(10,286)AMXLAT,AMNLAT,AMXLON,AMNLON

214 CALL GCHN(ICHN2,IER)

CALL FOPEN(ICHN2,"NETWORK2.DT","B",73)

WRITE(ICHN2,286)AMXLAT,AMNLAT,AMXLON,AMNLON

286 FORMAT(2X,'MAP BOUNDED BY ',4(F7.2,2X),19X)

WRITE(10)" "

WRITE(10)"PLEASE DEPRESS RETURN KEY AFTER EACH ANSWER. "

WRITE(10)" "

ISTOR=0

DO 275 I=1,200

WRITE(10)"WHAT IS THE NEW CALL SIGN (HANDBOOK 5.) ? "

READ(11,298)ID

298 FORMAT(3A2)

CALL BINSR(ICANT,ILOC)

IF(ICANT.EQ.1)WRITE(10)"STATION ALREADY EXISTS..."

IF(ICANT.EQ.1)GO TO 261

WRITE(10)"WHAT IS THE NAME OF THIS STATION (MAX 20 CHAR.)?"

WRITE(10)"-----"

READ(11,297)IPLACE

297 FORMAT(10A2)

1 WRITE(10)"ENTER LATITUDE- EXAMPLE: 39 24 ENTER AS 39.24 (CR)"

ACCEPT IDGLT,IMNLT

IOUT=0

IF(IDGLT.LT.0.OR.IDGLT.GT.90)IOUT=1

IF(IMNLT.LT.0.OR.IMNLT.GT.60)IOUT=1

IF(IOUT.EQ.1)WRITE(10)"OUT OF BOUNDS...TRY AGAIN! "

IF(IOUT.EQ.1)GO TO 1

IOUT=0

2 WRITE(10)"ENTER LONGITUDE- EXAMPLE: 84 00 ENTER AS 84.00 (CR)"

ACCEPT IDGLN,IMNLN

IOUT=0

IF(IDGLN.GT.180.OR.IDGLN.LT.0)IOUT=1

IF(IMNLN.LT.0.OR.IMNLN.GT.60)IOUT=1

IF(IOUT.EQ.1)WRITE(10)"OUT OF BOUNDS...TRY AGAIN! "



```

IF(IOUT.EQ.1)GO TO 2
IOUT=0
WRITE(10)"ENTER FLOOD STAGE (NEAREST FOOT--NO DECIMAL) OR 0"
ACCEPT IFLOOD
WRITE(10)"ENTER STATION ELEVATION: EXAMPLE 13004.12"
ACCEPT ELEVA
WRITE(10)"ENTER AFOS ZOOM LEVEL: (0,1,2 OR 3;0=1:1)"
ACCEPT IZM
216 WRITE(10,296)ID,IPLACE,IDGLT,IMNLT,IDGLN,IMNLN,IZM,IFLOOD,ELEVA
296 FORMAT(1X,'YOU ENTERED THIS INFO: ',/,1X,'CALL SIGN: ',
13A2,/,1X,'NAME: ',10A2,/,1X,'LAT DEG-MIN: ',I3,2X,I2,
2/,1X,'LON DEG-MIN: ',I3,2X,I2,/,1X,'AFOS ZOOM: ',I6,/,1X,
3'FLOOD STAGE: ',I6,/,1X,'ELEVATION: ',F8.2,/,/,1X,'CORRECT ?
4 1=YES 2=NO 3=DONT CONTINUE',/)
ACCEPT ISLCT
IF(ISLCT.EQ.1)GO TO 270
IF(ISLCT.EQ.2)GO TO 275
IF(ISLCT.EQ.3)GO TO 261
WRITE(10)"YOUR ANSWER WAS NOT ACCEPTED... TRY AGAIN! "
GO TO 216
270 ISTAR=ISTAR+1
ILCTN(ISTAR)=ILOC
CALL PIXELS(IDGLT,IMNLT,IDGLN,IMNLN,IPIX1,IPIX2,VALU1,VALU2)
IF(IPIX1.EQ.0.OR.IPIX2.EQ.0)WRITE(10,991)ID
991 FORMAT(1X,3A2,' EXCEEDS MAP BOUNDARY; PIXELS NOW = 0')
CALL CART(VALU1,VALU2,MAP,KLON,KLAT,IER)
DO 250 J=1,3
ISTRG(ISTAR,J)=ID(J)
250 CONTINUE
DO 252 J=4,13
ISTRG(ISTAR,J)=IPLACE(J-3) ; 4 TO 13 = 1 TO 10
252 CONTINUE
ISTRG(ISTAR,14)=IDGLT
ISTRG(ISTAR,15)=IMNLT
ISTRG(ISTAR,16)=IDGLN
ISTRG(ISTAR,17)=IMNLN
ISTRG(ISTAR,18)=KLAT/2
ISTRG(ISTAR,19)=KLON/2
C ISTRG(ISTAR,20)=IPIX1
ISTRG(ISTAR,21)=IPIX2
ISTRG(ISTAR,22)=IZM
ISTRG(ISTAR,23)=IFLOOD
ELEV(ISTAR)=ELEVA
WRITE(ICHN5,990)ID
990 FORMAT(1X,3A2,1X,'ADDED')
261 WRITE(10)"ADD ANOTHER STATION ? 1=NO 2=YES "
READ(11,995,ERR=261)IJJ
995 FORMAT(I1)
IF(IJJ.NE.1.AND.IJJ.NE.2)GO TO 261
IF(ISTAR.LT.20.AND.IJJ.EQ.2)GO TO 275
DO 264 K=1,ISTAR
ISAV(K)=K
264 CONTINUE
ISTAR2=ISTAR-1
DO 268 K=1,ISTAR ; BUBBLE SORT FOR NEW ADDITIONS
DO 267 J=1,ISTAR2

```



```

C***** IDENTICAL LOCATION IN INET ARRAY.
C***** COMPARE CALL SIGNS FOR ALPHABET.LISTG
IF(ISTRG(J,1).GT.ISTRG(J+1,1))GO TO 265
IF(ISTRG(J,1).LT.ISTRG(J+1,1))GO TO 267 ; ORDER OK
IF(ISTRG(J,2).GT.ISTRG(J+1,2))GO TO 265
IF(ISTRG(J,2).LT.ISTRG(J+1,2))GO TO 267
IF(ISTRG(J,3).GT.ISTRG(J+1,3))GO TO 265
IF(ISTRG(J,3).LT.ISTRG(J+1,3))GO TO 267
C***** EQUAL CALL SIGNS...BLANK OUT PREVIOUS
ISAV(J)=0
265 IHOLD=ISAV(J+1) ; REFERENCE STORAGE ARRAY LOC #
ISAV(J+1)=ISAV(J)
ISAV(J)=IHOLD
IHOLD=ILCTN(J+1) ; REFERENCE BINARY ARRAY LOC #
ILCTN(J+1)=ILCTN(J); USED TO EXECUTE NETWORK WRITES
ILCTN(J)=IHOLD
DO 269 KL=1,3
IHOLD=ISTRG(J,KL)
ISTRG(J,KL)=ISTRG(J+1,KL)
ISTRG(J+1,KL)=IHOLD
269 CONTINUE
267 CONTINUE
268 CONTINUE
KNTR=0
DO 280 K=1,ISTOR
KNTR=ILCTN(K)-KNTR
DO 282 KK=1,KNTR
READ(ICHN1,998,END=285)IBUF
WRITE(ICHN2,997)IBUF
998 FORMAT(36A2)
997 FORMAT(1X,36A2)
282 CONTINUE
IF(ISAV(K).EQ.0)GO TO 279
JJ=ISAV(K)
WRITE(ICHN2,996)(ISTRG(K,J),J=1,3),(ISTRG(JJ,J),J=4,23),ELEV(JJ)
996 FORMAT(1X,13A2,1X,I4,I2,I3,I2,1X,4I4,I2,I6,F8,2,1X)
279 KNTR=ILCTN(K)
280 CONTINUE
DO 284 KL=1,IREC
READ(ICHN1,998,END=285)IBUF
WRITE(ICHN2,997)IBUF
284 CONTINUE
285 Istor=0
CALL CLOSE(ICHN1)
CALL CLOSE(ICHN2)
CALL DELETE("NETWORK.DT")
CALL RENAM("NETWORK2.DT","NETWORK.DT",IER)
IF(IJJ.NE.2)GO TO 276
CALL LOADNT
CALL GCHN(ICHN1,IER)
CALL FOPEN(ICHN1,"NETWORK.DT","B",73)
CALL GCHN(ICHN2,IER)
CALL FOPEN(ICHN2,"NETWORK2.DT","B",73)
CALL FSEEK(ICHN1,1)
CALL FSEEK(ICHN2,0)
WRITE(ICHN2,286)AMXLAT,AMNLAT,AMXLON,AMNLON
275 CONTINUE
276 RETURN
END

```



NET3.FR

NETWORK SUBROUTINE: DELETES STATION  
DAN PROVDST-CINCINNATI RFC

C\*\*\*\*\*  
C\*\*\*\*\*

```
OVERLAY OV3
SUBROUTINE DLTDATA
COMMON INET(1500,3),ID(3),IPLACE(10),IREC,ILCTN(20)
COMMON AMXLAT,AMNLAT,AMXLON,AMNLON,MAP,ICHN5
DIMENSION IBUF3(36)
301 CALL GCHN(ICHN1,IER)
CALL OPEN(ICHN1,"NETWORK.DT",0,IER)
CALL GCHN(ICHN2,IER)
CALL OPEN(ICHN2,"NETWORK2.DT",0,IER)
READ(ICHN1,999,END=320)IBUF3
WRITE(ICHN2,998)IBUF3
999 FORMAT(36A2)
998 FORMAT(1X,36A2)
ISTOR=0
331 WRITE(10)"ENTER CALL SIGN OF STATION YOU WISH DELETED:"
READ(11,997)ID
997 FORMAT(3A2)
CALL BINSR(ICANT,ILOC)
IF(ICANT.EQ.0)WRITE(10)"STATION DOES NOT EXIST...TRY AGAIN!"
IF(ICANT.EQ.0)GO TO 303
DO 302 I=1,ISTOR
IF(ILCTN(I).EQ.ILOC)WRITE(10)"ALREADY DELETED!"
IF(ILCTN(I).EQ.ILOC)GO TO 303
302 CONTINUE
ISTOR=ISTOR+1
ILCTN(ISTOR)=ILOC
WRITE(ICHN5,990)ID
990 FORMAT(1X,3A2,1X,'DELETED')
303 WRITE(10)"DELETE ANOTHER STATION? 1=NO 2=YES "
READ(11,996,ERR=303)IJJ
IF(IJJ.NE.1.AND.IJJ.NE.2)GO TO 303
996 FORMAT(I1)
IF(ISTOR.LT.20.AND.IJJ.EQ.2)GO TO 331
C***** NO MORE INPUT OR 20 STATIONS...UPDATE MASTER
DO 305 I=1,ISTOR
ISTOR2=ISTOR-1
DO 306 J=1,ISTOR2
IF(ILCTN(J).LT.ILCTN(J+1))GO TO 306
IF(ILCTN(J).EQ.ILCTN(J+1))WRITE(10)"LOGIC ERROR"
ISAV=ILCTN(J+1)
ILCTN(J+1)=ILCTN(J)
ILCTN(J)=ISAV
306 CONTINUE
305 CONTINUE
I=1
DO 335 K=1,IREC
READ(ICHN1,999,END=336)IBUF3
IF(K.NE.ILCTN(I))GO TO 334
I=I+1
GO TO 335
334 WRITE(ICHN2,998)IBUF3
335 CONTINUE
336 ISTOR=0
CALL CLOSE(ICHN1)
CALL CLOSE(ICHN2)
CALL DELETE("NETWORK.DT")
```



```

CALL RENAM("NETWORK2.DT","NETWORK.DT",IER)
IF(IJJ.EQ.2)CALL LOADNT
IF(IJJ.EQ.2)GO TO 301
RETURN
320 CALL CLOSE(ICHN1)
CALL CLOSE(ICHN2)
CALL DELETE("NETWORK.DT")
CALL DELETE("NETWORK2.DT")
WRITE(10)"NO MASTER FILE EXISTS (NETWORK.DT)"
WRITE(10)"PLEASE ADD A STATION!"
RETURN
END

```

NET4.FR

```

C*****
C***** NETWORK SUBROUTINE:CHANGES STATION
C***** DAN PROVOST, CINCINNATI RFC
OVERLAY 004
SUBROUTINE CHANGE
COMMON INET(1500,3),ID(3),IPLACE(10),IREC,ILOCTN(20)
COMMON AMXLAT,AMNLAT,AMXLON,AMNLON,MAP,ICHN5
DIMENSION IBUF(36)
CALL GCHN(ICHN1,IER)
CALL FOPEN(ICHN1,"NETWORK.DT","B",73)
READ(ICHN1,998,END=435)IBUF
401 WRITE(10)"ENTER CALL SIGN OF STATION YOU WISH TO CHANGE:"
READ(11,999)ID
999 FORMAT(3A2)
CALL BINSR(ICANT,ILOC)
IF(ICANT.EQ.0)WRITE(10)"STATION DOES NOT EXIST...TRY AGAIN"
IF(ICANT.EQ.0)GO TO 410
CALL FSEEK(ICHN1,ILOC) ; ILOC = LOCATION + 1 FOR FSEEK
READ(ICHN1,998,END=410)IBUF
998 FORMAT(36A2)
WRITE(10)"RECORD CURRENTLY IN FILE: "
WRITE(10,997)IBUF
997 FORMAT(1X,36A2)
WRITE(10)
WRITE(10)"RE-ENTER CALL SIGN:"
READ(11,999)ID
IF(ID(1).EQ.IBUF(1).AND.ID(2).EQ.IBUF(2).AND.ID(3).EQ.IBUF(3))GOTO
1425
WRITE(10)"SINCE YOU ARE CHANGING THE CALL SIGN, PLEASE DELETE THEN"
WRITE(10)"ADD THIS STATION."
GO TO 410
425 WRITE(10)"WHAT IS THE NAME OF THIS STATION (MAX 20 CHAR.)?"
WRITE(10)"-----"
READ(11,996)IPLACE
996 FORMAT(10A2)
1 WRITE(10)"ENTER LATITUDE- EXAMPLE: 39 24 ENTER AS 39.24 (CR)"
ACCEPT IDGLT,IMNLT
IOUT=0
IF(IDGLT.LT.0.OR.IDGLT.GT.90)IOUT=1
IF(IMNLT.LT.0.OR.IMNLT.GT.60)IOUT=1
IF(IOUT.EQ.1)WRITE(10)"OUT OF BOUNDS...TRY AGAIN!"
IF(IOUT.EQ.1)GO TO 1
IOUT=0

```



```

2 WRITE(10)"ENTER LONGITUDE- EXAMPLE: 84 00 ENTER AS 84.00 (CR)"
ACCEPT IDGLN,IMNLN
IOUT=0
IF(IDGLN.LT.0.OR.IDGLN.GT.180)IOUT=1
IF(IMNLN.LT.0.OR.IMNLN.GT.60)IOUT=1
IF(IOUT.EQ.1)WRITE(10)"OUT OF BOUNDS...TRY AGAIN! "
IF(IOUT.EQ.1)GO TO 2
IOUT=0
WRITE(10)"ENTER FLOOD STAGE (NEAREST FOOT--NO DECIMAL) OR 0 "
ACCEPT IFLOOD
WRITE(10)"ENTER STATION ELEVATION: EXAMPLE 13004.12 "
ACCEPT ELEVA
WRITE(10)"ENTER AFOS ZOOM LEVEL: (0,1,2 OR 3;0=1:1)"
ACCEPT IZM
480 WRITE(10,995)ID,IPLACE,IDGLT,IMNLT,IDGLN,IMNLN,IZM,IFLOOD,ELEVA
995 FORMAT(1X,'YOU ENTERED THIS INFO: ',/,1X,'CALL SIGN: ',
13A2,/,1X,'NAME: ',10A2,/,1X,'LAT DEG-MIN: ',I3,2X,I2,
2/,1X,'LON DEG-MIN: ',I3,2X,I2,/,1X,'AFOS ZOOM: ',I6,/,1X
3'FLOOD STAGE: ',I6,/,1X,'ELEVATION: ',F8.2,/,1X,'CORRECT ?
4 1=YES 2=NO 3=DONT CONTINUE',/)
ACCEPT ISLCT
IF(ISLCT.EQ.1)GO TO 470
IF(ISLCT.EQ.2)GO TO 401
IF(ISLCT.EQ.3)GO TO 410
WRITE(10)"YOUR ANSWER WAS NOT ACCEPTED...TRY AGAIN! "
GO TO 480
470 CALL PIXELS(IDGLT,IMNLT,IDGLN,IMNLN,IPIX1,IPIX2,VALU1,VALU2)
IF(IPIX1.EQ.0.OR.IPIX2.EQ.0)WRITE(10,992)ID
992 FORMAT(1X,3A2,' EXCEEDS MAP BOUNDARY; PIXELS NOW =0 ')
CALL CART(VALU1,VALU2,MAP,KLON,KLAT,IER)
KLAT=KLAT/2 ;PMOD USES .PF MODE FOR PLT FORMAT
KLON=KLON/2 ; .CF MODE FOR GRAPHICS
CALL FSEEK(ICHN1,ILOC)
WRITE(ICHN1,994)ID,IPLACE,IDGLT,IMNLT,IDGLN,IMNLN,KLAT,KLON,IPIX1,
IPIX2,IZM,IFLOOD,ELEVA
994 FORMAT(1X,13A2,1X,I4,I2,I3,I2,1X,4I4,I2,I6,F8.2,1X)
410 WRITE(10)"CHANGE ANOTHER STATION? 1=NO 2=YES"
READ(11,993,ERR=410)IJJ
993 FORMAT(I1)
IF(IJJ.NE.1.AND.IJJ.NE.2)GO TO 410
IF(IJJ.EQ.2)GO TO 401
CALL CLOSE(ICHN1)
RETURN
435 WRITE(10)"NO MASTER FILE EXISTS (NETWORK.DT)."
WRITE(10)"PLEASE ADD A STATION!"
CALL CLOSE(ICHN1)
RETURN
END

```



NET5.FR

OVERLAY 005

SUBROUTINE ADJUST

COMMON INET(1500,3),ID(3),IPLACE(10),IREC,ILCTN(20)

COMMON AMXLAT,AMNLAT,AMXLON,AMNLON,MAP,ICHN5

DIMENSION IBUF(23),IBUF2(8)

CALL GCHN(ICHN,IER)

CALL FOPEN(ICHN,"NETWORK.DT","B",73)

READ(ICHN,999,END=520)ID

WRITE(10)"TO MOVE STATION LEFT= - NBR      RIGHT= + NBR"

WRITE(10)"                                      DOWN= - NBR      UP = + NBR "

WRITE(10)"I.E. -20,0 MOVES STATION 20 PIXELS LEFT, 0 PIXELS UP "

501 WRITE(10)"ENTER CALL SIGN OF STATION YOU WISH TO ADJUST:"

READ(11,999)ID

999 FORMAT(3A2,66X)

CALL BINSR(ICANT,ILOC)

IF(ICANT.EQ.0)WRITE(10)"STATION DOES NOT EXIST...TRY AGAIN!"

