NOAA WESTERN REGION COMPUTER PROGRAMS AND PROBLEMS NWS WRCP - NO. 57 (REVISED)

WATCHDOG

William R. Schneider
Craig C. Peterson

WSFO Seattle
WSFO Los Angeles

June 1991

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

PREFACE

This Western Region publication series is a subset of our Technical Memorandum series. This series will be devoted exclusively to the exchange of information on and documentation of computer programs and related subjects. This series was initiated because it did not seem appropriate to publish computer program papers as Technical Memoranda; yet, we wanted to share this type of information with all Western Region forecasters in a systematic way. Another reason was our concern that in the developing AFOS-era there would be unnecessary and wasteful duplication of effort in writing computer programs in National Weather Service (NWS). Documentation and exchange of ideas and programs envisioned in this series hopefully will reduce such duplication. We also believe that by publishing the programming work of our forecasters, we will stimulate others to use these programs or develop their own programs to take advantage of the computing capabilities AFOS makes available.

We solicit computer-oriented papers and computer programs from forecasters for us to publish in this series. Simple and short programs should not be prejudged as unsuitable.

The great potential of the AFOS-era is strongly related to local computer facilities permitting meteorologists to practice in a more scientific environment. It is our hope that this series will help in developing this potential into reality.

NOAA WESTERN REGION COMPUTER PROGRAMS AND PROBLEMS NWS WRCP

1 Standardized Format for Computer Series. Revised January 1984. (PB85 109668)
2 AFOS Crop and Soil Information Report Programs. Kenneth B. Mielke, July 1979. (PB85 110419)
3 Decoder for Significant Level Transmissions of Raobs. John A. Jannuzzi, August 1979. (PB85 109676)
4 Precipitable Water Estimate. Elizabeth Morse, October 1979.
5 Utah Recreational Temperature Program. Kenneth M. Labas, November 1979.
6 Normal Maximum/Minimum Temperature Program for Montana. Kenneth B. Mielke, December 1979. (PB85 112878)
7 Plotting of Ocean Wave Energy Spectral Data. John R. Zimmerman, December 1979. (PB85 112860)
8 Raob Plot and Analysis Routines. John A. Jannuzzi, January 1980.
9 The SWAB Program. Morris S. Webb, Jr., April 1980. (PB80 196041)
10 Flash-Flood Procedure. Donald P. Laurine and Ralph C. Hatch, April 1980. (PB80 298658)
11 Program to Forecast Probability of Summer Stratus in Seattle Using the Durst Objective Method. John R. Zimmerman, May 1980.
12 Probability of Sequences of Wet and Dry Days. Hazen H. Bedke, June 1980. (PB80 223340)
13 Automated Montana Hourly Weather Roundup. Joe L. Johnston, July 1980. (PB81 102576)
14 Lightning Activity Levels. Mark A. Mollner, July 1980. (PB81 108300)
15 Two FORTRAN Applications of Wind-Driven Ekman Water Transport Theory: Upwelling Index and Storm Tide. Kent S. Short, July 1980. (PB81 102568)
16 AFOS System Local Data Base Save and Rebuild Procedures or a Master Doomsday Program. Brian W. Finke, July 1980. (PB81 108342)
17 AFOS/RDOS Translator Subroutine. Morris S. Webb, Jr., August 1980. (PB81 108334)
18 AFOS Graphics Creation from FORTRAN. Alexander E. MacDonald, August 1980. (PB81 205304)
19 DATAKEY0/ Repair Program. Paul D. Tolleson, August 1980. (PB81 102543)
20 Contiguous File Transfer from the DPCM to the DCM. Paul D. Tolleson, September 1980. (PB81 128035)
21 Freezing Level Program. Kenneth B. Mielke, September 1980. (PB81 128043)
22 Radar Boresighting Verification Program. Thomas E. Adler, November 1980. (PB81 182677)
23 Accessing the AFOS Data Base. Matthew Peroutka, January 1981. (PB81 190266)
24 AFOS Work Processor. Morris S. Webb, Jr., February 1981. (PB81 210007)
25 Automated Weather Log for Terminal Forecasting. John A. Jannuzzi, February 1981. (PB81-210999)
26 Program to Computer Downwind Concentrations from a Toxic Spill. John R. Zimmerman, February 1981. (PB81 205296)
27 Animation of AFOS Graphics. James R. Fors, August 1987 (revision). (PB87 220109/AS)
28 AFOS Interactive Graphics. James R. Fors, Don Laurine, and Sandy MacDonald, April 1981. (PB85 110401)
29 Computer Programs for Aviation Forecast Transmission. Kenneth B. Mielke and Matthew R. Peroutka, May 1981. (PB85 110518)
30 AFOS Product Collective Program. Morris S. Webb, Jr., September 1981. (PB85 109841)
32 Automation of Hourly Aviation Observation Calculations. W. Paul Duval, October 1981. (PB85 109650)
33 Mesoscale Objective Analysis. Andrew J. Spry and Jeffrey L. Anderson, December 1981. (PB85 109825)
34 Orographic Snowfall Rate Model for Alta, Utah. Steven K. Todd and Glenn E. Rasch, December 1981. (PB85 109874)
35 F-6 Monthly Climatic Summary Program For AFOS. Peter G. Mueller, May 1982. (PB85 109858)
36 Soaring Forecast Program. D.S. Toronto and G. R. Lussky, Revised March 1986. (PB86 173523/AS)
37 Program to Work Up Climatic Summary Weather Service Forms (F-6, F-52). Peter G. Mueller, August 1982. (PB85 109866)
38 The Hovmoller Diagram. Pamela A. Hudadoff, September 1982. (PB85 112159)
39 850-Millibar Charts Derived from Surface Data. Jeffrey L. Anderson, December 1982. (PB85 112175)
40 AFOS Vector Graphic to Grid Point Program. James R. Fors, December 1982. (PB85 109544)
41 A Pilot Briefing Program for the Background Partition. Kenneth B. Mielke and Joe L. Johnston, March 1983. (PB85 109551)
42 AEV Local Verification for Aviation, Precipitation, and Temperature Programs: AV, REL, TEM. Timothy W. Barker, Revised September 1987. (PB88 115662/AS)
43 OBLOG. Nancy Larsen, December 1983. (PB85 109528)
44 Communications Software for Olympics Micromation Computer System. Glen Sampson, June 1984. (PB85 109510)
45 PLOTFILE Appender. Wendy L. Wolf, July 1984. (PB85 109502)
46 Spectral Wave Data Analysis (Non-Directional). Lawrence Dunn, August 1984. (PB85 109577)
47 Isentropic Objective Analysis. Jeffrey L. Anderson, August 1984. (PB85 112167)
48 Hurricane Plotting Program. Paul D. Tolleson, October 1984. (PB85 121432)
49 Hemispheric Spectral Wave Analysis (Waves 0 to 7). Mary F. Milkovich, August 1985. (PB86 108719/AS)
50 AOS Graphic to Grid Point Conversion and Departure from Normal Programs. Jeffrey L. Anderson and Mark A. Mathewson, August 1985. (PB85 248110/AS)
51 Sunrise/Sunset and Moonrise/Moonset. Glenn R. Lussky, January 1986 (Revised). (PB86 157229/AS)
52 Objective Contour Analysis Using the Surface of Least Bending (Spline Analysis). Les Colin, November 1985. (PB86 128675/AS)
53 DATACOL - AFOSPLOT Program. Donald P. Laurine and Timothy K. Helble, February 1986. (PB86 161866/AS)
54 Hemispheric Spectral Analysis Program. Craig C. Peterson, April 1986. (PB 183662/AS)
55 Convective Cross Section Analysis. Timothy W. Barker, June 1987. (PB87 204566)
56 SWELL Program. Craig C. Peterson, August 1987. (PB87 229795/AS)
57 Watchdog Program. William R. Schneider and Craig C. Peterson, October 1988. (PB89 122535/AS)
58 Daily Climate Summary for MAPSO. Joe L. Johnston, August 1989. (PB89 230841/AS) - Revised May 1991
59 SEAPLOT. Bob Diaz and Steve Todd, December 1989. (PB90 151333/AS)
60 NWWS Product Retransmission Program. William R. Schneider, March 1990. (PB90 199092/AS)
61 A System of Collecting RAWS Data For Dissemination over AFOS. Dennis D. Gettman, January 1991. (PB91-153460)