IF(ICANT.EQ.0)GO TO 510

CALL FSEEK(ICHN,ILOC)

READ(ICHN,998,END=515)IBUF,IPIX1,IPIX2,IBUF2

998 FORMAT(13A2,1X,10A2,2I4,A2,1X,7A2)

WRITE(10)"STATION CURRENTLY IN FILE:"

WRITE(10,997)IBUF,IPIX1,IPIX2,IBUF2

WRITE(10)" "

WRITE(10)"ENTER ADJUSTMENT FACTORS IN PIXELS:LAT,LON (IE. -20,5)"

ACCEPT ILT,ILN

IPIX1=IPIX1+ILT

IPIX2=IPIX2+ILN

CALL FSEEK(ICHN,ILOC)

WRITE(ICHN,997)IBUF,IPIX1,IPIX2,IBUF2

997 FORMAT(1X,13A2,1X,10A2,2I4,A2,1X,7A2)

505 WRITE(10)"ADJUST ANOTHER STATION? 1=NO 2=YES"

READ(11,996,ERR=505)IJJ

996 FORMAT(I1)

IF(IJJ.NE.1.AND.IJJ.NE.2)GOTO 505

IF(IJJ.EQ.2)GO TO 501

510 CALL CLOSE(ICHN)

RETURN

515 WRITE(10)"INVALID SEARCH...TRY AGAIN!"

GO TO 505

520 WRITE(10)"NO MASTER FILE EXISTS (NETWORK.DT)."

WRITE(10)"PLEASE ADD A STATION!"

RETURN

END

NET6.FR

C\*\*\*\*\*  
C\*\*\*\*\* NETWORK SUBROUTINE: NEW WINDOW PIXELS  
C\*\*\*\*\* DAN PROVOST, CINCINNATI RFC

OVERLAY 006

SUBROUTINE NEWMAP

COMMON INET(1500,3),ID(3),IPLACE(10),IREC,ILCNT(20)

COMMON AMXLAT,AMNLAT,AMXLON,AMNLON,MAP,ICHN5

DIMENSION IBUF(13)



```

CALL GCHN(ICHN1, IER)
CALL OPEN(ICHN1, "NETWORK.DT", 0, IER)
CALL GCHN(ICHN2, IER)
CALL FOPEN(ICHN2, "NETWORK2.DT", "B", 73)
READ(ICHN1, 699, END=630)BMXLAT, BMNLAT, BMXLON, BMNLON
WRITE(10)"CURRENT WINDOW:"
WRITE(10, 698)BMXLAT, BMNLAT, BMXLON, BMNLON
699 FORMAT(17X, 4(F7.2, 2X), 19X)
WRITE(10)"RE-DEFINE NEW MAP WINDOW:"
WRITE(10)"ENTER MAX MAP LATITUDE (DEG. DEC: IE: 38.85 (CR)):"
ACCEPT CMXLAT
WRITE(10)"ENTER MIN MAP LATITUDE (DEG. DEC: IE: 35.00 (CR)):"
ACCEPT CMNLAT
WRITE(10)"ENTER MAX MAP LONGITUDE (DEG. DEC: IE: 89.20 (CR)):"
ACCEPT CMXLON
WRITE(10)"ENTER MIN MAP LONGITUDE (DEG. DEC: IE: 77.00 (CR)):"
ACCEPT CMNLON
WRITE(10, 698)CMXLAT, CMNLAT, CMXLON, CMNLON
WRITE(ICHN2, 698)CMXLAT, CMNLAT, CMXLON, CMNLON
698 FORMAT(2X, 'MAP BOUNDED BY ', 4(F7.2, 2X), 19X)
ISWITCH=0
DO 610 I=1, 1500
READ(ICHN1, 697, END=620)IBUF, I1, I2, I3, I4, I5, I6, I7, I8, I9, IFLOOD, ELEV
AMXLAT=BMXLAT
AMNLAT=BMNLAT
AMXLON=BMXLON
AMNLON=BMNLON
JPIX1=I7
JPIX2=I8
CALL PIXELS(I1, I2, I3, I4, I7, I8, VALU1, VALU2)
JPIX1=JPIX1-I7
JPIX2=JPIX2-I8
IF(JPIX1.EQ.0.AND.JPIX2.EQ.0)GO TO 605
WRITE(10, 695)(IBUF(II), II=1, 3), JPIX1, JPIX2
695 FORMAT(1X, 'STATION ', 3A2, ' HAD PIXEL ADJUSTMENTS OF ', I5, ', ', I5)
605 AMXLAT=CMXLAT
AMNLAT=CMNLAT
AMXLON=CMXLON
AMNLON=CMNLON
697 FORMAT(13A2, 1X, I4, I2, I3, I2, 1X, 4I4, I2, I6, F8.2, 1X)
696 FORMAT(1X, 13A2, 1X, I4, I2, I3, I2, 1X, 4I4, I2, I6, F8.2, 1X)
CALL PIXELS(I1, I2, I3, I4, I7, I8, VALU1, VALU2)
IF(I7.EQ.0.OR.I8.EQ.0)WRITE(10, 694)(IBUF(II), II=1, 3)
694 FORMAT(1X, 3A2, ' EXCEEDS MAP BOUNDARY; PIXELS NOW = 0 ')
WRITE(ICHN2, 696)IBUF, I1, I2, I3, I4, I5, I6, I7, I8, I9, IFLOOD, ELEV
610 CONTINUE
620 WRITE(10)"ALL NEW LOCAL PIXELS MATCH NEW MAP WINDOW WITH NO ADJUSTMENTS."
CALL CLOSE(ICHN1)
CALL CLOSE(ICHN2)
CALL DELETE("NETWORK.DT")
CALL RENAME("NETWORK2.DT", "NETWORK.DT", IER)
RETURN
630 WRITE(10)"NO MASTER FILE EXIST (NETWORK.DT)."
WRITE(10)"PLEASE ADD A STATION!"
CALL CLOSE(ICHN1)
CALL CLOSE(ICHN2)
RETURN
END

```



## NET7.FR

```

SUBROUTINE LOADNT
COMMON INET(1500,3),ID(3),IPLACE(10),IREC,ILCTN(20)
COMMON AMXLAT,AMNLAT,AMXLON,AMNLON,MAP,ICHNS
IREC=0
K=0
DO 60 I=1,20
60 ILCTN(I)=K
CALL GCHN(ICHN,IER)
CALL OPEN(ICHN,"NETWORK.DT",0,IER)
READ(ICHN,999,END=100)AMXLAT,AMNLAT,AMXLON,AMNLON
999 FORMAT(17X,4(F7.2,2X),19X)
998 FORMAT(3A2)
IREC=0
DO 80 I=1,1500
READ(ICHN,998,END=81)(INET(I,J),J=1,3)
IREC=IREC+1
80 CONTINUE
81 CALL CLOSE(ICHN)
RETURN
100 CALL CLOSE(ICHN)
WRITE(10)"NO MASTER FILE EXISTS (NETWORK.DT). "
WRITE(10)"PLEASE ADD A STATION!"
RETURN
END

```

## NET8.FR

```

SUBROUTINE PIXELS(IDGLT,IMNLT,IDGLN,IMNLN,IPIX1,IPIX2,VALU1,VALU2)
COMMON INET(1500,3),ID(3),IPLACE(10),IREC,ILCTN(20)
COMMON AMXLAT,AMNLAT,AMXLON,AMNLON,MAP,ICHNS
C***** ESTABLISH POLAR STEREOGRAPHIC PROJECTION *****
C***** CONVERT FROM DEGREES TO RADIANs ***
RLATHI=AMXLAT * .017453
RLATLO=AMNLAT * .017453
RLONHI=AMXLON * .017453
RLONLO=AMNLON * .017453
ADJ=((AMXLON + AMNLON)/2.0) - 90.
RLONLO= -RLONLO
RLONHI= -RLONHI
COEF1= 4680. * TAN(.785396 - .5 * RLATHI)
COEF3= 4680. * TAN(.785396 - .5 * RLATLO)
COEF2= RLONLO + ADJ * .017453
COEF4= RLONHI + ADJ * .017453
RLONHI= -RLONHI
RLONLO= -RLONLO
XTR= COEF1 * COS(COEF2)
XTL= COEF1 * COS(COEF4)
XBR= COEF3 * COS(COEF2)
XBL= COEF3 * COS(COEF4)
YTR= COEF1 * SIN(COEF2)
YTL= COEF1 * SIN(COEF4)
YBR= COEF3 * SIN(COEF2)
YBL= COEF3 * SIN(COEF4)

```



```

RT= YTR
IF(YTR.LT.YTL)RT=YTL
RB=YBR
IF(YBR.GT.YBL)RB=YBL
RR=XTR
IF(XTR.LT.XBR)RR=XBR
RL=XTL
IF(XTL.GT.XBL)RL=XBL
XA=4095./<(RR-RL)
YA=3071./<(RT-RB)
IF(XA.GT.YA)XA=YA
YA=XA
XB=RL*XA
YB=RB*YA
DGLT=IDGLT
DGLN=IDGLN
FMNLT=IMNLT
FMNLN=IMNLN
VALU1=DGLT+(FMNLT*100/6000)
VALU2=DGLN+(FMNLN*100/6000)
VALU3=VALU1 * .017453
VALU4=VALU2 * .017453
XM=4680. * TAN(.785396 - .5 * VALU3)
YM=VALU4 - ADJ * .017453
IPIX2=XM * COS(YM) * XA - XB
IPIX1=(-XM * SIN(YM)) * YA-YB
IF(IPIX1.GT.3071.OR.IPIX1.LT.0)IPIX1=0
IF(IPIX2.GT.4095.OR.IPIX2.LT.0)IPIX2=0
IF(IPIX1.EQ.0)IPIX2=0
IF(IPIX2.EQ.0)IPIX1=0
RETURN
END

```

NET9.FR

C\*\*\*\*\*  
C\*\*\*\*\*

NETWORK SUBROUTINE: BINARY SEARCH  
DAN PROUDST-CINCINNATI RFC

```

SUBROUTINE BINSR(ICANT,ILOC)
COMMON INET(1500,3),KD(3),IPLACE(10),IREC,ILCTN(20)
COMMON AMXLAT,AMNLAT,AMXLON,AMNLON,MAP,ICHNS
ILOC=0 ; LOCATION OF LAST ID CHECKED OR MATCHED
ICANT=0 ; ASSUME NO MATCH UNTIL FOUND
ILO=0 ; "AA" IS LESS THAN "AB"
IHI=IREC+1
10 IAM=(ILO+IHI)/2
IF(IAM-ILO)95,95,20
20 IF(KD(1)-INET(IAM,1))30,60,40
30 IHI=IAM ;SEARCH IS FOR LOWER HALF
GO TO 10
40 ILO=IAM ;SEARCH IS FOR UPPER HALF
GO TO 10
60 CONTINUE
IF(INET(IAM,1).NE.KD(1))GO TO 95
DO 65 NELT=2,3
IF(KD(NELT)-INET(IAM,NELT))70,65,80
65 CONTINUE

```



```

GO TO 90
70 IAM=IAM-1
IF(IAM.EQ.0)GO TO 100
NELT=2
IF(INET(IAM,NELT).LT.KD(NELT))GO TO 95
GO TO 60
80 IAM=IAM+1
IF(IAM.GT.IREC)GO TO 98
NELT=2
IF(INET(IAM,NELT).GT.KD(NELT))GO TO 95
GO TO 60
90 CONTINUE
ILOC=IAM
ICANT=1 ; MATCH OF ID FOUND
RETURN
95 IF(IAM.EQ.0)GO TO 100
DO 99 KK=1,3
IF(KD(KK)-INET(IAM,KK))98,99,100
99 CONTINUE
98 IAM=IAM-1
100 ILOC=IAM
RETURN
END

```

```

-----
C***** NETBAS PROGRAM
C***** WRITTEN BY: DAN PROVOST
C***** 5020 FEDERAL OFFICE BLDG
C***** CINCINNATI, OHIO 45202
C***** FTS 684-2371
C PROGRAM ALLOWS DOWNSTREAM ORDER ENTRY AND ELEMENT SELECTION
C STATIONS ARE ALREADY IN MONITOR.DT FROM NETWORK PROGRAM
C RLDR NETBAS NETBAS1 NETBAS2 TRNS UTIL.LB BG.LB FORT.LB
C
C MAIN PROGRAM
C
-----

```

```

DIMENSION INAM(35), ITRANS(20,35), ISTRG2(6), IBUF(36)
INTEGER YES
COMMON/LLL/ISTRG1(6)
DATA ISTRG1/"PUP.ST" "/"
WRITE(10)**NETBAS: PROGRAM EXECUTING !"
YES="Y "
IEOF=0
IADD="AD"
IDEL="DE"
IELEM1=35 ;NBR=1 ID=3 BASIN=1 #OF ELE=1 + 29 ELEM =35 LOCATIONS
IELEM2=29

```



```

IST=0 ; FILE DOES NOT EXISTS, 1=EXISTS
CALL STAT(ISTRG1,INAM,IST)
CALL CORREC(IDONT)
CALL STATUS
CALL GCHN(ICHN1,IER)
CALL OPEN(ICHN1,"MONITOR.DT",0,IER)
J1=1
J2=1
J3=1
J4=1
J5=1
J6=1
J7=1
J8=1
J9=1
WRITE(10)"ALL DATA ELEMENTS ARE TRANSMITTED TO WEATHER WIRE:"
WRITE(10)"DO YOU WISH TO SUPPRESS SOME ELEMENTS?"
READ(11,986)IWXWR,K
986 FORMAT(A1,A2)
IF(IWXWR.NE.YES)GO TO 3
WRITE(10)"IF YOU WISH TO SEND THIS ELEMENT TO WX WIRE ANSWER YES!"
WRITE(10)" "
WRITE(10)"PRECIPITATION?"
READ(11,986)J,K
IF(J.NE.YES)J1=0
WRITE(10)"RIVER STAGES?"
READ(11,986)J,K
IF(J.NE.YES)J2=0
WRITE(10)"24-HOUR CHANGES IN STAGES?"
READ(11,986)J,K
IF(J.NE.YES)J3=0
WRITE(10)"FLOOD STAGE?"
READ(11,986)J,K
IF(J.NE.YES)J4=0
WRITE(10)"SNOW (TTL AND NEW)?"
READ(11,986)J,K
IF(J.NE.YES)J5=0
WRITE(10)"TEMPERATURES?"
READ(11,986)J,K
IF(J.NE.YES)J6=0
WRITE(10)"WATER EQUIVALENT?"
READ(11,986)J,K
IF(J.NE.YES)J7=0
WRITE(10)"PRESENT WEATHER?"
READ(11,986)J,K
IF(J.NE.YES)J8=0
WRITE(10)"RIVER FLOW?"
READ(11,986)J,K
IF(J.NE.YES)J9=0
3 ICOUNT=0
5 DO 49 II=1,20 ; MAX 20 STATION IN MONITOR.DT
10 READ(ICHN1,999,END=48)INAM
ICOUNT=ICOUNT+1
IF(INAM(4).EQ.IADD)GO TO 20
IF(INAM(4).NE.IDEL)GO TO 10
999 FORMAT(3A2,1X,A2,31A2)
WRITE(10)"FROM YOUR AFOS STATION LISTING...ENTER STATION NUMBER"
WRITE(10,998)(INAM(I),I=1,3)
998 FORMAT(1X,'TO DELETE STATION: ',3A2)

```



```

WRITE(10)"?"
ACCEPT ITRANS(II,1)
ITRANS(II,2)=9999
GO TO 49
20 IF(IST.NE.1)GO TO 17
WRITE(10)"FROM YOUR AFOS STATION LISTING...ENTER STATION NUMBER"
WRITE(10,997)(INAM(I),I=1,3)
997 FORMAT(1X,'WHICH WILL APPEAR AFTER ',3A2,' (SEE DOCUMENTATION)')
WRITE(10)"?"
ACCEPT ITRANS(II,1)
17 WRITE(10,990)(INAM(I),I=1,3)
990 FORMAT(1X,'ENTER ',3A2,' BASIN NUMBER')
IF(IST.NE.1)ITRANS(II,1)=1 ;AVOIDS ENTERING 1 FOR 1ST EXECUTION
WRITE(10)"?"
ACCEPT ITRANS(II,2)
ITRANS(II,3)=INAM(1)
ITRANS(II,4)=INAM(2)
ITRANS(II,5)=INAM(3)
21 WRITE(10)"MAXIMUM NUMBER OF DEFAULT ELEMENTS REPORTING?"
ACCEPT IMAX
IF(IMAX.GT.IELEM2)IMAX=IELEM2
IF(IMAX.GT.IELEM2)WRITE(10)"MAXIMUM ALLOWED= ",IELEM2
IF(IMAX.EQ.0)WRITE(10)"IF IT DOESN'T REPORT ANYTHING...WHY BOTHER?!"
IF(IMAX.EQ.0)GO TO 21
ITRANS(II,6)=IMAX
WRITE(10)"ENTER ELEMENT ORDER YOU WILL USE FOR DEFAULT INPUT:"
ILIMIT=6+IMAX
TYPE "ENTER",IMAX," ELEMENT NUMBERS SEPARATED BY , OR (CR):"
ACCEPT (ITRANS(II,L),L=7,ILIMIT)
ILIMIT=ILIMIT+1
IF(ILIMIT.GT.35)GO TO 49
DO 46 LL=ILIMIT,35
ITRANS(II,LL)=0
46 CONTINUE
49 CONTINUE
GO TO 50
48 IEOF=1
50 DO 60 I=1,ICOUNT
JCOUNT=ICOUNT-1
DO 55 J=1,JCOUNT
IF(ITRANS(J,1).LT.ITRANS(J+1,1))GO TO 55
IF(ITRANS(J,2).LE.ITRANS(J+1,2))GO TO 55
DO 47 KK=1,IELEM1
ISAV=ITRANS(J,KK)
ITRANS(J,KK)=ITRANS(J+1,KK)
ITRANS(J+1,KK)=ISAV
47 CONTINUE
55 CONTINUE
60 CONTINUE
IF(ICOUNT.EQ.0.AND.IWXWR.NE.YES)GO TO 75
CALL GCHN(ICHN2,IER)
CALL OPEN(ICHN2,"PUP.ST",0,IER)
CALL GCHN(ICHN3,IER)
CALL OPEN(ICHN3,"PUP2.ST",0,IER)
READ(ICHN2,989,END=58)IBUF
IF(IWXWR.EQ.YES)GO TO 58
WRITE(ICHN3,988)IBUF
GO TO 57
58 WRITE(ICHN3,987)J1,J2,J3,J4,J5,J6,J7,J8,J9

```



```

IWXWR=0
57 K=1
JBASN=1
IJ=1
IREC=0
IEND=0
DO 65 J=1,1500
READ(ICHN2,995,END=63)INAM
IREC=IREC+1
GO TO 62
63 IF(K.EQ.(ICOUNT+1))GO TO 66
IEND=1
62 IF(K.GT.ICOUNT)GO TO 68
IF(ITRANS(K,1).GT.IREC.AND.IEND.EQ.0)GO TO 68
IF(ITRANS(K,2).EQ.9999.AND.ITRANS(K,1).EQ.IREC)GO TO 69
IB=35 ; MAX # OF LOCATIONS WRITTEN
IF(ITRANS(K,2).GE.JBASN)GO TO 61
ITRANS(K,2)=JBASN ; CORRECTION MADE
WRITE(10,992)IJ,(ITRANS(K,JJ),JJ=3,5)
992 FORMAT(1X,I4,1X,3A2,' BASIN NBR OUT OF SEQUENCE... AUTOMATIC CORREC
TION INITIATED!')
989 FORMAT(36A2)
988 FORMAT(1X,36A2)
987 FORMAT(1X,'WX WIRE: ',9I1,53X)
61 JBASN=ITRANS(K,2)
WRITE(ICHN3,996)IJ,(ITRANS(K,JJ),JJ=3,5),(ITRANS(K,2))
1,(ITRANS(K,LL),LL=6,IB)
IJ=IJ+1
K=K+1
GO TO 62
68 IF(IEND.EQ.1.AND.K.GT.ICOUNT)GO TO 66
IF(ITRANS(K,2).EQ.9999.AND.ITRANS(K,1).EQ.IREC)GO TO 69
IF(INAM(5).GE.JBASN)GO TO 67
KJ=IJ-1
WRITE(10,991)KJ
991 FORMAT(1X,'STATION ',I3,' BASIN OUT OF SEQUENCE!',
1/,1X,'DELETE THIS STATION ...RE-ENTER WITH CORRECT BASIN')
67 WRITE(ICHN3,994)IJ,(INAM(LL),LL=2,35)
JBASN=INAM(5)
IJ=IJ+1
GO TO 65
69 K=K+1
995 FORMAT(I3,3A2,I2,1X,30A2)
994 FORMAT(1X,I3,3A2,I2,1X,30A2)
996 FORMAT(1X,I3,3A2,I2,1X,30I2)
65 CONTINUE
66 CALL CLOSE(ICHN2)
CALL CLOSE(ICHN3)
CALL DELETE("PUP.ST")
CALL RENAM("PUP2.ST","PUP.ST",IER)
IDONT=1
IF(IEOF.EQ.0)GO TO 75
CALL CLOSE(ICHN1)
CALL DELETE("MONITOR.DT")
75 IF(ICOUNT.EQ.0.AND.IDONT.EQ.0)STOP
CALL GCHN(ICHN5,IER)
CALL OPEN(ICHN5,"PUP.FL",0,IER)
30 READ(ICHN5,993,END=31)ICODE,ISTRG2
993 FORMAT(I2,2X,6A2)

```



```

IF(ICODE.NE.0)GO TO 30 ; SEARCHES CCCNNNXXX FOR 81 CODE
IF(IEOF.NE.0)WRITE(10)"AFOS FILE UPDATED...PROGRAM CONTINUING!"
WRITE(10,998)(ISTRG2(LLL),LLL=1,5)
980 FORMAT(1X,"OBTAIN NEW AFOS STATION LISTING FROM: ",4A2,A1)
CALL CLOSE(ICHN5)
CALL PUP(ISTRG1,ISTRG2)
IST=1
ICOUNT=0
IF(IEOF.EQ.0)GO TO 5
WRITE(10)"NETBAS: PROGRAM COMPLETED!"
STOP
31 WRITE(10)"NO CCCNNNXXX FOUND FOR CODE 81 IN PUP.FL"
WRITE(10)"FOR ASSISTANCE PLEASE SEE DOCUMENTATION"
WRITE(10)"UNABLE TO STORE STATION LIST IN AFOS..."
CALL CLOSE(ICHN5)
IF(ICOUNT.EQ.0)STOP
WRITE(10)"RDOS UPDATE CONTINUING..."
ICOUNT=0
IST=1
GO TO 5
END

```

NETBAS1.FR

```

-----
C*****
C***** NETBAS SUBROUTINE # 1
C***** WRITTEN BY: DAN PROVOST
C***** 5020 FEDERAL OFFICE BLDG
C***** CINCINNATI, OHIO 45202
C***** FTS 684-2371
C SUBROUTINE CHECKS FILE STATUS
C-----

```

```

SUBROUTINE STATUS
DIMENSION IBUF(7),ISTRG1(6),ISTRG2(6),IBUF2(22)
ISTRG1(1)="PU"
ISTRG1(2)="PF"
ISTRG1(3)="L."
ISTRG1(4)="FM"
ISTRG1(5)=" "
ISTRG1(6)=" "
IBS=0
IFL=0
CALL STAT("PUP.FL",IBUF2,IFL) ; CHECKS CONTROL FILE PRESENCE
IF(IFL.EQ.1)GO TO 30
WRITE(10)"FILE DOES NOT EXIST: PUP.FL "
10 WRITE(10)"ENTER CCCNNNXXX000 OF YOUR CHOICE (000=ADDRESSEE) "
WRITE(10)"EDIT THIS CCCNNNXXX AND FILL IN THE INFORMATION "
WRITE(10)"?"
READ(11,999)ISTRG2
999 FORMAT(6A2)
15 WRITE(10)"AFTER EDITING THIS CCCNNNXXX EXECUTE ON AFOS:"
CALL GDIR(IBUF,IER)
CALL PUP(ISTRG1,ISTRG2)
WRITE(10,998)(ISTRG2(LLL),LLL=1,5),(IBUF(I),I=1,5),(IBUF(I),I=1,5)

```



```

998 FORMAT(' SAVE:',4A2,A1,1X,5A2,' :PUPPY.TS',/, ' RUN:',5A2,' :NETBAS' /)
STOP
30 CALL STAT("PUP.BS",IBUF2,IBS)
IF( IBS.EQ.1)GO TO 40
WRITE(10)"FILE DOES NOT EXIST: PUP.BS "
ISTRG1(2)="PB"
ISTRG1(3)="S. "
CALL GCHN(ICHN1,IER)
CALL OPEN(ICHN1,"PUP.FL",0,IER)
32 READ(ICHN1,997,END=35)ICODE,ISTRG2
997 FORMAT(I2,2X,6A2)
IF(ICODE.NE.80)GO TO 32
GO TO 15
35 WRITE(10)"NO CODE FOUND IN CONTROL FILE..."
GO TO 10
40 RETURN
END

```

NETBAS2.FR

```

C-----
C*****
C***** NETBAS SUBROUTINE # 2
C***** WRITTEN BY: DAN PROVOST
C***** 5020 FEDERAL OFFICE BLDG
C***** CINCINNATI, OHIO 45202
C***** FTS 684-2371
C SUBROUTINE INCORPORATES CORRECTION FROM AFOS CCCNNNXXX
C STORED AS PUPPY.TS; UPDATES THESE CHANGES INTO RDO5 FILES SIMILAR
C TO THEIR CORRESPONDING CCCNNNXXX VERSIONS
C-----
SUBROUTINE CORREC(IDONT)
DIMENSION IBAS(50),IBUF(39),ISTA(35),ISTRG2(6),IBUF2(36)
COMMON/ISTPUP/ISTRG1(6)
DATA ISTRG1/"PUPPY.01"/
IDONT=0 ; DONT UPDATE PUP.ST
CALL STAT("PUPPY.TS",IBUF,ITS)
IF(ITS.NE.1)RETURN
IBASIN="88" ; BEGINNING CODES
IFILES="77"
IEND="99"
CALL GCHN(ICHN1,IER)
CALL OPEN(ICHN1,"PUPPY.TS",0,IER)
CALL GCHN(ICHN2,IER)
CALL OPEN(ICHN2,"PUPPY.01",0,IER) ; REWRITE PUPPY.TS RENUMBERED
READ(ICHN1,999,END=11)ISTRG2 ; IDENTIFIES CCCNNNXXX OF ORIGIN
999 FORMAT(6A2)
WRITE(ICHN2,990)
990 FORMAT(1X,/)
12 READ(ICHN1,998,END=11)IBUF
998 FORMAT(39A2)
WRITE(ICHN2,997)IBUF
997 FORMAT(1X,39A2)
IF( IBUF(1).NE.IBASIN.AND.IBUF(1).NE.IFILES)GO TO 12 ; CODE NOT YET FOUND
IF( IBUF(1).EQ.IFILES) GO TO 80

```



```

CALL GCHN(ICHN3, IER)
CALL FOPEN(ICHN3, "PUP.BS", "B", 81)
DO 15 I=1, 51 ; 50 BASINS MAXIMUM
READ(ICHN1, 995, END=16) IBAS(I), IBUF
IF( IBAS(I).EQ.0) IDONT=1
995 FORMAT(I2, 39A2)
IJJ=I-1
IF( IBAS(I).EQ.89) IJJ=89
IF( IBAS(I).EQ.99) IJJ=99
WRITE(ICHN2, 994) IJJ, IBUF
994 FORMAT(1X, I2, 39A2)
IF( IBAS(I).EQ.99) GO TO 16
WRITE(ICHN3, 994) IJJ, IBUF
15 CONTINUE
16 CALL CLOSE(ICHN1)
CALL CLOSE(ICHN2)
CALL CLOSE(ICHN3)
CALL PUP(ISTRG1, ISTRG2)
CALL STAT("PUP.FL", IBUF, IER)
IF( IER.NE.1) RETURN
CALL STAT("PUP.ST", IBUF, IER)
IF( IER.NE.1) GO TO 87
IF( IDONT.EQ.0) RETURN

```

C\*\*\*----- REWRITES DOWNSTREAM LISTING IN NEW ORDER

```

CALL GCHN(ICHN1, IER)
CALL OPEN(ICHN1, "PUP.FL", 0, IER)
30 READ(ICHN1, 993, END=56) ICODE, ISTRG2
993 FORMAT(I2, 2X, 6A2)
IF( ICODE.NE.81) GO TO 30 ; READS CCCNNNXXX FOR STATION LIST
ICOUNT=0
IBS=0
J=1
IF( IBAS(1).EQ.89) J=J+1
IF( IBAS(J).EQ.0) IBS=1
ICODE=1
CALL GCHN(ICHN4, IER)
CALL OPEN(ICHN4, "PUP.ST", 0, IER)
CALL GCHN(ICHN5, IER)
CALL OPEN(ICHN5, "PUP2.ST", 0, IER)
READ(ICHN4, 987, END=51) IBUF2
WRITE(ICHN5, 989) IBUF2
987 FORMAT(36A2)
989 FORMAT(1X, 36A2)
DO 50 I=1, 50
DO 55 II=1, 1500
READ(ICHN4, 992, END=52) ISTA
992 FORMAT(I3, 3A2, I2, 1X, 30A2)
IF( ISTA(5).NE.ICODE) GO TO 51
ISTA(5)=ISTA(5)+IBS
WRITE(ICHN5, 991) ISTA
55 CONTINUE
51 J=J+1
IF( IBAS(J).EQ.0) IBS=IBS+1
ISTA(5)=ISTA(5)+IBS
WRITE(ICHN5, 991) ISTA
ICODE=ICODE+1
50 CONTINUE
991 FORMAT(1X, I3, 3A2, I2, 1X, 30A2)

```



```

52 CALL CLOSE(ICHN4)
   CALL CLOSE(ICHN5)
   CALL DELETE("PUP.ST")
   CALL RENAM("PUP2.ST", "PUP.ST", IER)
   ISTRG1(4)="ST"
   CALL PUP(ISTRG1, ISTRG2)
   GO TO 87
56 WRITE(10)"NO CCCNNNXXX FOUND FOR CODE 81 IN PUP.FL"
   WRITE(10)"FOR ASSISTANCE PLEASE SEE DOCUMENTATION"
   WRITE(10)"UNABLE TO STORE STATION LIST IN AFOS..."
   STOP "PROGRAM ABORTING"
11 WRITE(10)"NO PUPPY.TS VALID CODE"
   STOP "PROGRAM ABORTING"
30 CALL GCHN(ICHN5, IER)
   CALL OPEN(ICHN5, "PUP.FL", 0, IER)
   WRITE(ICHN5, 997)IBUF
   DO 85 II=1, 100
   READ(ICHN1, 998, END=86)IBUF
   IF(IBUF(1).EQ.IEND)GO TO 86
   WRITE(ICHN5, 997)IBUF
85 CONTINUE
86 CALL CLOSE(ICHN1)
   CALL CLOSE(ICHN2)
   CALL CLOSE(ICHN5) ; CHANNEL 2 NOT NEEDED FOR PUP.FL
87 CALL DELETE("PUPPY.TS")
   CALL DELETE("PUPPY.01")
   RETURN
   END

```



```

C-----
C***** PUPPY PROGRAM DAN PROVOST
C***** MAIN PGM NWS RIVER FORECAST CENTER
C***** 5020 FEDERAL OFFICE BLDG
C***** CINCINNATI, OHIO 45202
C PROGRAM ALLOWS FREE AND TAGGED FORMATTED DATA ENTRY INTO AFOS
C OUTPUT YIELDS PLT FORMAT AND WX WIRE MSGS
C RLDR PUPPY [PUP7,PUP1,PUP4,PUP3,PUP10,PUP13] PUP<2 5 6 8 9 11 12 16> TRNS
C <UTIL BG FORT>.LB 17/C
C-----

```

```

COMMON INET(1500,3), ILCTN(1500,3), ITIME(9), IDATE(3), ICHN4
COMMON ICHN2, ICODE, IDAY(3), ISTA, IREC, IBASN(25), ID(3), IFND4
COMMON ISNOW, IWTRE, ICTL
EXTERNAL OV1, OV2, OV4, OV5, OV6, OV3, OV7
EXTERNAL LOADNT, ADMCK, LOADST, LOADIN, LOADYS, PLTFM, WEATHR
DIMENSION IBUF(15), ISTRG2(6), IWXR(9), IRAWC(13)
IBLANK=" "
WRITE(10)"**PUPPY-000: PUPPY PROGRAM EXECUTING!"
CALL GCHN(ICHN8, IER)
CALL OVOPN(ICHN8, "PUPPY.OL", IER)
CALL OVLOD(ICHN8, OV1, 1, IER)
CALL LOADNT
CALL FOURL(OV1, IER)
CALL OVLOD(ICHN8, OV2, 1, IER)
CALL ADMCK
CALL FOURL(OV2, IER)
CALL OVLOD(ICHN8, OV3, 1, IER)
CALL LOADST
CALL FOURL(OV3, IER)
CALL OVLOD(ICHN8, OV4, 1, IER)
CALL LOADIN
CALL FOURL(OV4, IER)
CALL LOADYS
CALL RESET
CALL GCHN(ICHN8, IER)
CALL OVOPN(ICHN8, "PUPPY.OL", IER)
CALL OVLOD(ICHN8, OV6, 1, IER)
CALL PLTFM
CALL FOURL(OV6, IER)
CALL GCHN(ICHN, IER)
CALL OPEN(ICHN, "PUP.ST", 0, IER)
READ(ICHN, 999, END=5, ERR=5)IWXR
IF(ICODE.EQ.4.OR.ICODE.EQ.5)IWXR(3)=0 ; SUPPRESS CHANGES
999 FORMAT(10X, 9I1)
CALL RESET
GO TO 6
5 WRITE(10)"**PUPPY-001: NO WEATHER WIRE SELECTIONS AVAILABLE!"
GO TO 10
6 CALL GCHN(ICH10, IER)
CALL OPEN(ICH10, "PUP.FL", 0, IER)
7 READ(ICH10, 997, END=8, ERR=8)ICD, IRAWC
997 FORMAT(I2, 2X, 13A1, 61X)
IF(ICD.NE.86)GO TO 7
REWIND ICH10
GO TO 9
8 DO 11 I=1, 13
IRAWC(I)=IBLANK
11 CONTINUE

```



```

REWIND ICH10
9 READ( ICH10, 998, END=10 ) ICD, ISTRG2, IBUF
998 FORMAT( I2, 2X, 6A2, 2X, 15( I2, 1X ), 15X )
IF( ICD.NE.78 ) GO TO 9
CALL WEATHR( ISTRG2, IBUF, IWXR, IRAWC, IHR, IMIN )
GO TO 9
10 REWIND ICH10
CALL GCHN( ICHN8, IER )
CALL OVOPN( ICHN8, "PUPPY.OL", IER )
12 READ( ICH10, 998, END=15 ) ICD, ISTRG2
IF( ICD.NE.84 ) GO TO 12
CALL OVLOD( ICHN8, OV7, 1, IER )
CALL RFC( ISTRG2, IRAWC, IHR, IMIN )
CALL FOURL( OV7 )
CALL CLOSE( ICH10 )
WRITE( 10 ) "**PUPPY-002: PROGRAM COMPLETED!"
STOP
15 WRITE( 10 ) "**PUPPY-003: NO RFC CODE: PROGRAM COMPLETED!"
CALL CLOSE( ICH10 )
STOP
END