NOAA Western Region Computer Programs and Problems NWS WRCP NO. 57 (Revised)

WATCHDOG

William R. Schneider
Weather Service Forecast Office
Seattle, Washington

Craig C. Peterson
Weather Service Forecast Office
Los Angeles, California

June 1991

UNITED STATES
DEPARTMENT OF COMMERCE
Robert A. Mosbacher, Secretary

National Oceanic and
Atmospheric Administration
John A. Knauss, Under Secretary and Administrator

National Weather Service
Elbert W. Friday, Jr., Assistant Administrator for Weather Services

This publication has been reviewed and is approved for publication by Scientific Services Division, Western Region

Ken Mielke

Kenneth B. Mielke, Chief
Scientific Services Division
Salt Lake City, Utah

TABLE OF CONTENTS

WATCHDOG PROGRAM

I. INTRODUCTION

A. Purpose and Motivation

The WATCHDOG program is a highly efficient multi-tasking assembly language background scheduler that runs continuously in the AFOS background partition. It concurrently performs five tasks: time schedules background programs, event schedules background programs, monitors the ADMs for "RUN:" commands, emulates the CLI at the dasher, and monitors the AFOS database for late or missing products.

B. Version 11.00

There have been two significant changes since Watchdog version 10.02, as well as a host of other minor improvements. The efficiency of monitoring the AFOS database (this program only reads DATAKEY0) has been greatly improved through the use of read-only Direct AFOS Database Access (DADA). Also, a very efficient late product monitor feature has been added. Details of all enhancements since version 10.02 are discussed in Section IV. Instructions on setting up the new late product monitor can be found in Section II.B.

II. METHODOLOGY AND SOFTWARE STRUCTURE

A. General

WATCHDOG is a package of six different AFOS background programs. Three of these programs WDSETUP.SV, WDRESTART.SV, and WDRST.SV are utility programs. The heart of WATCHDOG consists of three programs that run at three different levels of program execution. These programs are WDOG.SV, WDCLI.SV, and WD.SV. A description of these programs is given below:

WDOG.SV - WDOG.SV is the WATCHDOG initialization and interrupt handling program and runs at level 1, one level above BGMON.SV, which runs at level 0. WDOG.SV only runs when WATCHDOG is first started (as with an AFOS reboot) or when the background partition receives an interrupt. WDOG.SV has several functions which are listed below:

1. It checks to see if AFOS is running. If it is not, it returns to BGMON.SV.

2. It determines if MODIFY is running. If it is, the program suspends itself until modify has been completed.

3. It checks to see if a dasherless boot has been executed. If it has, the program waits until a valid time is entered at an ADM. If a valid time is not entered within 3 minutes, the program returns to BGMON.SV.