```

```

-----
C***** PUPPY PROGRAM DAN PROVOST
C***** PUP1.FR NWS RIVER FORECAST CENTER
C***** 5020 FEDERAL OFFICE BLDG
C***** CINCINNATI, OHIO 45202

```

```

C SUBROUTINE ADMCK: CHECKS ADM ENTRY FOR CORRECT PROGRAM DATE
C AND PROGRAM OPTION 1,2,3,4 OR 5

```

```

-----
C***** PUP SUBROUTINE #1 D PROVOST
OVERLAY OV2
SUBROUTINE ADMCK
COMMON INET( 1500, 3 ), ILCTN( 1500, 3 ), ITIME( 9 ), LDATE( 3 ), ICHN4
COMMON ICHN2, ICODE, IDAY( 3 ), ISTA, IREC, IBASN( 25 ), ID( 3 ), IFND4
COMMON ISNOW, IWTRE, ICTL
DIMENSION IBUF( 39 ), ISTN( 4 ), JDATE( 3 ), KDATE( 3 ), IDATE( 3 )
DO 5 I=1, 1500 ; BLANK OUT ARRAY
DO 6 K=1, 3
ILCTN( I, K )=0
6 CONTINUE
5 CONTINUE
JTIM1="TI"
JTIM2="ME"
IRFC="RR"
ICMT="CC"
IFND4=0 ; PUP.DT RECORD COUNTER
ICTL=0 ; PUP.CT RECORD COUNTER
CALL DATE( IDAY, IER )

```



```

IDAY(3)=IDAY(3)-((IDAY(3)/100)*100) ; CONVERT 1981 TO 81
CALL STAT("PUPPY.IN",IBUF,IIN)
CALL STAT("PUP.DT",IBUF,IDT)
CALL STAT("PUP.YS",IBUF,IYS)
CALL STAT("PUP.CT",IBUF,ICT)
IF(IIN.EQ.1)GO TO 20
WRITE(10)**PUPPY-101: PUPPY.IN INPUT FILE DOES NOT EXIST!"
GO TO 900
20 CALL GCHN(ICHN2,IER)
CALL OPEN(ICHN2,"PUPPY.IN",0,IER)
21 READ(ICHN2,997,END=22,ERR=22)ITIM1,ITIM2
ISTA=ISTA+1
997 FORMAT(8X,2A2)
IF(ITIM1.NE.JTIM1.OR.ITIM2.NE.JTIM2)GO TO 21
READ(ICHN2,996,END=22)ITIME(1),IDATE,(ITIME(J),J=2,9),
1LDATE,ICODE
17 IF(ICODE.GE.1.AND.ICODE.LE.5)GO TO 18
WRITE(10)**PUPPY-102: INVALID PROGRAM OPTION!"
GO TO 900
996 FORMAT(8X,I4,3X,I2,3X,I2,3X,I2,2X,8A2,2X,I2,3X,
1I2,3X,I2,8X,I1)
C FILE ICHN2-INPUT WILL REMAIN OPEN !!!
18 IF(IDATE(1).EQ.IDAY(1).AND.IDATE(2).EQ.IDAY(2))GO TO 23
WRITE(10)**PUPPY-103: WRONG DATE IN INPUT FILE!"
CALL CLOSE(ICHN2)
GO TO 900
23 IF(ITIME(1).GE.1200.AND.ITIME(1).LT.1800.AND.ICODE.LE.3)GO TO 28
IF(ICODE.GE.4)GO TO 26
WRITE(10)**PUPPY-104: NEW DAY (12Z) OPTION ;OFF-TIME DATA!"
GO TO 900
26 IF(ITIME(1).LT.1200.OR.ITIME(1).GE.1800)GO TO 28 ; VALID OFF-TIME
IF(ICODE.EQ.5)GO TO 28 ; CORRECTION CODE
WRITE(10)**PUPPY-105: OFF-TIME OPTION ;12Z DATA!"
WRITE(10)**PUPPY-106: OPTION CORRECTED TO 12Z DATA!"
ICODE=1
28 IF(ICODE.LE.3)ITIME(1)=1200
CALL GCHN(ICHN4,IER)
CALL FOPEN(ICHN4,"PUP.DT","B",17)
CALL FSEEK(ICHN4,0)
IF(IDT.EQ.1)GO TO 30
29 WRITE(ICHN4,994)IDATE,ITIME(1)
CALL FSEEK(ICHN4,0)
30 READ(ICHN4,995,END=29)JDATE,JTIME
IFND4=IFND4+1
IF(IYS.EQ.0)CALL CRAND("PUP.YS")
995 FORMAT(3(I2,1X),1X,I4,2X)
CALL JULN(IDAY,IJUL)
CALL JULN(JDATE,JJUL)
IF(IJUL.NE.1)GO TO 51
IJUL=366
IYR=(IDAY(3)-1)/4
YR=IDAY(3)
YR=((YR-1)/4)-IYR
IF(YR.EQ.(0.))IJUL=367
51 GO TO (100,200,300,400,500),ICODE
GO TO 17 ; INVALID CODE AT THIS POINT
100 ITEST=IJUL-JJUL
IF(ITEST.EQ.0.AND.JTIME.EQ.1200)GO TO 300
IF(ITEST.NE.1.OR.JTIME.NE.1200)GO TO 110

```



```

CALL CLOSE(ICHN4)
CALL DELETE("PUP.YS")
CALL RENAM("PUP.DT","PUP.YS",IER)
IF(ICT.EQ.1)CALL DELETE("PUP.CT")
CALL GCHN(ICHN4,IER)
CALL FOPEN(ICHN4,"PUP.DT","B",17)
GO TO 111
110 CALL CLOSE(ICHN4) ;INDICATES .DT NOT YESTERDAY
CALL DELETE("PUP.DT"); OR .DT IS NOT 12Z DATA
IF(ICT.EQ.1)CALL DELETE("PUP.CT")
CALL GCHN(ICHN4,IER)
CALL FOPEN(ICHN4,"PUP.DT","B",17)
CALL FSEEK(ICHN4,0)
111 WRITE(ICHN4,994)IDAY,ITIME(1)
IFND4=1
GO TO 800
200 ITEST=IJUL-JJUL
IF(ITEST.NE.0.OR.JTIME.NE.1200)GO TO 100
DO 210 I=1,2000
CALL FSEEK(ICHN4,IFND4)
READ(ICHN4,993,END=215)ID,IFND
IF(ID(1).EQ.IRFC.AND.ID(2).EQ.IRFC)ICTL=ICTL+1
IF(ID(1).EQ.IRFC.AND.ID(2).EQ.IRFC)IFND4=IFND4+1
IF(ID(1).EQ.IRFC.AND.ID(2).EQ.IRFC)GO TO 210
IF(ID(1).EQ.ICMT.AND.ID(2).EQ.ICMT)ICTL=ICTL+1
IF(ID(1).EQ.ICMT.AND.ID(2).EQ.ICMT)IFND4=IFND4+1
IF(ID(1).EQ.ICMT.AND.ID(2).EQ.ICMT)GO TO 210
993 FORMAT(3A2,2X,I4,4X)
CALL BINSR(ICANT,ILOC)
IF(ICANT.EQ.0)GO TO 209
ILCTN(ILOC,1)=IFND4; BEGINNING OF RECORD= IFND4
IFND4=(IFND/2)+IFND4+1
GO TO 210
209 WRITE(10,992)ID
IFND4=(IFND/2)+IFND4+1
992 FORMAT(1X,'**PUPPY-107: ',3A2,'NO LONGER IN MASTER!')
210 CONTINUE
215 GO TO 800
300 ITEST=IJUL-JJUL
IF(ITEST.NE.0.OR.JTIME.NE.1200)GO TO 100
IF(ICODE.EQ.1)WRITE(10)**PUPPY-108: USING OPTION 3."
DO 310 I=1,2000
CALL FSEEK(ICHN4,IFND4)
READ(ICHN4,993,END=315)ID,IFND
IF(ID(1).EQ.IRFC.AND.ID(2).EQ.IRFC)ICTL=ICTL+1
IF(ID(1).EQ.IRFC.AND.ID(2).EQ.IRFC)IFND4=IFND4+1
IF(ID(1).EQ.IRFC.AND.ID(2).EQ.IRFC)GO TO 310
IF(ID(1).EQ.ICMT.AND.ID(2).EQ.ICMT)ICTL=ICTL+1
IF(ID(1).EQ.ICMT.AND.ID(2).EQ.ICMT)IFND4=IFND4+1
IF(ID(1).EQ.ICMT.AND.ID(2).EQ.ICMT)GO TO 310
IFND4=(IFND/2)+1+IFND4
CALL BINSR(ICANT,ILOC)
IF(ICANT.EQ.1)GO TO 310
WRITE(10,992)ID
310 CONTINUE
315 GO TO 800
400 IF(JTIME.NE.1200)GO TO 410
CALL CLOSE(ICHN4)
CALL DELETE("PUP.YS")

```



```

CALL RENAM("PUP.DT", "PUP.YS", IER)
IF(ICT.EQ.1)CALL DELETE("PUP.CT")
CALL GCHN(ICHN4, IER)
CALL FOPEN(ICHN4, "PUP.DT", "B", 17)
CALL FSEEK(ICHN4, 0)
GO TO 411
410 CALL CLOSE(ICHN4)
CALL DELETE("PUP.DT")
IF(ICT.EQ.1)CALL DELETE("PUP.CT")
CALL GCHN(ICHN4, IER)
CALL FOPEN(ICHN4, "PUP.DT", "B", 17)
CALL FSEEK(ICHN4, 0)
411 WRITE(ICHN4, 994)IDAY, ITIME(1)
IFND4=1
GO TO 800
500 JJUL=IJUL
JTIME=1200
GO TO 300
800 CALL GCHN(ICHN5, IER)
CALL FOPEN(ICHN5, "PUP.YS", "B", 17)
CALL FSEEK(ICHN5, 0)
READ(ICHN5, 995, END=801)KDATE, KTIME
CALL JULN(KDATE, KJUL)
IF(KJUL.NE.1)GO TO 802
KJUL=366
KYR=(KDATE(3)-1)/4
YR=KDATE(3)
YR=((YR-1)/4)-IYR
IF(YR.EQ.(0.))KJUL=367
802 ITEST=IJUL-KJUL
IF(ITEST.NE.1.OR.KTIME.NE.1200)GO TO 801
CALL CLOSE(ICHN5) ; ACTUAL PREVIOUS DAYS FILE
RETURN
801 CALL CLOSE(ICHN5)
IF(ICODE.EQ.4.AND.KTIME.EQ.1200.AND.ITEST.EQ.0)RETURN
CALL DELETE("PUP.YS")
RETURN
994 FORMAT(1X, 3(I2, 1X), 1X, I4, 2X)
22 WRITE(10)**PUPPY-109: TIME STATEMENT NOT IN INPUT FILE!
CALL CLOSE(ICHN2)
900 WRITE(10)**PUPPY-110: PROGRAM NOT EXECUTED!
STOP
END

```



```

C-----
C***** PUPPY PROGRAM DAN PROVOST
C***** PUP2.FR NWS RIVER FORECAST CENTER
C***** 5020 FEDERAL OFFICE BLDG
C***** CINCINNATI, OHIO 45202

```

```

C SUBROUTINE BINSR: PERFORMS BINARY SEARCH OF MASTER NETWORK
C FILE (NETWORK.DT) TO LOCATE STATION
C-----

```

```

C***** NETWORK SUBROUTINE BINARY SEARCH
C***** DAN PROVOST-CINCINNATI RFC

```

```

SUBROUTINE BINSR(ICANT,ILOC)
COMMON INET(1500,3),ILCTN(1500,3),ITIME(9),IDATE(3),ICHN4
COMMON ICHN2,ICODE,IDAY(3),ISTA,IREC,IBASN(25),KD(3),IFND4
COMMON ISNOW,IWTRE,ICTL
ILOC=0 ; LOCATION OF LAST ID CHECKED OR MATCHED
ICANT=0 ; ASSUME NO MATCH UNTIL FOUND
ILO=0 ; "AA" IS LESS THAN "AB"
IHI=IREC+1
10 IAM=(ILO+IHI)/2
IF(IAM-ILO)95,95,20
20 IF(KD(1)-INET(IAM,1))30,60,40
30 IHI=IAM ;SEARCH IS FOR LOWER HALF
GO TO 10
40 ILO=IAM ;SEARCH IS FOR UPPER HALF
GO TO 10
60 CONTINUE
IF(INET(IAM,1).NE.KD(1))GO TO 95
DO 65 NELT=2,3
IF(KD(NELT)-INET(IAM,NELT))70,65,80
65 CONTINUE
GO TO 90
70 IAM=IAM-1
IF(IAM.EQ.0)GO TO 100
NELT=2
IF(INET(IAM,NELT).LT.KD(NELT))GO TO 95
GO TO 60
80 IAM=IAM+1
IF(IAM.GT.IREC)GO TO 98
NELT=2
IF(INET(IAM,NELT).GT.KD(NELT))GO TO 95
GO TO 60
90 CONTINUE
ILOC=IAM
ICANT=1 ; MATCH OF ID FOUND
RETURN
95 IF(IAM.EQ.0)GO TO 100
DO 99 KK=1,3
IF(KD(KK)-INET(IAM,KK))98,99,100
99 CONTINUE
98 IAM=IAM-1
100 ILOC=IAM
RETURN
END

```



```

-----
C***** PUPPY PROGRAM
C***** DAN PROVOST
C***** PUP3.FR NWS RIVER FORECAST CENTER
C***** 5020 FEDERAL OFFICE BLDG
C***** CINCINNATI, OHIO 45202

```

```

C
C SUBROUTINE LOADST: LOADS DOWNSTREAM STATION LISTING AND STORES
C LOCATION IN ARRAY ILCTN(ILOC,2)
C

```

```

-----
C***** PUP SUBROUTINE #3 D PROVOST

```

```

OVERLAY 003
SUBROUTINE LOADST
COMMON INET(1500,3), ILCTN(1500,3), ITIME(9), IDATE(3), ICHN4
COMMON ICHN2, ICODE, IDAY(3), ISTA, IREC, IBASN(25), ID(3), IFND4
COMMON ISNOW, IWTRE, ICTL
CALL GCHN(ICHN1, IER)
CALL OPEN(ICHN1, "PUP.ST", 0, IER)
READ(ICHN1, 999, END=10) ID, ISTN
ISTART=0
DO 5 I=1, 1500
READ(ICHN1, 999, END=10) ID, ISTN
999 FORMAT(3X, 3A2, I2)
IF(ISTN.EQ. ISTART) GO TO 7
IBASN(ISTN)=I
ISTART=ISTN
7 CALL BINSR(ICANT, ILOC)
IF(ICANT.EQ. 1) GO TO 6
WRITE(10, 998) ID, ISTN
998 FORMAT(1X, '**PUPPY-301: ', 3A2, ' DOES NOT EXIST IN NETWORK!')
GO TO 5
6 ILCTN(ILOC, 2)=I
5 CONTINUE
10 CALL CLOSE(ICHN1)
RETURN
END

```



```

C-----
C***** PUPPY PROGRAM
C***** DAN PROVOST
C***** PUP4.FR NWS RIVER FORECAST CENTER
C***** 5020 FEDERAL OFFICE BLDG
C***** CINCINNATI, OHIO 45202

```

```

C SUBROUTINE LOADIN: READS INPUT FILE PUPPY.IN; DETERMINES RECORD FORMAT
C THRU PUP6.FR AND PUP9.FR SUBROUTINES
C-----

```

```

OVERLAY OV4

```

```

SUBROUTINE LOADIN ; PUP4.FR

```

```

COMMON INET(1500,3), ILCTN(1500,3), ITIME(9), IDATE(3), ICHN4

```

```

COMMON ICHN2, ICODE, IDAY(3), ISTA, IREC, IBASN(25), ID(3), IFND4

```

```

COMMON ISNOW, IWTRE, ICTL

```

```

DIMENSION IBUF(31), IFT(500), ISTN(74), ISTRG2(6), ISTRG1(6)

```

```

DIMENSION ICH(3), IRS(3)

```

```

KCMT=ICTL

```

```

ICONT=0

```

```

IFORM=0

```

```

ICLINE=0

```

```

K=0

```

```

IP01=0

```

```

ISTART=0

```

```

ISTR="XX" ; CALL SIGN FOR STRANGER STATION

```

```

ICMT="CC" ; CALL SIGN FOR WX WIRE COMMENT LINE

```

```

INFO="II" ; CALL SIGN FOR IGNORED PRE-FORMAT LINE

```

```

IBLK=" " ; CALL SIGN FOR DATA CONTINUATION LINE

```

```

IEND="NN" ; CALL SIGN FOR END OF DATA

```

```

IRFC="RR" ; CALL SIGN FOR COMMENT LINE TO RFC ONLY

```

```

CALL GCHN(ICHN7, IER)

```

```

CALL OPEN(ICHN7, "FORMAT.DT", 0, IER)

```

```

CALL GCHN(ICHN8, IER)

```

```

CALL FOPEN(ICHN8, "PUP.CT", "B", 75)

```

```

CALL FSEEK(ICHN8, ICTL)

```

```

CALL GCHN(ICHN9, IER)

```

```

CALL FOPEN(ICHN9, "PUP.ST", "B", 73) ; DOWNSTREAM + ELEM FILE

```

```

ICHN2-- INPUT DATA FILE IS ALREADY OPEN

```

```

DO 110 I=1,2000 ; UP TO 2000 LINES OF INPUT

```

```

CALL CHSAV(ICHN2, ICH)

```

```

READ(ICHN2, 999, END=112) ID, ISTN

```

```

999 FORMAT(1X, 3A2, 74A1)

```

```

IF(ID(1).EQ.IEND.AND.ID(2).EQ.IEND)GO TO 80; END

```

```

IF(ID(1).EQ.INFO.AND.ID(2).EQ.INFO)GO TO 110; IGNORE LINE

```

```

IF(ID(1).EQ.IRFC.AND.ID(2).EQ.IRFC)GO TO 40 ; COMMENT TO RFC ONLY

```

```

IF(ID(1).EQ.ICMT.AND.ID(2).EQ.ICMT)GO TO 40 ; COMMENT TO WX WIRE AND RFC

```

```

IF(ID(1).EQ.IBLK.AND.ID(2).EQ.IBLK)GO TO 50 ; DATA CONTINUATION LINE

```

```

IF(ID(1).EQ.ISTR.AND.ID(2).EQ.ISTR)GO TO 60 ; STRANGER STATION

```

```

IF(ISTART.EQ.1)GO TO 80

```

```

CALL BINSR(ICANT, ILOC)

```

```

IF(ICANT.EQ.0)GO TO 83 ; INVALID CALL SIGN

```

```

ISTART=1

```

```

ICONT=0

```

```

ICLINE=0

```

```

IFIELD=0

```

```

IFREE=0

```

```

CALL FORMT(ISTN, K, IFREE, IFT, ICHN1, ICHN2, IFIELD, IFORM)

```

```

IF(IFREE.EQ.2)GO TO 91

```



```

DO 10 II=1,3
IRS(II)=ICH(II) ; SAVE CHANNEL INFO
IBUF(II)=ID(II) ; SAVE CURRENT CALL SIGN
10 CONTINUE
GO TO 110
40 IF(ISTART.EQ.0.AND.ICLINE.EQ.0)GO TO 85 ; UNASSIGNED COMMENT LINE
IF(IFREE.EQ.2)GO TO 110 ; ASSIGNED PREVIOUS RECORD DATA NO GOOD
IF(ICLINE.EQ.1)GO TO 41
CALL ENDREC(ICHN7,IFREE,ISTN,K,IFIELD,IFT,IRS,IBUF,ILOC,ICHN9)
IFIELD=0
IFORM=0
ICLINE=1
GO TO 110 ; WILL REREAD COMMENT LINE
41 KCMT=KCMT+1
WRITE(ICHN8,996)ISTN
WRITE(ICHN4,995)ID,KCMT ; WRITES FILE LOCATION TO PUP.DT
IFND4=IFND4+1
ISTART=0
GO TO 110
50 IF(ISTART.EQ.0)GO TO 87 ; UNASSIGNED CONTINUATION LINE
IF(IFREE.EQ.2)GO TO 110
ICONT=1
CALL FORMT(ISTN,K,IFREE,IFT,ICHN1,ICHN2,IFIELD,IFORM)
IF(IFREE.EQ.2)GO TO 89
GO TO 110
60 IF(ISTART.EQ.1)GO TO 80
CALL STRANGER(ICHN5,IP01,ISTN)
ICLINE=0
ISTART=0
GO TO 110
80 IF(ISTART.EQ.0.OR.ICLINE.EQ.1)GO TO 81
IF(IFIELD.EQ.0)CALL CHRST(ICHN2,ICH) ; ENDREC RETURNS IMMEDIATELY
CALL ENDREC(ICHN7,IFREE,ISTN,K,IFIELD,IFT,IRS,IBUF,ILOC,ICHN9)
81 IF(ID<1>.EQ.IEND.AND.ID<2>.EQ.IEND)GO TO 112
IFORM=0
ISTART=0
GO TO 110
83 WRITE(10)**PUPPY-401: UNKNOWN STATION ID- "
WRITE(10,992)ID,ISTN
ICLINE=0
ISTART=0
GO TO 110
85 WRITE(10)**PUPPY-402: COMMENT LINE;NO VALID CALL SIGN-"
WRITE(10,992)ID,ISTN
GO TO 110
87 WRITE(10)**PUPPY-403: CONTINUATION LINE;NO VALID CALL SIGN-"
WRITE(10,992)ID,ISTN
GO TO 110
89 IF(ICONT.EQ.0)GO TO 91
WRITE(10,991)(IBUF(L),L=1,3)
GO TO 92
91 WRITE(10,991)ID
92 WRITE(10,992)ID,ISTN
IFORM=0
IFIELD=0
110 CONTINUE
996 FORMAT(1X,74A1)
995 FORMAT(1X,3A2,2X,I4,4X)
992 FORMAT(1X,3A2,74A1)

```



```

991 FORMAT(1X, '**PUPPY-404: BAD CHARACTER IN STATION- ',3A2, 'DATA:')
112 IF(IP01.EQ.0)GO TO 113
WRITE(ICHN5,990)
990 FORMAT(////////,1X, 'NNNN')
CALL CLOSE(ICHN5)
CALL GCHN(ICHN6,IER)
CALL OPEN(ICHN6,"PUP.FL",0,IER)
DO 35 II=1,50
READ(ICHN6,997,END=31)IBF,ISTRG2
IF(IBF.EQ.83)GO TO 32 ; 83=STRANGER FILE CODE
35 CONTINUE
31 WRITE(10)**PUPPY-405: NO CCCNNNXXX (CODE 83) IN CONTROL FILE."
WRITE(10)**PUPPY-406: FILE PUP.01 IS STILL AVAILABLE IN ROOS."
CALL CLOSE(ICHN6)
RETURN
32 ISTRG1(1)="PU"
ISTRG1(2)="P."
ISTRG1(3)="01"
ISTRG1(4)=" "
ISTRG1(5)=" "
ISTRG1(6)=" "
CALL PUP(ISTRG1,ISTRG2)
113 CALL RESET
RETURN
997 FORMAT(I2,2X,30A2)
END

```

```

C-----
C***** PUPPY PROGRAM
C***** DAN PROVOST
C***** PUP5.FR NWS RIVER FORECAST CENTER
C***** 5020 FEDERAL OFFICE BLDG
C***** CINCINNATI, OHIO 45202

```

```

C
C SUBROUTINE LOADYS: LOADS YESTERDAYS DATA LOCATION FROM PUP.YS INTO
C ARRAY ILCTN(ILOC,3); USED IF COMPUTING 24 HR CHG.
C
C-----

```

```

C***** PUP SUBROUTINE #5 D PROVOST
C***** LOAD YESTERDAY'S DATA
SUBROUTINE LOADYS
COMMON INET(1500,3),ILCTN(1500,3),ITIME(9),IDATE(3),ICHN4
COMMON ICHN2,ICODE,IDAY(3),ISTA,IREC,IBASN(25),ID(3),IFND4
COMMON ISNOW,IWTRE,ICTL
IF(ICODE.EQ.4.OR.ICODE.EQ.5)RETURN
CALL GCHN(ICHN1,IER)
CALL FOPEN(ICHN1,"PUP.YS","B",17)
IRFC="RR" ; CALL SIGN FOR COMMENT LINE TO RFC
ICMT="CC" ; CALL SIGN FOR COMMENT LINE TO WX WIRE AND RFC
IFIND=1
DO 10 I=1,10000
CALL FSEEK(ICHN1,IFIND)

```



```

READ(ICHN1,999,END=15)ID,INBR
IF(ID(1).EQ.IRFC.AND.ID(2).EQ.IRFC)IFIND=IFIND+1
IF(ID(1).EQ.IRFC.AND.ID(2).EQ.IRFC)GO TO 10
IF(ID(1).EQ.ICMT.AND.ID(2).EQ.ICMT)IFIND=IFIND+1
IF(ID(1).EQ.ICMT.AND.ID(2).EQ.ICMT)GO TO 10
INBR=INBR/2
999 FORMAT(3A2,2X,I4,4X)
CALL BINSR(ICANT,ILOC)
IF(ICANT.EQ.1)GO TO 7
WRITE(10,998)ID
998 FORMAT(1X,'**PUPPY-501: ',3A2,'DOES NOT EXIST IN MASTER!')
IFIND=IFIND+INBR+1
GO TO 10
7 ILCTN(ILOC,3)= IFIND
IFIND=IFIND+INBR+1
10 CONTINUE
15 CALL CLOSE(ICHN1)
RETURN
END

```

```

C-----
C***** PUPPY PROGRAM
C***** DAN PROVOST
C***** PUP6.FR NWS RIVER FORECAST CENTER
C***** 5020 FEDERAL OFFICE BLDG
C***** CINCINNATI, OHIO 45202
C
C SUBROUTINE FORMT : DETERMINES THE RECORD FORMAT BY COUNTING CHARACTERS
C AND SPACES; ALSO DETERMINES WHETHER INPUT IS FREE
C OR TAGGED BY SENSING : / , =
C-----

```

```

SUBROUTINE FORMT(ISTN,K,IFREE,IFT,ICHN1,ICHN2,IFIELD,IFORM)
DIMENSION ISTN(74),IFT(500)
COMMON/LLL/IDATA(16)
DATA IDATA/'1 2 3 4 5 6 7 8 9 0 . - , / = "/
IFT(1)="( "
IFT(2)="1 "
IFT(3)="X "
IFT(4)=", "
IFT(5)="3 "
IFT(6)="A "
IFT(7)="2 "
IFT(8)=", "
IFORM=IFORM+1 ; INDICATES # OF RECORDS FORMATTES
IF(IFORM.EQ.1)GO TO 3
IFT(K)="/ "
IFT(K+1)=", "
IFT(K+2)="7 "
IFT(K+3)="X "
IFT(K+4)=", "
K=K+5
3 IF(IFORM.EQ.1)K=9 ; INDICATED FIRST RECORD BEING FORMATTED
IEND=0

```



```

IX=0
ICAR=0
DO 500 I=1,74
DO 6 II=1,16 ; 16 ACCEPTABLE CHARAC IN IDATA
IF(ISTN(I).EQ.IDATA(II))GO TO 7
6 CONTINUE
WRITE(10)**PUPPY-601: INVALID CHARACTER IN RECORD"
IFREE=2
RETURN
7 IF(II.GE.13.AND.II.LE.15)IFREE=1 ; 0=ASSIGNED FORMAT
1=TAGGED FORMAT 2=INVALID FORMAT
IF(ICAR.EQ.0.AND.II.GT.12)GO TO 52 ; BLANK OR INVALID DELIMITER
IF(IX.EQ.0.AND.II.LE.12)GO TO 62; VALID CHARACTER
IF(IX.NE.0.AND.II.LE.12)GO TO 60
50 IFIELD=IFIELD+1
IF(ICAR.LE.9)GO TO 51
ICAR1=ICAR/10
ICAR=ICAR-(ICAR1*10)
IF(ICAR.EQ.0)ICAR=10
IF(ICAR1.EQ.0)ICAR1=10
IFT(K)="F "
IFT(K+1)=IDATA(ICAR1)
IFT(K+2)=IDATA(ICAR)
IFT(K+3)="."
IFT(K+4)="0 "
IFT(K+5)=","
K=K+6
ICAR=0
GO TO 52
51 IFT(K)="F "
IFT(K+1)=IDATA(ICAR)
IFT(K+2)="."
IFT(K+3)="0 "
IFT(K+4)=","
K=K+5
ICAR=0
52 IX=IX+1
GO TO 500
60 IF(IX.LE.9)GO TO 61
IX1=IX/10
IX=IX-(IX1*10)
IF(IX.EQ.0)IX=10
IF(IX1.EQ.0)IX1=10
IFT(K)=IDATA(IX1)
IFT(K+1)=IDATA(IX)
IFT(K+2)="X "
IFT(K+3)=","
K=K+4
IX=0
GO TO 62
61 IFT(K)=IDATA(IX)
IFT(K+1)="X "
IFT(K+2)=","
K=K+3
IX=0
62 ICAR=ICAR+1
500 CONTINUE
IF(IEND.EQ.1)GO TO 503
IEND=1

```



```
IF(II.LE.11)GO TO 50
GO TO 60
503 CONTINUE
RETURN
END
```

```
C-----
C***** PUPPY PROGRAM
C***** DAN PROVOST
C***** PUP7.FR NWS RIVER FORECAST CENTER
C***** 5020 FEDERAL OFFICE BLDG
C***** CINCINNATI, OHIO 45202
C
C SUBROUTINE LOADNT: FIRST SUBROUTINE EXECUTED WHICH LOADS THE NETWORK.DT
C FILE INTO ARRAY INET(1500,3) ; MAXIMUM 1500 STNS
C-----
```

```
OVERLAY OV1
SUBROUTINE LOADNT
COMMON INET(1500,3), ILCTN(1500,3), ITIME(9), IDATE(3), ICHN4
COMMON ICHN2, ICODE, IDAY(3), ISTA, IREC, IBASN(25), ID(3), IFND4
COMMON ISNOW, IWTR, ICTL
IREC=0
K=0
DO 60 I=1,20
60 ILCTN(I)=K
CALL GCHN(ICHN, IER)
CALL OPEN(ICHN, "NETWORK.DT", 0, IER)
READ(ICHN, 999, END=100) AMXLAT, AMNLAT, AMXLON, AMNLON
999 FORMAT(18X, 4(F7.2, 2X))
998 FORMAT(3A2)
DO 80 I=1,1500
READ(ICHN, 998, END=81) (INET(I, J), J=1, 3)
IREC=IREC+1
80 CONTINUE
81 CALL CLOSE(ICHN)
RETURN
100 CALL CLOSE(ICHN)
WRITE(10) "##PUPPY-701: NO MASTER NETWORK FILE!"
WRITE(10) "##PUPPY-702: ATTEMPT FATAL!!!"
STOP
END
```



```

C-----
C***** PUPPY PROGRAM
C***** DAN PROVOST
C***** PUP8.FR NWS RIVER FORECAST CENTER
C***** 5020 FEDERAL OFFICE BLDG
C***** CINCINNATI, OHIO 45202

```

```

C
C SUBROUTINE STRANGER: WRITES STRANGER FILE FROM ANY ENTRY WITH THE
C CALL SIGN: XXXXX
C-----

```

```

SUBROUTINE STRANGER(ICHN5,IP01,ISTN)
COMMON INET(1500,3),ILCTN(1500,3),ITIME(9),IDATE(3),ICHN4
COMMON ICHN2,ICODE,IDAY(3),ISTA,IREC,IBASN(25),ID(3),IFND4
COMMON ISNOW,IWTRE,ICTL
DIMENSION ISTN(74),IBUF(31),IRAWC(13)
IF(IP01.EQ.1)GO TO 15
CALL GCHN(ICHN5,IER)
CALL OPEN(ICHN5,"PUP.01",0,IER)
CALL GCHN(ICHN6,IER)
CALL OPEN(ICHN6,"PUP.FL",0,IER)
5 READ(ICHN6,992,END=6)ICD,IRAWC
992 FORMAT(I2,2X,13A1)
IF(ICD.NE.86)GO TO 5
GO TO 7
6 DO 8 II=1,13
IRAWC(II)=IBLANK
8 CONTINUE
7 CALL CLOSE(ICHN6)
CALL FGTIM(IHR,IMIN,ISEC)
WRITE(ICHN5,993)IRAWC,IDAY(2),IHR,IMIN
993 FORMAT(1X,13A1,3I2)
CALL GCHN(ICHN6,IER)
CALL OPEN(ICHN6,"PUP.BS",0,IER)
DO 16 II=1,50
READ(ICHN6,997,END=17)IBUF
IF(IBUF(1).EQ.89)GO TO 19 ; FOUND 89 CODE(WSFO ID)
16 CONTINUE
17 WRITE(10)"PUPPY-801: UNKNOWN WSFO ID;ASSUMING BLANKS!"
DO 18 II=1,39
IBUF(II)=IBLK
18 CONTINUE
CALL CLOSE(ICHN6)
19 WRITE(ICHN5,996)(IBUF(L),L=2,31),(ITIME(L),L=2,9),IDATE
996 FORMAT(1X,30A2,/,1X,8A2,1X,I2,'/',I2,'/',I2,' MISCELLANEOUS HY
1DROLOGIC OBSERVATIONS',/)
15 WRITE(ICHN5,994)ISTN
997 FORMAT(I2,2X,30A2)
994 FORMAT(1X,74A1)
IP01=1
RETURN
END

```



PUPPY PROGRAM

DAN PROVOST  
 NWS RIVER FORECAST CENTER  
 5020 FEDERAL OFFICE BLDG  
 CINCINNATI, OHIO 45202

PUP9.FR

SUBROUTINE ENDREC: REREADS PREVIOUS RECORDS WHICH WERE FORMATTED  
 BY SUB FORMT

```

SUBROUTINE ENDREC(IC7,IFREE,ISTN,K,IFIELD,IFT,IRS,IBUF,ISR,IC9)
COMMON INET(1500,3),ILCTN(1500,3),ITIME(9),IDATE(3),ICHN4
COMMON ICHN2,ICODE,IDAY(3),ISTA,IREC,IBASN(25),ID(3),IFND4
COMMON ISNOW,IWTRE,ICTL
DIMENSION ISTN(74),ELEM(29),IFT(500),ADATA(58),IRS(3),IBUF(3)
IF(IFIELD.EQ.0)GO TO 73 ; VALID CALL SIGN BUT NO DATA (DONT PROCESS)
IFT(K)=" "
K=K+2 ; CLOSSES FORMAT ARRAY
DO 10 I=K,500
  IFT(I)=" "
10 CONTINUE
  REWIND IC7
  WRITE(IC7,999)IFT
999 FORMAT(1X,80A1)
  REWIND IC7
  K=K/2
  READ(IC7,998,END=71,ERR=71)(IFT(LL),LL=1,K)
998 FORMAT(40A2)
  CALL CHRST(ICHN2,IRS) ; RESTORE INPUT FILE FOR RE-READ
  READ(ICHN2,IFT,END=72,ERR=72)IBUF,(ADATA(KK),KK=1,IFIELD)
  IF(IFREE.EQ.1)GO TO 20
  ISK=(ILCTN(ISR,2))
  CALL FSEEK(IC9,ISK)
  READ(IC9,995,ERR=70,END=70)ID,IBN,IMXEL,ELEM
  IBN2=IFIELD*2
  CALL FSEEK(ICHN4,IFND4)
  WRITE(ICHN4,997)(IBUF(N),N=1,3),IBN2
  WRITE(ICHN4,996)(ELEM(N),ADATA(N),N=1,IFIELD)
  ILCTN(ISR,1)=IFND4
  IFND4=IFND4+IFIELD+1
997 FORMAT(1X,3A2,2X,14,4X)
996 FORMAT(2X,F3.0,',',F11.2)
  IPLT=IPLT+IBN+1
  RETURN
20 JFIELD=(IFIELD/2)
  FIELD=(IFIELD/2)-JFIELD
995 FORMAT(3X,3A2,I2,1X,30F2.0)
  IF(FIELD.NE.(0.))GO TO 26
  DO 25 KK=1,IFIELD,2
  IF(ADATA(KK).GT.29.OR.ADATA(KK).EQ.(0.))GO TO 26
25 CONTINUE
  CALL FSEEK(ICHN4,IFND4)
  WRITE(ICHN4,997)IBUF,IFIELD
  WRITE(ICHN4,996)(ADATA(L),L=1,IFIELD)
  ILCTN(ISR,1)=IFND4
  IFND4=IFND4+1+(IFIELD/2)
  RETURN
26 WRITE(10)**PUPPY-902: UNMATCHED OR INVALID ELEMENT!"
  