4. The program automatically runs the setup program, WDSETUP.SV, if changes have been made to DATAKEY0 or the WISHLIST. It also runs the setup program if WATCHDOG has not been run within the last 120 minutes.

5. The program handles all console (ctrl-A and ctrl-C) and program (trap) interrupts while WATCHDOG is running.

6. If the auto-reboot option is enabled, it makes sure that the program RESTART.SV is on the SYSZ partition.

7. It stores any programs that were in the WATCHDOG queue into the file WDQUE for use by the auto-rerun feature.

8. It throws away unwanted data sent to the background through the foreground-background communications interface by AFOS after an interrupt.

9. Finally, the program swaps to WDCLI.SV, which executes the WDK indirect.

WDCLI.SV - This program is a slightly modified version of the DATA GENERAL CLI program and runs at level 2 in the background. Its main task is to run the indirect WDK, which is the command file for WDCLI.SV. This indirect file contains the following commands:

```
WD.SV
CMD.MC
@WDK@
```

The first command is the WATCHDOG background scheduler program. The next command is the macro containing commands queued by WATCHDOG for WDCLI to execute. The last command recalls the indirect so it will run continuously.

WD.SV - This is the main WATCHDOG program; it runs at level 3. This program consists of several tasks described below.

WD - This task performs the following functions:

1. It checks to see if changes have been made to the WDSCHD or WDLATE files. If so, the WDSETUP.SV program is run.

2. It checks to see if any jobs are in the rerun queue. If jobs are in the rerun queue, a listing is output to the dasher.

3. It reads all data files into memory above NMAX. In addition, it allocates buffer space for the program.

4. Finally, WD queues up four other tasks and then kills itself. These four tasks, NTMON, FGMON, WDBMON, and DS run concurrently.

NTMON - This task performs three functions:

1. It passes through the entries in the time schedule file WDTIME every minute to determine if any programs need to be run. If any valid time scheduled programs are found, they are placed in a command queue and the program termination task, STOP, is activated.

2. It passes through the late product monitor entries from the file WDLATE every minute. This feature allows sites to automatically monitor the AFOS database for late products. If the creation time of a product falls outside of an assigned window, a message is stored into one of the AFOS products CCCMISWD(1-9), CCCMISWDR or CCCMISWDS depending on the console specified by the operator in the WDLATE file. The products CCCMISWD(1-9) are set to alarm at their corresponding console number by the ASM/APL. CCCMISWDR and CCCMISWDS are used to store alarm messages to be transmitted synchronously to another site specified in the WDLATE file. Multiple late messages for the same console are all stored at one time. However, late messages which are transmitted synchronously (through CCCMISWDR or CCCMISWDS) are stored individually.

3. The new system day and time are written to WD.DT.

WDBMON - This task checks the AFOS database for receipt of new products (associated with keys) and for updated RDOS files. The key and file types are entered into the WDSCHD file. Each time the WD.SV program is run, WDBMON begins searching for new products. The DATAKEY0 pointers associated with the keys, and the creation date and time of any RDOS files, are stored in the data file WD.DT. If one or more new products or files are found, the associated command is placed in a command queue, and the STOP task is activated. If no new products have arrived, WDBMON waits for 30 seconds before checking the database again.

FGMON - This task monitors the foreground/background communications area for ADM "RUN" commands. If any "RUN" commands are found, they are placed in a command queue and the STOP task is activated. If foreground/background communications fail, FGMON checks to see if the WATCHDOG auto reboot feature is enabled. If not, the program terminates and returns to WDOG.SV which passes control back to the level 0 program (BGMON.SV). If the auto reboot feature is enabled, WDBMON renames the program SYSZ:RESTART.SV to SYSZ:RESTART.BU. WDBMON then establishes a link entry from SYSZ:RESTART.SV to the WATCHDOG restart program WDRESTART.SV. The file, WD.DT, is updated in order to save the current date and time. Finally, WATCHDOG boots into the SYSZ partition and executes the program WDRESTART.SV (see WDRESTART.SV below for details).

DS - This task monitors the dasher for input and emulates all CLI supported dasher features. If a dasher command is entered, the command is buffered and the STOP task is activated.

STOP - If at any time a task has programs to run, the STOP task is activated. This task performs an orderly shutdown of WD.SV and writes any commands to the CMD.MC file. The program then exits and runs these commands. Several programs

may be run at once as each task may have queued commands and activated the stop task. This allows for rapid throughput of the background partition.

The three other WATCHDOG save files WDRESTART.SV, WDSETUP.SV, and WDRST.SV are described below:

WDRESTART.SV - This program is used only during an auto-reboot. The program sets the current system time and date by reading the creation time and date of the WD.DT file (which is the current time minus a few seconds). WDRESTART then re-establishes the normal AFOS restart program by unlinking SYSZ:RESTART.SV from SYSZ: WDRESTART.SV and renaming SYSZ: RESTART.BU to SYSZ:RESTART.SV. Finally, WDRESTART chains to the CLI (BGMON.SV) which in turn executes the start-up file AFOS.ST. AFOS.ST contains the command @AFOS@. The indirect file "AFOS" must contain commands to restart AFOS System Z and Watchdog.