```



```

WRITE(10,997)IBUF,IFIELD,(ADATA(KK),KK=1,IFIELD)
RETURN
70 WRITE(10)**PUPPY-906: ERROR IN PUP.ST FILE."
RETURN
71 WRITE(10)**PUPPY-907: ERROR IN FORMAT.DT FILE."
RETURN
72 WRITE(10)**PUPPY-908: ERROR READING INPUT FILE!"
RETURN
73 WRITE(10,994)ID
994 FORMAT(1X,**PUPPY-909: ',3A2,' NO DATA FOUND!')
RETURN
END

```

```

-----
C***** PUPPY PROGRAM
C***** DAN PRODUST
C***** PUP10.FR NWS RIVER FORECAST CENTER
C***** 5020 FEDERAL OFFICE BLDG
C***** CINCINNATI, OHIO 45202

```

```

C
C SUBROUTINE PLTFM: READS PUP.DT AND ENTERS LATEST STATION DATA
C INTO 2 PLT FILES: NATIONAL AND LOCAL FORMATS
C
-----

```

```

OVERLAY 006
SUBROUTINE PLTFM
COMMON INET(1500,3), ILCTN(1500,3), ITIME(9), IDATE(3), ICHN4
COMMON ICHN2, ICODE, IDAY(3), ISTA, IREC, IBASN(25), ID(3), IFND4
COMMON ISNOW, IWTR, ICTL
DIMENSION IELM(29), JELM(29,10), IOUT(400), NPIX1(4), NPIX2(4)
DIMENSION LPIX1(4), LPIX2(4), LID(6), IVALUE(10), IZM(6)
DIMENSION ISTRG2(6), ISTRG1(6)
ISTRG1(1)="PU" ; ISTRG1 IS RDDS OUTPUT FILE
ISTRG1(2)="P." ; ISTRG2 IS AFOS CCHNNXXX
ISTRG1(4)=" "
ISTRG1(5)=" "
ISTRG1(6)=" "
IDEC=" "
IBLNK=" "
ICOM=","
ISEMI=";"
ISNOW=0 ; NO SNOW YET FOUND- SUPPRESSES WX WIRE SNOW COL.
IWTR=0 ; NO W.EQ. YET FOUND- SUPPRESSES WX WIRE WTR EQUIV
CALL GCHN(ICHN1,IER)
CALL ERROR(IER,**PUPPY-1001: GETTING CHANNEL')
CALL FOPEN(ICHN1,"PUP.DT","B",17)
CALL GCHN(ICHN2,IER)
CALL ERROR(IER,**PUPPY-1002: GETTING CHANNEL')
CALL FOPEN(ICHN2,"NETWORK.DT","B",73)
CALL GCHN(ICHN3,IER)

```



```

CALL ERROR( IER, '**PUPPY-1003: GETTING CHANNEL' )
CALL FOPEN( ICHN3, "PUP.YS", "B", 17 )
CALL GCHN( ICHN4, IER )
CALL ERROR( IER, '**PUPPY-1004: GETTING CHANNEL' )
CALL OPEN( ICHN4, "PUP.02", 0, IER )
CALL ERROR( IER, '**PUPPY-1005: OPENING FILE' )
CALL GCHN( ICHN5, IER )
CALL ERROR( IER, '**PUPPY-1006: GETTING CHANNEL' )
CALL OPEN( ICHN5, "PUP.03", 0, IER )
CALL ERROR( IER, '**PUPPY-1007: OPENING FILE' )
CALL GCHN( ICHN6, IER )
CALL OPEN( ICHN6, "PUP.FL", 0, IER )
ITM=ITIME(1)/100
CALL FGTIM( IHR, IMIN, ISEC )
I79=0
I87=0
3 READ( ICHN6, 994, END=4 ) IFILE, ISTRG2
IF( IFILE.EQ.79 ) I79=1
IF( IFILE.EQ.79 ) WRITE( ICHN4, 991 ) ( ISTRG2( LL ), LL=1, 5 ), IHR, IDAY( 2 ),
1 IDAY( 1 ), IDAY( 3 ), IHR, IMIN, ITM, IDAY, ISTRG2( 4 ), ISTRG2( 5 )
IF( IFILE.EQ.87 ) I87=1
IF( IFILE.EQ.87 ) WRITE( ICHN5, 991 ) ( ISTRG2( LL ), LL=1, 5 ), IHR, IDAY( 2 ),
1, IDAY( 1 ), IDAY( 3 ), IHR, IMIN, ITM, IDAY, ISTRG2( 4 ), ISTRG2( 5 )
IF( I79.NE.1.OR.I87.NE.1 ) GO TO 3
GO TO 6
4 IF( I79.NE.1 ) WRITE( 10, 993 )
IF( I79.NE.1 ) WRITE( 10 ) "**PUPPY-1012: BLANKS ASSUMED!"
IF( I87.NE.1 ) WRITE( 10, 992 )
IF( I87.NE.1 ) WRITE( 10 ) "**PUPPY-1013: BLANKS ASSUMED!"
6 DO 500 I=1, IREC ; SEARCH INET ELM
IF( ILCTN( I, 1 ).EQ.0 ) GO TO 500
DO 5 II=1, 6
LID( II )=IBLNK
5 CONTINUE
DO 10 II=1, 29
DO 15 KK=1, 10
JELM( II, KK )=IBLNK
15 CONTINUE
IELM( II )=IBLNK
10 CONTINUE
CALL FSEEK( ICHN1, ( ILCTN( I, 1 ) ) )
READ( ICHN1, 999, END=90, ERR=90 ) ID, IFIELD
IFIELD=IFIELD/2
DO 100 K=1, IFIELD
READ( ICHN1, 998, END=90, ERR=90 ) KL, IVALUE
IF( KL.LE.0.OR.KL.GT.29 ) GO TO 100
IELM( KL )=KL
DO 150 LL=1, 10
IF( IVALUE( LL ) .EQ. IDEC .AND. KL .EQ. 6 ) GO TO 100
IF( IVALUE( LL ) .EQ. IDEC .AND. KL .EQ. 15 ) GO TO 100
JELM( KL, LL )=IVALUE( LL )
150 CONTINUE
100 CONTINUE
IELM( 1 )=1
JELM( 1 )="8 " ; INSERT STATION CIRCLE
999 FORMAT( 3A2, 2X, I4, 4X )
998 FORMAT( I3, 3X, 10A1 )
CALL FSEEK( ICHN2, I )
READ( ICHN2, 997 ) LID, NPIX2, NPIX1, LPIX2, LPIX1, IZM( 1 )
997 FORMAT( 6A1, 33X, 16A1, 1X, A1, 15X )
IZM( 2 )="0 "

```



```

IZM(3)="0 "
IZM(4)="0 "
IZM(5)="0 "
IZM(6)="6 " ; SEE PMOD REFERENCE MANUAL
IF(IELM(15).NE.IBLNK.OR.IELM(6).NE.IBLNK)ISNOW=1
IF(IELM(16).NE.IBLNK)IWTR=1
J=1
DO 249 N=1,6
IF(LID(N).EQ.IBLNK)GO TO 249
IOUT(J)=LID(N)
J=J+1
249 CONTINUE
IOUT(J)=ICOM
J=J+1
INEW=29
DO 248 N=1,29
IF(IELM(INEW).NE.IBLNK)GO TO 247
INEW=INEW-1
248 CONTINUE
247 DO 250 N=1,INEW
IF(IELM(N).EQ.IBLNK)GO TO 245
DO 252 L=1,10
IF(JELM(N,L).EQ.IBLNK)GO TO 252
IOUT(J)=JELM(N,L)
J=J+1
252 CONTINUE
245 IOUT(J)=ICOM
J=J+1
IN=J
IF(J.LE.44.OR.J.GT.54)GO TO 262
DO 261 J=IN,54
261 IOUT(J)=IBLNK
GO TO 250
262 IF(J.LE.117.OR.J.GT.126)GO TO 264
DO 263 J=IN,126
263 IOUT(J)=IBLNK
GO TO 250
264 IF(J.LE.189.OR.J.GT.198)GO TO 266
DO 265 J=IN,198
265 IOUT(J)=IBLNK
GO TO 250
266 IF(J.LE.261.OR.J.GT.270)GO TO 268
DO 267 J=IN,270
267 IOUT(J)=IBLNK
GO TO 250
268 IF(J.LE.333.OR.J.GT.332)GO TO 250
DO 270 J=IN,332
270 IOUT(J)=IBLNK
250 CONTINUE
IOUT(J)=ISEMI
WRITE(ICHN4,996)NPIX1,ICOM,NPIX2,ICOM,IZM,ICOM,
1ICOM,(IOUT(L),L=1,J)
996 FORMAT(1X,72A1)
WRITE(ICHN5,996)LPIX1,ICOM,LPIX2,ICOM,IZM,ICOM,
1ICOM,(IOUT(L),L=1,J)
GO TO 500
90 WRITE(10)**PUPPY-1008: ERROR READING FILE**
500 CONTINUE
WRITE(ICHN4,995)

```



```

WRITE(ICHN5,995)
995 FORMAT(1X,'END OF DATA',/,1X,'NNNN')
CALL RESET
I79=0 ; IF =1 DONT SEARCH FOR NATIONAL PLT CCCNNNXXX
I87=0 ; IF =1 DONT SEARCH FOR LOCAL PLT CCCNNNXXX
CALL GCHN(ICHN1,IER)
CALL ERROR(IER,'**PUPPY-1009: GETTING CHANNEL')
CALL OPEN(ICHN1,"PUP.FL",0,IER)
550 READ(ICHN1,994,END=600)IFILE,ISTRG2
994 FORMAT(I2,2X,6A2)
IF(IFILE.EQ.79)I79=1
IF(IFILE.EQ.79)ISTRG1(3)="02"
IF(IFILE.EQ.79)CALL PUP(ISTRG1,ISTRG2)
IF(IFILE.EQ.87)I87=1
IF(IFILE.EQ.87)ISTRG1(3)="03"
IF(IFILE.EQ.87)CALL PUP(ISTRG1,ISTRG2)
IF(I79.NE.1.OR.I87.NE.1)GO TO 550
GO TO 601
600 IF(I79.NE.1)WRITE(10,993)
993 FORMAT(1X,'**PUPPY-1010: NO CCCNNNXXX FOR NATIONAL PLT')
IF(I87.NE.1)WRITE(10,992)
992 FORMAT(1X,'**PUPPY-1011: NO CCCNNNXXX FOR LOCAL PLT')
601 CALL RESET
991 FORMAT(1X,4A2,A1,'0010200020481536285014250097501688',6I2,
1/,1X,'1224,0275,10000Z,',I2,'Z ',I2,'/',I2,'/',I2,16X,')',
2/,1X,'1224,0259,10000Z,',A2,A1,' HYDROLOGIC OBSERVATIONS ;')
RETURN
END

```

```

-----
C***** PUPPY PROGRAM
C***** DAN PROVDST
C***** PUP11.FR NWS RIVER FORECAST CENTER
C***** 5020 FEDERAL OFFICE BLDG
C***** CINCINNATI, OHIO 45202

```

```

C SUBROUTINE WEATHR: READS PUP.DT AND ENTERS LATEST STATION DATA
C BY BASIN INTO WEATHER WIRE COMPATIBLE FILES
C

```

```

-----
SUBROUTINE WEATHR(ISTRG2,IBUF,INXWR,IRAWC,IHR,IMIN)
COMMON INET(1500,3),ILCTN(1500,3),ITIME(9),IDATE(3),ICHN4
COMMON ICHN2,ICODE,IDAY(3),ISTA,IREF,IBASN(25),ID(3),IFND4
COMMON ISNOW,IWTRE,ICTL
DIMENSION INXWR(9),IBUF(15),ISTRG2(6),ISTRG1(6),INAME(6)
DIMENSION VALU1(29),VALU2(29),IELEM(29),JELEM(29)
DIMENSION IBNAM(36),IBUF2(36),ISW(6),IRAWC(13),KID(3)
WRITE(10)**PUPPY-1100: PROCESSING WEATHER WIRE FILE!
ISTRG1(1)="PU"
ISTRG1(2)="P."
ISTRG1(3)="04"

```



```

ISTRG1(4)=" "
ISTRG1(5)=" "
ISTRG1(6)=" "
CALL ERROR( IER, '**PUPPY-1115- FILE DELETE ERROR' )
      VALU1=TODAYS DATA      VALU2=YESTERDAYS DATA
      IELEM=TODAYS ELEMENTS   JELEM=YESTERDAYS ELEMENTS
      .....HEADING TITLES.....
      IWXR(1)=PCPN
      IWXR(2)=STAGES
      IWXR(3)=CHANGE IN STAGES
      IWXR(4)=FLOOD STAGE
      IWXR(5)=SNOW
      IWXR(6)=TEMPS
      IWXR(7)=WATER EQUIV
      IWXR(8)=PRESENT WX
      IWXR(9)=FLOW
CALL WXHDG( IWXR, ICHN, IDAY, ITIME, ISNOW, IWTRE, IRAWC, IHR, IMIN )
ICMT="CC"
IRFC="RR" ; COMMENT LINE TO RFC ONLY
IBLNK=" "
IF( ICODE.EQ.5 )WRITE( ICHN, 985 )
IF( ICODE.EQ.2.OR.ICODE.EQ.3 )WRITE( ICHN, 984 )
985 FORMAT( 1X, '..... CORRECTED DATA.....', / )
984 FORMAT( 1X, '..... ADDITIONAL DATA.....', / )
987 FORMAT( 3A2, 1X, I5, 4X )
888 FORMAT( 1X, 6A2, Z )
887 FORMAT( 5X, Z )
886 FORMAT( 1X, I4, Z )
885 FORMAT( 1X, ' MSG ', Z )
884 FORMAT( 1X, F5.2, Z )
883 FORMAT( 6X, Z )
882 FORMAT( 1X, F8.2, Z )
881 FORMAT( 1X, ' MISSING ', Z )
880 FORMAT( 9X, Z )
879 FORMAT( 1X, '+', Z )
878 FORMAT( 2X, Z )
875 FORMAT( 1X, ' MISSING ', Z )
874 FORMAT( 3X, 'DOWNSTREAM', 9X, ' MISSING ', 5X, Z )
873 FORMAT( 4X, 'OBSERVED FLOW ', F8.0, ' CU FT/SEC', Z )
872 FORMAT( 4X, 'OBSERVED FLOW MISSING ', Z )
871 FORMAT( 1X, F5.1, Z )
870 FORMAT( 7X, Z )
869 FORMAT( 1X, ' MSG ', Z )
868 FORMAT( 10X, Z )
867 FORMAT( 4X, Z )
866 FORMAT( 1X, I3, Z )
865 FORMAT( 13X, Z )
864 FORMAT( 2X, 6A2, Z )
862 FORMAT( 3X, 'DOWNSTREAM', 9X, F8.2, Z )
861 FORMAT( 1X, ' TRACE ', Z )
860 FORMAT( 2X )
859 FORMAT( 1X, ' T ', Z )
858 FORMAT( 36X, Z )
857 FORMAT( //, 1X, 'NNNN', // )
CALL GCHN( ICHN1, IER )
CALL FOPEN( ICHN1, "PUP.DT", "B", 17 )
CALL GCHN( ICHN2, IER )
CALL FOPEN( ICHN2, "PUP.ST", "B", 73 )
I99=0 ; CHECK FOR 99 = ALL BASINS TO WX WIRE

```



```

IWRITE=0
IWRT2=0
IBS2=1
IJ=1
DO 1000 I=1,15;ALLOWS 15 BASINS/WXWIR FILE OR 99= ALL
IF( IBUF(I).GT.25.OR.IBUF(I).LE.0)I99=1
IF( I99.EQ.1.AND.IBUF(1).NE.99)GO TO 605
IF( IBUF(1).NE.99)IJ=IBASN( IBUF(I) )
IF( IBUF(1).EQ.99)GO TO 300
IWRITE=0
IBS2=IBUF(I)
300 CALL FSEEK( ICHN2, IJ )
DO 500 J=1,1500
READ( ICHN2, 997, END=600 ) INBR, ID, IBS
997 FORMAT( I3, 3A2, I2, 61X )
IF( IBS.NE.IBS2)IWRITE=0
IF( IBS.NE.IBS2.AND.IBUF(1).NE.99)GO TO 1000
CALL BINSR( ICANT, ILOC )
IF( ICANT.EQ.0)GO TO 500
IF( ILCTN( ILOC, 1 ) EQ.0)GO TO 500
IWRT2=1
IJ=J
DO 401 LL=1,29
IELEM( LL )=0
VALU1( LL )=0
JELEM( LL )=0
VALU2( LL )=0
401 CONTINUE
CALL FSEEK( ICHN1, ( ILCTN( ILOC, 1 ) ) )
READ( ICHN1, 987 ) ID, IFIELD
IFIELD=IFIELD/2
DO 405 L=1, IFIELD
READ( ICHN1, 993 ) IDATA, VALUE
IF( IDATA.LE.0.OR.IDATA.GT.29)GO TO 405
IELEM( IDATA )=IDATA
VALU1( IDATA )=VALUE
405 CONTINUE
IOFF=0
IF( IWXR( 1 ) EQ.1.AND.IELEM( 2 ).NE.0)IOFF=1
IF( IWXR( 2 ) EQ.1.AND.IELEM( 7 ).NE.0)IOFF=1
IF( IWXR( 2 ) EQ.1.AND.IELEM( 8 ).NE.0)IOFF=1
IF( IWXR( 5 ) EQ.1.AND.IELEM( 6 ).NE.0)IOFF=1
IF( IWXR( 5 ) EQ.1.AND.IELEM( 15 ).NE.0)IOFF=1
IF( IWXR( 6 ) EQ.1.AND.IELEM( 12 ).NE.0)IOFF=1
IF( IWXR( 6 ) EQ.1.AND.IELEM( 13 ).NE.0)IOFF=1
IF( IWXR( 6 ) EQ.1.AND.IELEM( 14 ).NE.0)IOFF=1
IF( IWXR( 7 ) EQ.1.AND.IELEM( 16 ).NE.0)IOFF=1
IF( IWXR( 8 ) EQ.1.AND.IELEM( 5 ).NE.0)IOFF=1
IF( IWXR( 9 ) EQ.1.AND.IELEM( 9 ).NE.0)IOFF=1
IF( IOFF.EQ.0)GO TO 500
C IOFF INDICATES DATA EXISTS AND SHOULD BE WRITTEN
ILAST=6
IF( IELEM( 5 ).EQ.0)ILAST=5
IF( IELEM( 16 ).EQ.0.AND.ILAST.EQ.5)ILAST=4
IF( IELEM( 12 ).EQ.0.AND.IELEM( 13 ).EQ.0.AND.IELEM( 14 ).EQ.0.AND
1 ILAST.EQ.4)ILAST=3
IF( IELEM( 15 ).EQ.0.AND.IELEM( 6 ).EQ.0.AND.ILAST.EQ.3)ILAST=2
IF( IELEM( 8 ).EQ.0.AND.ILAST.EQ.2)ILAST=1
IF( IWRITE.EQ.1)GO TO 400

```



```

CALL GCHN(ICHN6, IER)
CALL FOPEN(ICHN6, "PUP.BS", "B", 81)
CALL FSEEK(ICHN6, IBS)
READ(ICHN6, 996, END=1000) LL, IBNAM
WRITE(ICHN, 995) IBNAM
IWRITE=1
IBS2=IBS
CALL CLOSE(ICHN6)
996 FORMAT(I2, 2X, 36A2, 4X)
995 FORMAT(1X, /, 1X, 36A2)
C----- BEGIN PROCESSING WX WIRE DATA
400 CALL GCHN(ICHN9, IER)
CALL ERROR( IER, '**PUPPY-1101: GETTING CHANNEL')
CALL FOPEN(ICHN9, "NETWORK.DT", "B", 73)
CALL FSEEK(ICHN9, ILOC)
READ(ICHN9, 992) INAME, IFLOOD
CALL CLOSE(ICHN9)
992 FORMAT(6X, 6A2, 41X, I4, 9X)
991 FORMAT(6A2, 4X)
993 FORMAT(I3, 3X, F10.2)
IXTRA=0
WRITE(ICHN, 888) INAME
IF(IXWR(4).EQ.0) IFLOOD=0
IF(IFLOOD.EQ.0) WRITE(ICHN, 887)
IF(IFLOOD.NE.0) WRITE(ICHN, 886) IFLOOD
C----- PRECIPITATION
IF(IXWR(1).EQ.0) WRITE(ICHN, 883)
IF(IXWR(1).EQ.0) GO TO 410
IF(IELEM(2).EQ.0) WRITE(ICHN, 883)
IF(IELEM(2).EQ.0) GO TO 410
IF(VALU1(2).EQ.(99.99)) WRITE(ICHN, 885)
IF(VALU1(2).EQ.(88.88)) WRITE(ICHN, 861)
IF(VALU1(2).EQ.(88.88)) GO TO 410
IF(VALU1(2).NE.(99.99)) WRITE(ICHN, 884) VALU1(2)
C----- HEADWATER RIVER STAGE
410 IF(ILAST.EQ.1) GO TO 470
IF(IXWR(2).EQ.0) WRITE(ICHN, 880)
IF(IXWR(2).EQ.0) GO TO 420
IF(IELEM(8).EQ.0) WRITE(ICHN, 880)
IF(IELEM(8).EQ.0) GO TO 420
IF(VALU1(8).EQ.(99.99)) WRITE(ICHN, 881)
IF(VALU1(8).NE.(99.99)) WRITE(ICHN, 882) VALU1(8)
C----- HEADWATER 24 HR CHANGE RIVER STAGE
420 IF(IXWR(3).EQ.0. OR. IXWR(2).EQ.0) WRITE(ICHN, 870)
IF(IXWR(3).EQ.0. OR. IXWR(2).EQ.0) GO TO 430
IF(IELEM(8).EQ.0) WRITE(ICHN, 870)
IF(IELEM(8).EQ.0) GO TO 430
IF(VALU1(8).EQ.(99.99)) WRITE(ICHN, 870)
IF(VALU1(8).EQ.(99.99)) GO TO 430
IF(ILCTN(ILOC, 3).EQ.0) WRITE(ICHN, 870)
IF(ILCTN(ILOC, 3).EQ.0) GO TO 430
CALL GCHN(ICHN3, IER)
CALL ERROR( IER, '**PUPPY-1102: GETTING CHANNEL')
ILC=ILCTN(ILOC, 3)
CALL FOPEN(ICHN3, "PUP.YS", "B", 17)
CALL FSEEK(ICHN3, ILC)
READ(ICHN3, 987) KID, KFLD
KFLD=KFLD/2
IF(KFLD.GT.29. OR. KFLD.LE.0) IELEM(8)=0

```



```
IF(KFLD.GT.29.OR.KFLD.LE.0)GO TO 420
DO 426 L=1,KFLD
READ(ICHN3,993)LDATA,VALUE
IF(LDATA.LE.0.OR.LDATA.GT.29)GO TO 426
JELEM(LDATA)=LDATA
VALU2(LDATA)=VALUE
```

426 CONTINUE

```
CALL CLOSE(ICHN3)
IF(VALU2(8).EQ.0.OR.VALU2(8).EQ.(99.99))WRITE(ICHN,870)
IF(VALU2(8).EQ.0.OR.VALU2(8).EQ.(99.99))GO TO 430
VALUE=VALU1(8)-VALU2(8)
IF(VALUE.GT.(0.))WRITE(ICHN,879)
IF(VALUE.LE.(0.))WRITE(ICHN,878)
WRITE(ICHN,871)VALUE
```

----- SNOW

430 IF(ILAST.EQ.2)GO TO 470

```
WRITE(ICHN,878)
IF(INXWR(5).EQ.0)WRITE(ICHN,887)
IF(INXWR(5).EQ.0)GO TO 435
IF(IELEM(15).EQ.0)WRITE(ICHN,887)
IF(IELEM(15).EQ.0)GO TO 435
IF(VALU1(15).EQ.(99.99))WRITE(ICHN,869)
IF(VALU1(15).EQ.(88.88))WRITE(ICHN,859)
IF(VALU1(15).EQ.(88.88))GO TO 435
IVALUE=VALU1(15)
IF(VALU1(15).NE.(99.99))WRITE(ICHN,886)IVALUE
```

435 IF(INXWR(5).EQ.0)WRITE(ICHN,887)

```
IF(INXWR(5).EQ.0)GO TO 440
IF(IELEM(6).EQ.0)WRITE(ICHN,887)
IF(IELEM(6).EQ.0)GO TO 440
IF(VALU1(6).EQ.(99.99))WRITE(ICHN,869)
IF(VALU1(6).EQ.(88.88))WRITE(ICHN,859)
IF(VALU1(6).EQ.(88.88))GO TO 440
IVALUE=VALU1(6)
IF(VALU1(6).NE.(99.99))WRITE(ICHN,886)IVALUE
```

----- TEMPERATURE

440 IF(ILAST.EQ.3)GO TO 470

```
IF(INXWR(6).EQ.0)WRITE(ICHN,868)
IF(INXWR(6).EQ.0)GO TO 450
IVAL1=VALU1(12)
IVAL2=VALU1(13)
IVAL3=VALU1(14)
IF(IELEM(12).EQ.0)WRITE(ICHN,867)
IF(IELEM(12).EQ.0)GO TO 441
IF(VALU1(12).EQ.(99.99))WRITE(ICHN,867)
IF(VALU1(12).NE.(99.99))WRITE(ICHN,866)IVAL1
```

441 IF(IELEM(13).EQ.0)WRITE(ICHN,867)

```
IF(IELEM(13).EQ.0)GO TO 442
IF(VALU1(13).EQ.(99.99))WRITE(ICHN,867)
IF(VALU1(13).NE.(99.99))WRITE(ICHN,866)IVAL2
```

442 IF(IELEM(14).EQ.0)WRITE(ICHN,867)

```
IF(IELEM(14).EQ.0)GO TO 450
IF(VALU1(14).EQ.(99.99))WRITE(ICHN,867)
IF(VALU1(14).NE.(99.99))WRITE(ICHN,866)IVAL3
```

----- WATER EQUIVALENT

450 WRITE(ICHN,879)

```
IF(INXWR(7).EQ.0)WRITE(ICHN,883)
IF(INXWR(7).EQ.0)GO TO 460
IF(IELEM(16).EQ.0)WRITE(ICHN,883)
```



```
IF(IELEM(16).EQ.0)GO TO 460
IF(VALU1(16).EQ.(99.99))WRITE(ICHN,885)
IF(VALU1(16).EQ.(88.88))VALU1(16)=(0.)
IF(VALU1(16).NE.(99.99))WRITE(ICHN,884)VALU1(16)
```

C----- PRESENT WEATHER

```
460 IF(ILAST.EQ.5)GO TO 470
IF(IWXWR(8).EQ.0)WRITE(ICHN,865)
IF(IWXWR(8).EQ.0)GO TO 470
IF(IELEM(5).EQ.0)WRITE(ICHN,865)
IF(IELEM(5).EQ.0)GO TO 470
IF(VALU1(5).EQ.(99.99))WRITE(ICHN,875)
IF(VALU1(5).EQ.(99.99))GO TO 470
IVAL5=VALU1(5)-1
CALL GCHN(ICHN7,IER)
CALL FOPEN(ICHN7,"PUP.SN","B",17)
CALL FSEEK(ICHN7,IVAL5)
READ(ICHN7,991)ISW
WRITE(ICHN,864)ISW
CALL CLOSE(ICHN7)
470 WRITE(ICHN,860)
```

C----- TAILWATER RIVER STAGE

```
IF(IWXWR(2).EQ.0)GO TO 475
IF(IELEM(7).EQ.0)GO TO 480
IXTRA=1
IF(VALU1(7).EQ.(99.99))WRITE(ICHN,874)
IF(VALU1(7).NE.(99.99))WRITE(ICHN,862)VALU1(7)
475 IF(IWXWR(3).EQ.0.OR.IWXWR(2).EQ.0)GO TO 480
IF(IELEM(7).EQ.0)GO TO 480
IF(VALU1(7).EQ.(99.99))GO TO 480
IF(ILCTN(ILOC,3).EQ.0)WRITE(ICHN,870)
IF(ILCTN(ILOC,3).EQ.0)GO TO 480
IF(VALU2(7).EQ.0.OR.VALU2(7).EQ.(99.99))WRITE(ICHN,870)
IF(VALU2(7).EQ.0.OR.VALU2(7).EQ.(99.99))GO TO 480
VALUE=VALU1(7)-VALU2(7)
IF(VALUE.GT.0)WRITE(ICHN,879)
IF(VALUE.LE.0)WRITE(ICHN,878)
WRITE(ICHN,871)VALUE
```

C----- OBSERVED FLOW

```
480 IF(IWXWR(9).EQ.0)GO TO 490
IF(IELEM(9).EQ.0)GO TO 490
IXTRA=1
IF(IWXWR(2).EQ.0.OR.IELEM(7).EQ.0)WRITE(ICHN,858)
IF(VALU1(9).EQ.(99.99))WRITE(ICHN,872)
IF(VALU1(9).NE.(99.99))WRITE(ICHN,873)VALU1(9)
490 IF(IXTRA.EQ.1)WRITE(ICHN,860)
495 READ(ICHN1,987,END=500,ERR=500)ID,ILINE
IF(ID(1).EQ.IRFC.AND.ID(2).EQ.IRFC)GO TO 495
IF(ID(1).EQ.ICMT.AND.ID(2).EQ.ICMT)GO TO 498
GO TO 500
498 CALL GCHN(ICHN4,IER)
CALL FOPEN(ICHN4,"PUP.CT","B",75)
CALL FSEEK(ICHN4,(ILINE-1))
READ(ICHN4,988,END=500,ERR=500)IBUF2
988 FORMAT(36A2,2X)
WRITE(ICHN,989)IBUF2
989 FORMAT(1X,36A2)
CALL CLOSE(ICHN4)
GO TO 495
```



```

500 CONTINUE
600 IF(IBUF(1).EQ.99)GO TO 1001
   GO TO 1000
605 IF(IBUF(1).EQ.0.AND.IBUF(1).NE.99)GO TO 1001
   WRITE(10)**PUPPY-1105:  BASIN NUMBER OUT OF BOUNDS!"
   I99=0
1000 CONTINUE
1001 WRITE(ICHN,857)
   CALL CLOSE(ICHN)
   CALL CLOSE(ICHN1)
   CALL CLOSE(ICHN2)
   IF(INRT2.EQ.0)GO TO 1002
   CALL PUP(ISTRG1,ISTRG2)
   RETURN
1002 WRITE(10,986)(ISTRG2(L),L=1,5)
   986 FORMAT(1X,**PUPPY-1104:  ',4A2,A1,'EMPTY;NOT STORED IN AFOS.')
```

```

C-----
C*****PUPPY PROGRAM*****
C*****DAN PROVOST*****
C*****PUP12.FR*****
C*****NWS RIVER FORECAST CENTER*****
C*****5020 FEDERAL OFFICE BLDG*****
C*****CINCINNATI, OHIO 45202*****
```

```

C
C SUBROUTINE JULN:  DETERMINES THE JULIAN DATE OF THE LDATE ARRAY
C                   PASSED AND RETURNS VALUE LJUL
C
C-----
```

```

C*****PUP SUBROUTINE #13*****
SUBROUTINE JULN(LDATE,LJUL)
DIMENSION LDATE(3)
COMMON/JLL/IDAYS(12)
DATA IDAYS/31,28,31,30,31,30,31,31,30,31,30,31/
IYR=(LDATE(3)/4) ; CHECK FOR LEAP YEAR
IYR=LDATE(3)-(IYR*4)
IF(IYR.EQ.0)IDAYS(2)=29
LJUL=0
DO 5 I=1,12
IF(LDATE(1).EQ.I)GO TO 6
LJUL=LJUL+IDAYS(I)
5 CONTINUE
6 LJUL=LJUL+LDATE(2)
RETURN
END
```



```

C-----
C***** PUPPY PROGRAM
C***** DAN PROVOST
C***** PUP13.FR NWS RIVER FORECAST CENTER
C***** 5020 FEDERAL OFFICE BLDG
C***** CINCINNATI, OHIO 45202

```

```

C SUBROUTINE RFC: GENERATES A MESSAGE CONTAINING ALL DATA FOR RFC USES.
C
C
C-----

```

```

OVERLAY DU7
SUBROUTINE RFC(ISTRG2,IRAWC,IHR,IMIN)
COMMON INET(1500,3),ILCTN(1500,3),ITIME(9),IDATE(3),ICHN4
COMMON ICHN2,ICODE,IDAY(3),ISTA,IREC,IBASN(25),ID(3),IFND4
COMMON ISNOW,IWTRE,ICTL
DIMENSION ISTRG2(6),ISTRG1(6),IBNAM(36),ISW(6),INAME(6)
DIMENSION IELEM(29),VALU1(29),IRAWC(13),IBUF2(36),IBUF(15)
WRITE(10)**PUPPY-1300: PROCESSING RFC FILE!
ICMT="CC"
IRFC="RR"
IBLANK=" "
CALL GCHN(ICHN,IER)
CALL OPEN(ICHN,"PUP.05",0,IER)
CALL GCHN(ICHN6,IER)
CALL FOPEN(ICHN6,"PUP.BS","B",81)
CALL FSEEK(ICHN6,0)
3 READ(ICHN6,888,END=5)ICD,IBNAM
888 FORMAT(I2,2X,36A2,4X)
IF(ICD.NE.89)GO TO 3
GO TO 7
5 DO 6 I=1,36
6 IBNAM(I)=IBLANK
WRITE(10)**PUPPY-1301: NO WSFO ID FOUND ASSUME BLANK"
7 DO 4 I=1,12
IF(IRAWC(I).EQ.IBLANK.AND.IRAWC(I+1).EQ.IBLANK)GO TO 4
WRITE(ICHN,886)IRAWC(I)
4 CONTINUE
CALL CLOSE(ICHN6)
WRITE(ICHN,887>IDAY(2),IHR,IMIN,IBNAM,(ITIME(L),L=2,9),IDAY
887 FORMAT(2X,3I2,/,1X,36A2,
1/,1X,'HYDROLOGIC OBSERVATIONS ',
28A2,2X,I2, '/', I2, '/', I2,/,1X,/,1X)
886 FORMAT(1X,A1,Z)
IF(ICODE.EQ.5)WRITE(ICHN,985)
IF(ICODE.EQ.2.OR.ICODE.EQ.3)WRITE(ICHN,984)
985 FORMAT(1X,'.....CORRECTED DATA.....',/)
984 FORMAT(1X,'.....ADDITIONAL DATA.....',/)
ISTRG1(1)="PU"
ISTRG1(2)="P."
ISTRG1(3)="05"
ISTRG1(4)=" "
ISTRG1(5)=" "
ISTRG1(6)=" "
VALU1=TODAYS DATA VALU2=YESTERDAYS DATA
IELEM=TODAYS ELEMENTS JELEM=YESTERDAYS ELEMENTS
987 FORMAT(3A2,1X,I5,4X)
885 FORMAT(1X,'CALL STATION',16X,'STAGES',5X,' RIVER DAM SNOW CURM
1AXMIN',/,1X,'SIGN NAME',9X,'PCPN HDWATER TLWATER FLOW GATES TTL2