WDSETUP.SV - This program must be run the first time the WATCHDOG software is installed. The program inputs the current date and time, the DATAKEY0 key pointers, the creation date and time of the RDOS files, and the creation date and time of the WDSCHD and WDLATE files, into the WD.DT file. It also checks the WDTIME, WDSCHD, and WDLATE files for correct syntax. After WATCHDOG is installed, it is usually not necessary to run this program again as WATCHDOG will run WDSETUP.SV automatically if it needs to be run. WATCHDOG will automatically run WDSETUP.SV under the following conditions:

1. If the WATCHDOG program has not been run in over two hours.

2. If the WDSCHD or WDLATE files have been changed.

3. If DATAKEY0 or the WISHLIST have been changed.

If the operator is experiencing any problem with WATCHDOG, running the WDSETUP program manually will normally solve the problem, or indicate where the problem is.

WDRST.SV - This program performs the following functions:

1. It writes the alarm destinations for the WATCHDOG program (e.g., to a console group or to the dasher).

2. It handles the auto-rerun feature.

3. It checks to make sure that programs are running from the master directory SYSZ (except dasher entered commands or programs). If the current directory is not SYSZ, WDRST.SV changes the current directory to SYSZ.

WDRST.SV is transparent to the user.

B. Structure of the WDSCHD, WDTIME, and WDLATE Files

The files WDSCHD and WDTIME define the time and event commands that are to be run at a particular office. The WDLATE file defines the products to be checked by

the WATCHDOG late product monitor. All of these files should be edited with the RDOS file editor.

Note: In the examples that follow, the blank lines between the commands have been inserted for clarity; there should not be any vertical spacing between these commands in the actual WDTIME, WDSCHD, and WDLATE files. In addition, dashes denote spaces in this text. However, there should not be any dashes in the actual WDTIME, WDSCHD, or WDLATE files.

WDSCHD

The WDSCHD file contains a list of key and file-triggered programs. There are three types of entries in the WDSCHD file which are outlined below:

1. Normal key entry format. Each time the key is updated in your database, WATCHDOG will run the associated command.

   CCCNNNXXX-A-COMMANDS

   where:

   CCCNNNXXX - any valid AFOS product in your database including graphics, plotfiles, etc. Keys with less than 9 characters must be padded with spaces.

   A - Alarm console ID number. Must be 1-9 or D (where D denotes the dasher). Any message produced by the program will be sent to this console.

   COMMAND - Any macro, program, or RDOS command. The command must be less than 65 characters.

2. Key/time combination format. When the key is updated in your database, WATCHDOG checks the creation time of the product. If the creation time falls within any of the time windows listed below the key, the associated command is run. If the creation time does not fall within any of the windows, the command on the last line is run. If the creation time does not fall within any of the windows and there is no command on the last line, nothing is run. The time windows may overlap, and, therefore, more than one command may be queued for a single key. There is no limit to the number of time windows for each key.

   CCCNNNXXX-#
   SHSM-EHEM-A-COMMAND
   SHSM-EHEM-A-COMMAND
   #----------A-COMMAND

   where:

   CCCNNNXXX - any valid AFOS product in your database. Keys with less than 9 characters must be padded with spaces.

   # - the first pound symbol indicates that time windows will follow.

SHSM - starting hour and minute of time window.

EHEM - ending hour and minute of time window.

A - alarm value. (Same as defined before.) Alarm value must be in 11th space.

COMMAND - command to be run.

- the last pound symbol indicates the last line of key/time entry. The alarm symbol and command following this pound symbol are optional.

3. RDOS file entry. If the RDOS file is updated, the associated command is run. If the file does not exist or is deleted, the command is not run. If a file has a use count other than zero, the command will not be run until the use count becomes zero.

$$-FILE---A-COMMAND

FILENAME

where:

$$-FILE - indicates RDOS file type entry.

A - alarm value. Must be in the 11th space.

COMMAND - command to be run.

FILENAME - any valid RDOS filename, including directory or partition name. The file should be left blank if no key or file scheduled programs are necessary. The spacing shown must be adhered to. Be sure to put an extra space or two after AFOS database keys of 8 or 7 letters, respectively. Some examples are given below:

LAXSAOLAX-D-HRR

NMCPLT50A-1-GTOD;PLOTUA

LAXMISNO6-D-TEKTRANS LAXMISNO6/I

LAXRRACA2-#

0000-0559-1-SHEF00

0600-1159-1-SHEF06

1200-1759-1-SHEF12

1800-2359-1-SHEF18

#---------- D MESSAGE ERROR IN SHEF

NMCGPH5IV-D-LOOP2

LAXWRKA20-#

0045-0332-D-SAODEC1.MC

0150-0600-D-SAODEC2.MC

#

LAXFRH7--2-FRHPLOT LAX/C

$$-FILE---D-DECODE

SYSZ:DIRB:OBDATA

LAXSAOLAX-4-GTOD;MONITR.MC

$$-FILE---D-BILL.MC

BILL.DP

(- denotes a space)

Approximately 250 entries are allowed in the WDSCHD file. If the file is too large, the WDSETUP program will output an error message.

WDTIME

The WDTIME file controls the time queued programs. There are nine types of entries. The spacing shown must be adhered to. Examples of these are given below:

I---0005-01-D-HRR

M-011812----D-MWO

D---0712----1-SDB

S-5807

S-0130-0140

R

R-0015

W---1357

----1830----D-QPF

(- denotes a space)

where:

I denotes a command to be triggered at intermediate times (e.g., at 0005Z and every hour after, run the program HRR, and ignore the alarm messages). The numbers after the start time indicate the repeat interval (i.e., 01, run every hour, 02, run every two hours, etc.). The start time must be the first time at or after 0000Z the program is to be run.

M denotes a command to be triggered on a particular day at a particular time (e.g., on the first day of the month at 1812Z, run the program MWO and ignore the alarm messages).

D denotes a command to be triggered once a day at a particular time (e.g., daily at 0712Z run the program SDB. Send the alarm messages to console number 1).

S denotes a silent period for key-queued commands. There are two formats for this feature. The command S 5807 will prevent any commands in the file WDSCHD from running between two minutes before the hour and seven minutes after the hour. The command S 0130 0140 prevents any key and file-triggered commands from running between 0130Z and 0140Z. When the silent period feature is used, it prevents time-scheduled programs (like the state weather roundup) and ADM-queued programs (like the FTA and TWB splitter) from being preempted by key and file-triggered programs.

R enables the auto-reboot feature (should not be used at non-System Z dual sites). There are two formats for this feature. The command R 0015 will enable the auto-reboot feature between the hours starting at 0000Z and ending at 1500Z and disable the auto-reboot feature during the rest of the day. The command R will enable the auto-reboot feature continuously. For System Z sites, the indirect AFOS startup file "AFOS" must contain the commands to start AFOS System Z and Watchdog, and the front panel switches on the Eclipse must be set to -1 (all in the "up" position).

W denotes a command to be triggered on a particular day of the week at a particular time. On the first line, the day of the week is entered (e.g., Sunday=1, Monday=2, ...Saturday=7). As many as six days of the week can be input in a single entry. On the second line the trigger time is entered as well as the alarm and the command. In the example on page 7, on either Sunday, Tuesday, Thursday, or Saturday at 1830Z, the program QPF will be run.

As with the WDSCHD file, the command that is queued can be any macro, indirect, program, RDOS command, etc., as long as the command is less than 65 characters. If there are no time-scheduled entries, the file is left blank.

Approximately 250 entries are allowed in the WDTIME file. If the file is too large, WDSETUP will output an error message.

WDLATE

The WDLATE file contains a list of the times and keys needed to monitor the database for late or missing products. If the creation time of a product falls outside of an assigned window, a message is stored in the AFOS database. Ten separate AFOS keys are used to store the late product messages. The keys are CCCMISWD(1-9), CCCMISWDR, and CCCMISWDS. This allows messages to be directed to AFOS consoles 1-9. The products CCCMISWDR and CCCMISWDS are for remote monitoring, and a synchronous address can be specified for these products. These keys should all be alarmed at the appropriate consoles. Examples are given below:

I---0000-01-R-LAXSAOLAX-0011- RLA

D---0030----3-LAXSTPCA--0030

M-011830----3-LAXOPULAX-0600

W---23456

----1530----4-LAXFTABIH-0030

The entries I, D, M, and W are the same as the WDTIME file and have been described above. The alarms can be 1-9, R or S (corresponding to AFOS keys CCCMISWD(1-9), CCCMISWDR, and CCCMISWDS). After the alarm, the AFOS key to be monitored is entered. Next, the window is given in hours and minutes. Finally, the synchronous address for remote monitoring is given. The examples are described in full below:

In the first example, the product LAXSAOLAX is monitored at 0000Z and every hour after that. If the product is more than 11 minutes old, the key CCCMISWDR will be stored into the database and sent to RLA.

In the second example, every day at 0030Z LAXSTPCA will be checked. If it is more than 30 minutes old, a message will be stored in AFOS under CCCMISWD3.

In the third example, on the first day of the month the product LAXOPULAX is checked. If it is more than 6 hours old, a message will be stored in AFOS under CCCMISWD3.

In the fourth example, Monday through Friday LAXFTABIH is checked at 1530Z. If it is more than 30 minutes old, a message will be stored in CCCMISWD4.

Approximately 250 entries are allowed in the WDLATE file. If the file is too large, WDSETUP will output an error message.

III. OPERATING THE WATCHDOG PROGRAM

A. General

Follow the loading instructions given in Part A.

WATCHDOG is started by entering the command WDOG at the dasher. Although the WATCHDOG program can be started manually, it is more convenient to enter the command WDOG at the end of the AFOS initialization indirects, including MODIFY. During MODIFY, the WATCHDOG program will print the message WATCHDOG WAITING FOR MODIFY TO COMPLETE. When MODIFY is finished WATCHDOG will run automatically.

If the AFOS year entered is less than 1988 or greater than 1998 the message "YEAR ENTERED IS LESS THAN 1988 OR GREATER THAN 1998" will be output to the dasher every 30 seconds. If a correct time is not entered after 3 minutes, WDOG returns to BGMON. This gives the operator time to set the correct AFOS time during a dasherless reboot.