```



```

24HR TEMPS WT',/)
884 FORMAT(29X,'WATER EQUIVALENT: ',F5.2)
883 FORMAT(29X,'SYNOPTIC WX: ',6A2)
882 FORMAT(29X,'6-HOURLY FLOW 1PM: ',F8.1,' CFS')
881 FORMAT(29X,'6-HOURLY FLOW 7PM: ',F8.1,' CFS')
880 FORMAT(29X,'6-HOURLY FLOW 1AM: ',F8.1,' CFS')
879 FORMAT(29X,'6-HOURLY FLOW 7AM: ',F8.1,' CFS')
878 FORMAT(29X,'7PM TAILWATER STAGE: ',F8.2)
877 FORMAT(29X,'7PM HEADWATER STAGE: ',F8.2)
876 FORMAT(29X,'PREV 7AM TW STAGE: ',F8.2)
875 FORMAT(29X,'PREV 7AM HW STAGE: ',F8.2)
874 FORMAT(29X,'PREV 7AM PRECIP.: ',F5.2)
853 FORMAT(1X,9A2,Z)
860 FORMAT(5X,Z)
861 FORMAT(1X,I4,Z)
868 FORMAT(1X,F5.2,Z)
867 FORMAT(6X,Z)
866 FORMAT(1X,F8.0,Z)
865 FORMAT(1X,F8.2,Z)
864 FORMAT(9X,Z)
856 FORMAT(4X,Z)
857 FORMAT(1X,I3,Z)
852 FORMAT(1X,' T',Z)
850 FORMAT(1X,' M',Z)
849 FORMAT(2X)
848 FORMAT(29X,'WATER EQUIVALENT MISSING')
847 FORMAT(29X,'UNASSIGNED ELEMENT',I3,' ',F8.2)
846 FORMAT(//,1X,'NNNN')
WRITE(ICHN,885)
CALL GCHN(ICHN1,IER)
CALL FOPEN(ICHN1,"PUP.DT","B",17)
CALL GCHN(ICHN2,IER)
CALL FOPEN(ICHN2,"PUP.ST","B",73)
IWROTE=0
IJ=1
IBS2=0
CALL FSEEK(ICHN2,IJ)
DO 500 J=1,1500
READ(ICHN2,997,END=1001)INBR,ID,IBS
997 FORMAT(I3,3A2,I2,61X)
CALL BINSR(ICANT,ILOC)
IF(ICANT.EQ.0)GO TO 500
IF(ILCTN(ILOC,1).EQ.0)GO TO 500
IWROTE=1
DO 401 LL=1,29
IELEM(LL)=0
VALU1(LL)=0
401 CONTINUE
IF(IBS.EQ.IBS2)GO TO 400
CALL GCHN(ICHN6,IER)
CALL FOPEN(ICHN6,"PUP.BS","B",81)
CALL FSEEK(ICHN6,IBS)
READ(ICHN6,996,END=500)LL,IBNAM
WRITE(ICHN,995)IBNAM
IBS2=IBS
CALL CLOSE(ICHN6)
996 FORMAT(I2,2X,36A2,4X)
995 FORMAT(1X,/,1X,36A2)

```



```

C----- BEGIN PROCESSION RFC DATA
400 CALL FSEEK(ICHN1,(ILCTN(ILOC,1)))
    READ(ICHN1,987)ID,IFIELD
    IFIELD=IFIELD/2
    DO 405 L=1,IFIELD
    READ(ICHN1,993)IDATA,VALUE
    IF(IDATA.LE.0.OR.IDATA.GT.29)WRITE(10,983)ID,IDATA,VALUE
983 FORMAT(1X,'**PUPPY-1304: DATA ELEMENT EXCEEDED-',/,1X,
13A2,2X,I3,2X,F10.2)
    IF(IDATA.LE.0.OR.IDATA.GT.29)GO TO 405
    IELEM(IDATA)=IDATA
    VALU1(IDATA)=VALUE
405 CONTINUE
    CALL GCHN(ICHN9,IER)
    CALL ERROR(IER,'**PUPPY-1302: GETTING CHANNEL')
    CALL FOPEN(ICHN9,"NETWORK.DT","B",73)
    CALL FSEEK(ICHN9,ILOC)
    READ(ICHN9,992,END=500,ERR=500)INAME,IFLOOD
    CALL CLOSE(ICHN9)
992 FORMAT(6X,6A2,41X,I4,9X)
991 FORMAT(6A2,4X)
993 FORMAT(I3,3X,F10.2)
    IXTA=0
    WRITE(ICHN,853)ID,INAME
C----- PRECIPITATION
    IF(IELEM(2).EQ.0)WRITE(ICHN,867)
    IF(IELEM(2).EQ.0)GO TO 410
    WRITE(ICHN,868)VALU1(2)
C----- HEADWATER RIVER STAGE
410 IF(IELEM(8).EQ.0)WRITE(ICHN,864)
    IF(IELEM(8).EQ.0)GO TO 420
    WRITE(ICHN,865)VALU1(8)
C----- TAILWATER RIVER STAGE
420 IF(IELEM(7).EQ.0)WRITE(ICHN,864)
    IF(IELEM(7).EQ.0)GO TO 425
    WRITE(ICHN,865)VALU1(7)
C----- FLOW
425 IF(IELEM(9).EQ.0)WRITE(ICHN,864)
    IF(IELEM(9).EQ.0)GO TO 428
    WRITE(ICHN,866)VALU1(9)
C----- DAM GATES
428 IF(IELEM(10).EQ.0)WRITE(ICHN,860)
    IF(IELEM(10).EQ.0)GO TO 430
    IVAL1=VALU1(10)
    WRITE(ICHN,861)IVAL1
C----- SNOW
430 IF(IELEM(15).EQ.0)WRITE(ICHN,860)
    IF(IELEM(15).EQ.0)GO TO 435
    IF(VALU1(15).EQ.(99.99))WRITE(ICHN,850)
    IF(VALU1(15).EQ.(88.88))WRITE(ICHN,852)
    IF(VALU1(15).EQ.(88.88))GO TO 435
    IVALUE=VALU1(15)
    IF(VALU1(15).NE.(99.99))WRITE(ICHN,861)IVALUE
435 IF(IELEM(6).EQ.0)WRITE(ICHN,860)
    IF(IELEM(6).EQ.0)GO TO 440
    IF(VALU1(6).EQ.(99.99))WRITE(ICHN,850)
    IF(VALU1(6).EQ.(88.88))WRITE(ICHN,852)
    IF(VALU1(6).EQ.(88.88))GO TO 440
    IVALUE=VALU1(6)

```



```

IF(VALU1(6).NE.(99.99))WRITE(ICHN,861)IVALUE
C----- TEMPERATURE
440 IVAL1=VALU1(12)
    IVAL2=VALU1(13)
    IVAL3=VALU1(14)
    IVAL4=VALU1(11)
    IF(IELEM(12).EQ.0)WRITE(ICHN,856)
    IF(IELEM(12).EQ.0)GO TO 441
    IF(VALU1(12).EQ.(99.99))WRITE(ICHN,856)
    IF(VALU1(12).NE.(99.99))WRITE(ICHN,857)IVAL1
441 IF(IELEM(13).EQ.0)WRITE(ICHN,856)
    IF(IELEM(13).EQ.0)GO TO 442
    IF(VALU1(13).EQ.(99.99))WRITE(ICHN,856)
    IF(VALU1(13).NE.(99.99))WRITE(ICHN,857)IVAL2
442 IF(IELEM(14).EQ.0)WRITE(ICHN,856)
    IF(IELEM(14).EQ.0)GO TO 443
    IF(VALU1(14).EQ.(99.99))WRITE(ICHN,856)
    IF(VALU1(14).NE.(99.99))WRITE(ICHN,857)IVAL3
443 IF(IELEM(11).EQ.0)WRITE(ICHN,856)
    IF(IELEM(11).EQ.0)GO TO 450
    IF(VALU1(11).EQ.(99.99))WRITE(ICHN,856)
    IF(VALU1(11).NE.(99.99))WRITE(ICHN,857)IVAL4
C----- WATER EQUIVALENT
450 WRITE(ICHN,849)
    IF(IELEM(16).EQ.0)GO TO 460
    IF(VALU1(16).EQ.(99.99))WRITE(ICHN,848)
    IF(VALU1(16).EQ.(88.88))VALU1(16)=(0.)
    IF(VALU1(16).NE.(99.99))WRITE(ICHN,884)VALU1(16)
C----- PRESENT WEATHER
460 IF(IELEM(5).EQ.0)GO TO 470
    IF(VALU1(5).EQ.(99.99))GO TO 470
    IVAL5=VALU1(5)-1
    CALL GCHN(ICHN7,IER)
    CALL FOPEN(ICHN7,"PUP.SN","B",17)
    CALL FSEEK(ICHN7,IVAL5)
    READ(ICHN7,991)ISW
    WRITE(ICHN,883)ISW
    CALL CLOSE(ICHN7)
C----- 6-HOURLY FLOW DATA
470 IF(IELEM(17).EQ.0)GO TO 471
    WRITE(ICHN,882)VALU1(17)
471 IF(IELEM(18).EQ.0)GO TO 472
    WRITE(ICHN,881)VALU1(18)
472 IF(IELEM(19).EQ.0)GO TO 473
    WRITE(ICHN,880)VALU1(19)
473 IF(IELEM(20).EQ.0)GO TO 480
    WRITE(ICHN,879)VALU1(20)
C----- 7PM, 7AM STAGES + PREV 7AM PCPN DATA
480 IF(IELEM(21).EQ.0)GO TO 481
    WRITE(ICHN,878)VALU1(21)
481 IF(IELEM(22).EQ.0)GO TO 482
    WRITE(ICHN,877)VALU1(22)
482 IF(IELEM(23).EQ.0)GO TO 483
    WRITE(ICHN,876)VALU1(23)
483 IF(IELEM(24).EQ.0)GO TO 484
    WRITE(ICHN,875)VALU1(24)
484 IF(IELEM(4).EQ.0)GO TO 490
    WRITE(ICHN,874)VALU1(4)

```



C----- UNASSIGNED ELEMENTS

```

490 IF(IELEM(1).EQ.0)GO TO 491
    WRITE(ICHN,847)IELEM(1),VALU1(1)
491 IF(IELEM(3).EQ.0)GO TO 492
    WRITE(ICHN,847)IELEM(3),VALU1(3)
492 IF(IELEM(25).EQ.0)GO TO 493
    WRITE(ICHN,847)IELEM(25),VALU1(25)
493 IF(IELEM(26).EQ.0)GO TO 494
    WRITE(ICHN,847)IELEM(26),VALU1(26)
494 IF(IELEM(27).EQ.0)GO TO 495
    WRITE(ICHN,847)IELEM(27),VALU1(27)
495 IF(IELEM(28).EQ.0)GO TO 496
    WRITE(ICHN,847)IELEM(28),VALU1(28)
496 IF(IELEM(29).EQ.0)GO TO 497
    WRITE(ICHN,847)IELEM(29),VALU1(29)
497 READ(ICHN1,987,END=500,ERR=500)ID,ILINE
    IF(ID(1).EQ.IRFC.AND.ID(2).EQ.IRFC)GO TO 498
    IF(ID(1).EQ.ICMT.AND.ID(2).EQ.ICMT)GO TO 498
    GO TO 500
498 CALL GCHN(ICHN4,IER)
    CALL FOPEN(ICHN4,"PUP.CT","B",75)
    CALL FSEEK(ICHN4,(ILINE-1))
    READ(ICHN4,988,END=500,ERR=500)IBUF2
988 FORMAT(36A2,2X)
    WRITE(ICHN,989)IBUF2
989 FORMAT(1X,36A2)
    CALL CLOSE(ICHN4)
    GO TO 497
500 CONTINUE
1001 WRITE(ICHN,846)
    CALL CLOSE(ICHN)
    CALL CLOSE(ICHN1)
    CALL CLOSE(ICHN2)
    IF(IWROTE.EQ.0)GO TO 1002
    CALL PUP(ISTRG1,ISTRG2)
    RETURN
1002 WRITE(10,986)(ISTRG2(L),L=1,5)
986 FORMAT(1X,'**PUPPY-1303: ',4A2,A1,'EMPTY;NOT STORED IN AFOS.')
```

RETURN  
END



PUPPY PROGRAM

DAN PROVOST  
 NWS RIVER FORECAST CENTER  
 5020 FEDERAL OFFICE BLDG  
 CINCINNATI, OHIO 45202

PUP16.FR

SUBROUTINE WXHDG: DETERMINES WEATHER WIRE HEADING REQUIRED  
 BASED ON OPTIONS DETERMINED BY USER

```

SUBROUTINE WXHDG(IWXWR, ICHN, IDAY, ITIME, ISNOW, INTRE, IRAWC, IHR, IMIN)
DIMENSION IBNAM(36), IDAY(3), ITIME(9), IWXWR(9), IRAWC(13)
IBLANK=" "
IF(ISNOW.EQ.0)IWXWR(5)=0
IF(INTRE.EQ.0)IWXWR(7)=0
CALL FGTIM(IHR, IMIN, ISEC)
CALL GCHN(ICHN, IER)
CALL OPEN(ICHN, "PUP.04", 0, IER)
CALL GCHN(ICHN6, IER)
CALL FOPEN(ICHN6, "PUP.BS", "B", 81)
CALL FSEEK(ICHN6, 0)
3 READ(ICHN6, 888, END=5)ICD, IBNAM
888 FORMAT(I2, 2X, 36A2, 4X)
IF(ICD.NE.89)GO TO 3
GO TO 7
5 DO 6 I=1, 36
6 IBNAM(I)=IBLNK
WRITE(10)"**PUPPY-1601: NO WSFO ID FOUND ASSUME BLANK"
7 DO 4 I=1, 12
IF(IRAWC(I).EQ.IBLANK.AND.IRAWC(I+1).EQ.IBLANK)GO TO 4
WRITE(ICHN, 886)IRAWC(I)
4 CONTINUE
WRITE(ICHN, 887)IDAY(2), IHR, IMIN, IBNAM, (ITIME(L), L=2, 9), IDAY
887 FORMAT(2X, 3I2, /, 1X, 36A2,
1/, 1X, 'HYDROLOGIC OBSERVATIONS ',
28A2, 2X, I2, /, I2, /, I2, /, 1X, /, 1X)
886 FORMAT(1X, A1, 2)
WRITE(ICHN, 999)
IF(IWXWR(4).EQ.1)WRITE(ICHN, 998)
IF(IWXWR(4).NE.1)WRITE(ICHN, 997)
IF(IWXWR(1).EQ.1)WRITE(ICHN, 996)
IF(IWXWR(1).NE.1)WRITE(ICHN, 997)
IF(IWXWR(2).EQ.1)WRITE(ICHN, 995)
IF(IWXWR(2).NE.1)WRITE(ICHN, 991)
IF(IWXWR(3).EQ.1)WRITE(ICHN, 994)
IF(IWXWR(3).NE.1)WRITE(ICHN, 993)
IF(IWXWR(5).EQ.1)WRITE(ICHN, 992)
IF(IWXWR(5).NE.1)WRITE(ICHN, 991)
IF(IWXWR(6).EQ.1)WRITE(ICHN, 990)
IF(IWXWR(6).NE.1)WRITE(ICHN, 989)
IF(IWXWR(7).EQ.1)WRITE(ICHN, 988)
IF(IWXWR(7).NE.1)WRITE(ICHN, 977)
WRITE(ICHN, 986)
IF(IWXWR(4).EQ.1)WRITE(ICHN, 985)
IF(IWXWR(4).NE.1)WRITE(ICHN, 997)
IF(IWXWR(1).EQ.1)WRITE(ICHN, 984)
IF(IWXWR(1).NE.1)WRITE(ICHN, 997)
IF(IWXWR(2).EQ.1)WRITE(ICHN, 983)

```



```

IF(IWXWR(2).NE.1)WRITE(ICHN,991)
IF(IWXWR(3).EQ.1)WRITE(ICHN,982)
IF(IWXWR(3).NE.1)WRITE(ICHN,993)
IF(IWXWR(5).EQ.1)WRITE(ICHN,981)
IF(IWXWR(5).NE.1)WRITE(ICHN,991)
IF(IWXWR(6).EQ.1)WRITE(ICHN,980)
IF(IWXWR(6).NE.1)WRITE(ICHN,989)
IF(IWXWR(7).EQ.1)WRITE(ICHN,979)
IF(IWXWR(7).NE.1)WRITE(ICHN,997)
IF(IWXWR(8).EQ.1)WRITE(ICHN,978)
IF(IWXWR(8).NE.1)WRITE(ICHN,999)
WRITE(ICHN,987)
999 FORMAT(12X,Z)
998 FORMAT(1X,'FLOOD',Z)
997 FORMAT(6X,Z)
996 FORMAT(1X,' 24HR',Z)
995 FORMAT(1X,'  RIVER ',Z)
994 FORMAT(1X,' 24HR ',Z)
993 FORMAT(7X,Z)
992 FORMAT(1X,' SNOW ',Z)
991 FORMAT(9X,Z)
990 FORMAT(1X,'CURMAXMIN',Z)
989 FORMAT(10X,Z)
988 FORMAT(1X,' WATER')
987 FORMAT(1X,/)
986 FORMAT(1X,'STATION  ',Z)
985 FORMAT(1X,'STAGE',Z)
984 FORMAT(1X,' PCPN',Z)
983 FORMAT(1X,'  STAGE ',Z)
982 FORMAT(1X,'CHANGE',Z)
981 FORMAT(1X,' TTL24HR',Z)
980 FORMAT(1X,' TEMPS ',Z)
979 FORMAT(1X,' EQUIV',Z)
978 FORMAT(1X,' PRESENT WX ',Z)
977 FORMAT(7X)
CALL CLOSE(ICHN6)
RETURN
END

```



# 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 (D822)  
ENVIRONMENTAL DATA AND INFORMATION SERVICE  
NATIONAL OCEANIC AND ATMOSPHERIC ADMINISTRATION  
U.S. DEPARTMENT OF COMMERCE**

**6009 Executive Boulevard  
Rockville, MD 20852**

NOAA CENTRAL LIBRARY  
CIRC O08743 U63 no.3  
Provost, Dan PUPPY (AFOS hydrologic dat  
3 8398 0003 7512 5