When WATCHDOG is running, a "**W**" prompt is displayed at the dasher. If you interrupt WATCHDOG and programs are in the rerun queue, the following message will be output at the dasher: THE FOLLOWING JOBS ARE IN THE RERUN QUE (CTRL-O TO DELETE). If desired, the entire contents of the rerun queue can then be deleted by entering a ctrl-O at the dasher. The message "WATCHDOG QUE DELETED" will be output to the dasher after the rerun queue is deleted.

When WATCHDOG is running, you can use the dasher to enter CLI commands as before. If a program is running, commands entered at the dasher are buffered by RDOS. These commands will not display on the dasher until control has returned to the WATCHDOG program. Commands entered at the dasher will be processed slower than with the normal background monitor program. If you plan to do CLI work, it is recommended that you do the following:

1. Enter "WDCLI" at the dasher (this will push the CLI to level 4). You will receive a "**K**" prompt at the dasher.

2. Perform CLI commands as you normally would. Commands will now be processed much faster. However, during this time, all programs will be held until you return to the WATCHDOG program. The late product monitor is also suspended until you return. (Note: The FORTRAN compiler will not run at this level. To run the FORTRAN compiler, return to the WATCHDOG program using the instructions given in step 3 below).

3. When you are finished using the CLI, enter "POP" at the dasher. The WATCHDOG program will begin to run, and a "**W**" prompt will be output on the dasher.

If you wish to terminate the WATCHDOG program in an orderly manner, enter ctrl-W at the dasher. The message "WD TERM" will be printed at the dasher. Control will then return to BGMON and an "**R**" prompt will be output. If WATCHDOG is running

another program, the program may be interrupted by entering a ctrl-A at the dasher as before.

B. Creating Macro Files to Change Schedule Files

The schedule files WDSCHD, WDTIME, and WDLATE can easily be modified while the WATCHDOG program is running. This feature allows easy changes of the schedule files and is desirable for many reasons. For example, the files may need to be changed when a WSFO runs on one side only, if a WSO needs to back up a WSFO, if a forecaster who comes on shift wishes to run different programs than another forecaster, etc.

This can be accomplished by using a set of macros to make the necessary changes. The operator simply runs the macro while WATCHDOG is running; the changes are made to the schedule files and incorporated into the WATCHDOG program as it runs. For example, let's say that a WSO must back up its WSFO. It may be necessary for the WSO to run certain programs to produce the State Weather Roundup. All the WSO must do is run a macro that replaces their current time schedule file in WDTIME with one that contains the backup schedule. The commands in the macro would look like this:

```
CHATR WDTIME 0

DELETE WDTIME

XFER WDBACKUP WDTIME/R

CHATR WDTIME +P
```

The file WDBACKUP contains the time schedule file for backing up the WSFO (this time schedule would run the State Weather Roundup at the appropriate times). After the WSFO comes back on line, another macro can be run to return the WSO to normal operation.

```
CHATR WDTIME 0

DELETE WDTIME

XFER WDTIME.BU WDTIME/R

CHATR WDTIME +P
```

The file WDTIME.BU contains a copy of the time schedule file that is normally used.

The same procedure can be used for the WDSCHD or WDLATE files as WATCHDOG will automatically run WDSETUP if it determines that a new WDSCHD or WDLATE file exists (WDSETUP needs to be run only when changes are made to the WDSCHD or WDLATE files, not when changes are made to the WDTIME file). For example, let's say one forecaster wishes to have several key-triggered programs alarm at his/her console. A macro can be created for that forecaster to change the WDSCHD file. The macro might look like this:

CHATR WDSCHD 0

DELETE WDSCHD

XFER BILLSSCHD WDSCHD/R

CHATR WDSCHD +P

The file BILLSSCHD would contain a product schedule file with the console alarms set the way Bill wants them. The setup program WDSETUP will automatically run after this macro is completed. To reset the schedule file, another macro can be run. That macro would look like this:

CHATR WDSCHD 0

DELETE WDSCHD

XFER WDSCHD.BU WDSCHD/R

CHATR WDSCHD +P

The file WDSCHD.BU contains the normal schedule file.

Many macro files can be created to make changes to the WDSCHD, WDTIME, or WDLATE files as needed. A time schedule file may even contain a call to one of these macros so that it is reset automatically at a certain time.

C. Changing Schedule Files at a Remote Site

The WATCHDOG schedule files WDTIME, WDSCHD, or WDLATE can be changed at remote sites. This may be useful at a WSFO that wishes to change the WATCHDOG schedule at a WSO that is closed. The procedure to do this must be set up ahead of time by the WSFO and the WSO. There are many different ways to change schedule files at a remote site; one is outlined below.

Place the following entries in the WDSCHD file at the remote site (WSO).

$$-FILE-D-WDSCHDCHG.MC

WDSCHD1

$$-FILE-D-WDTIMECHG.MC

WDTIME1

The macro WDSCHDCHG.MC at the remote site (WSO) would look like this,

MESSAGE WSFO LAX CHANGING

MESSAGE WDSCHD AT WSO SAN

CHATR SYSZ:WDSCHD 0

DELETE/V SYSZ:WDSCHD

RENAME SYSZ:WDSCHD1 SYSZ:WDSCHD

and the macros WDTIMECHG.MC would look like this,

MESSAGE WSFO LAX CHANGING

MESSAGE WDTIME AT WSO SAN

CHATR SYSZ:WDTIME 0

DELETE/V SYSZ:WDTIME

RENAME SYSZ:WDTIME1 SYSZ:WDTIME

Now the WSFO can change the schedule files at the WSO by using the AFOS "MAIL" command. When the file WDSCHD1 or WDTIME1 is sent to the WSO by the WSFO, WATCHDOG will run the appropriate change macro. If the WDSCHD or WDLATE file is changed, WATCHDOG will also automatically run the WDSETUP program.

D. Starting and Terminating WATCHDOG From an ADM

WATCHDOG may be started from an AFOS ADM by entering the command "RUN:WDOG." If you wish to have the capability to terminate WATCHDOG from an ADM, you can create a macro, WDTERM.MC, to do so as follows:

WDTERM.MC

MESSAGE TERMINATING WATCHDOG

MESSAGE BYE BYE DOGGIE

POP

To terminate WATCHDOG at any ADM, enter "RUN:WDTERM". This may be useful in the event of a dasher failure. However, if programs are running, WATCHDOG will not terminate until the programs are finished and control is passed back to WATCHDOG.

E. WATCHDOG Auto-reboot

The automatic reboot feature allows the WATCHDOG program to automatically reboot AFOS if it determines that AFOS is not running. WATCHDOG will not attempt to reboot AFOS unless the auto-reboot feature is enabled in the WDTIME file. The auto-reboot feature is enabled by inserting an **R** in the WDTIME file as described in Section II, Part B.

If the auto-reboot feature is enabled and WATCHDOG determines AFOS is down, WATCHDOG will perform a full system reboot equivalent to entering BOOT SYSZ:SYS at the dasher.

During a WATCHDOG auto-reboot, the following messages will be printed on the dasher:

W

MASTER DEVICE RELEASED

WATCHDOG REBOOTING AFOS

These messages are followed by the normal AFOS start-up messages.

Automatic reboot will not occur under the following circumstances:

1. RDOS failure.
2. Background program hang.
3. Failure of certain tasks of the AFOS software (e.g. FICR).

IV. PROGRAM ENHANCEMENTS SINCE VERSION 10.02

The following is a list of the major changes to WATCHDOG since version 10.02:

1. All I/O in the main WATCHDOG program (WD.SV) has been converted to block I/O.
2. The three files, WDSCHD, WDTIME, and WDLATE, are read into memory above NMAX.
3. The speed and efficiency of database monitoring has greatly improved. The new keysearch method is about 60 times faster than the old program. As a result of this and improvements in the program design, a database monitoring pass takes about 1/100 of the time it took in version 10.02. For example, if there are 50 entries in the WDSCHD file, while the old version would take about 90 seconds to check them all, the new program takes less than one second.
4. As a result of the increased efficiency of the database monitor task, the time between monitor passes has been reduced from 3 minutes to 30 seconds. This allows jobs to run much sooner.
5. Auto-rerun is no longer an option. Auto-rerun is always enabled.
6. The setup program does more error checking. It will no longer allow the user to run WATCHDOG if errors are found in WDSCHD, WDLATE, or WDTIME.

7. Dasherless reboot time problems have been solved. The program now allows about 3 minutes after the system is up to reset the time.

8. The number of concurrent open files has been reduced from 14 to 6.

9. All ADM "RUN:" commands that are waiting in AFOS are captured and queued at the same time.

10. The program WDOG.SV automatically runs the setup program WDSETUP.SV whenever a change is made to DATAKEY0 (either a change to the Wishlist or through EDITMERGE). This is necessary as a result of changes in the program design.

11. The files ACMD.MC, DCMD.MC, NCMD.MC, RCMD.MC, TCMD.MC, and WDSTART.MC have been eliminated. The need for the file WD.SC has been eliminated in the main WATCHDOG program. Delete these files from the system.

12. WDOG.SV only prints out the rerun queue message if there are jobs in the rerun queue.

13. The rerun queue is easier to delete.

14. Unlike the previous version, WDSCHD, WDTIME, and WDLATE can be changed while the program is running without causing a "SYS ERROR." However, the changes do not become effective until the next time WD.SV swaps in.

15. If the auto-reboot feature is enabled WDOG now checks the status of the AFOS restart program RESTART.SV to ensure it is properly linked. This should prevent rebooting problems that result when the operator terminates the system in the middle of an auto-reboot.

16. The leap year computation in WDRESTART.SV has been corrected.

17. If the current directory is something other than SYSZ when Watchdog runs a program, the directory is changed to SYSZ. The only exception to this is when the command is entered at the dasher. This prevents the CLI from accidentally being left in the wrong directory.

18. A late product monitor has been added to the program. The automatic late product alarm feature allows sites to monitor their database for late or missing products. If the creation time of a product falls outside of an assigned window, WATCHDOG generates a message with a list of the late products and sends it to AFOS via the FSTORE process.

V. CAUTIONS AND RESTRICTIONS

The following cautions and restrictions should be noted by the users:

1. Do not CHATR the following files permanent:

   CMD.MC

   WDQUE

   CCCMISWD(1-9)

   CCCMISWDR

   CCCMISWDS

2. The WATCHDOG setup program WDSETUP.SV will only work when AFOS is up.

3. **The macro WDSTOP.MC must be included in the STOP or SITESTOP macro in order to clear and delete any unwanted CLI files. If this is not done, WATCHDOG will eventually stop working properly. The directory must be in SYSZ before entering the WDSTOP macro, i.e.,**

   DIR SYSZ
   WDSTOP.MC

4. Never enable the auto-reboot feature at a non-System Z dual site.

5. It is highly recommended that AFOS products in the Wishlist not be used for key triggers in WDSCHD and WDLATE.

WATCHDOG

PART A: PROGRAM INFORMATION AND INSTALLATION PROCEDURE

PROGRAM NAME: WATCHDOG AAL ID:
REVISION NO.: 11.00

PURPOSE: WATCHDOG is an automatic background scheduler. It allows for time and event scheduling, monitoring of the ADM and dasher for commands, and monitoring of the AFOS database for late or missing products.

PROGRAM INFORMATION:

Development Programmers: William R. Schneider
WSFO Seattle, FTS 392-6083

Craig C. Peterson
WSFO Los Angeles, FTS 793-7218

Maintenance Programmers: Same as above

Language: Data General ASSEMBLY language (except WDSETUP which is written in FORTRAN)

Save Files Creation Dates:

Name	Date	Version
WDOG.SV	05/31/89	11.00
WDCLI.SV	04/25/89	11.00
WD.SV	12/13/89	11.02
WDRST.SV	05/28/89	11.00
WDSETUP.SV	10/10/89	11.01
WDRESTART.SV	05/30/89	11.00

Running Time: Variable

Disk Space: Program Files - 145 RDOS Blocks
Disk Data Files - 5 RDOS Blocks

PROGRAM REQUIREMENTS:

Program Files:

Name	Location	Comments
WDOG.SV	SYSZ	Interrupt handler
WD.SV	SYSZ	Main program
WDRST.SV	SYSZ	Sets console alarms, handles auto-rerun
WDCLI.SV	SYSZ	WATCHDOG CLI
O	SYSZ	WATCHDOG CLI overlay
WDSETUP.SV	SYSZ	Setup program

WDRESTART.SV	SYSZ		Program for automatic reboot
E	SYSZ		WATCHDOG CLI error codes

Data Files:

Name	Location	R/W	Comments
WD.DT	SYSZ	R/W	Main data file
WDTIME	SYSZ	R	Time schedule
WDSCHD	SYSZ	R	Event schedule
WDLATE	SYSZ	R	Late products schedule
WDK	SYSZ	R	Command file for WDCLI.SV
WDSTOP.MC	SYSZ	R	Clears WDCLI files
WDCLR	SYSZ	R/W	List of CLI files to be cleared
CMD.MC	SYSZ	R/W	Command macro
WDQUE	SYSZ	R/W	Rerun queue
CCCMISWD(1-9,R,S)	SYSZ	W	Late products

AFOS PRODUCTS:

Name	Location	R/W	Comments
CCCMISWD(1-9,R,S)	SYSZ	W	Handles late entry alarms

LOAD LINES:

1. RLDR WDRST AFOSE.LB

2. RLDR WDOG WSTART UPD MOD SCOMP AFOSE.LB

3. RLDR WD NTMON WDBMON FGMON WDL WSTORE ST DS ECHO WDCHK ERROR SYS.LB AFOSE.LB

4. RLDR WDSETUP WTIME WLATE WSCHD WDSER COMP ALARM CONS KSRCF ENSCH URREV CFILE FSIZE RSTAT WISHFIX HEADN TOP.LB UTIL.LB FORT.LB UTIL.LB AFOSE.LB

5. RLDR WDRESTART

PROGRAM INSTALLATION:

1. Move all the files on the floppy to the SYSZ partition. The following files should be moved:

   WDOG.SV
   WD.SV
   WDCLI.SV
   O
   E
   WDRST.SV
   WDSETUP.SV

WDK
WDSTOP.MC
WDSCHD
WDTIME
WDLATE
WDRESTART.SV

If you do not have enough contiguous space for the O file (WD CLI overlay), move to the APPL1 directory and provide a link entry from SYSZ to APPL1.

2. Using the AFOS file editor, add the command "WDSTOP.MC" to your SITESTOP.MC macro. Make sure the system is in the master directory (SYSZ) before this macro is executed. This change to the SITESTOP.MC macro may produce messages at the dasher during AFOS start-up. Example:

   FILE DOES NOT EXIST: CLI.S0
   FILE DOES NOT EXIST: CLI.T0
   DELETED CLI.T2

   Do not change any of the commands in the WDSTOP.MC macro in any way.

3. Create the WDTIME, WDSCHD, and WDLATE schedule files as explained in Part II Section B. If the WDLATE file is longer than five pages, it will probably be necessary to create several smaller files and put them together using the CLI command "APPEND".

4. Add CCCMISWD(1-9), CCCMISWDR, and CCCMISWDS into the database or Wishlist.

5. Run the WATCHDOG setup program WDSETUP.SV at the dasher. AFOS must be up and running for the setup program to run properly. The following messages should be received at the dasher:

   WDSETUP PROGRAM - STARTING

   WDTIME - OK

   WDSCHD - OK

   WDLATE - OK

   WDSETUP PROGRAM - FINISHING

   If any errors are received, an error message is printed at the dasher, and the program terminates. Correct the errors and re-run the WDSETUP.SV program **(the setup program must complete successfully before WATCHDOG is run).**

6. Follow the instructions under Part III, "OPERATING THE WATCHDOG PROGRAM".

PART B: PROGRAM EXECUTION AND ERROR CONDITIONS

PROGRAM NAME: WATCHDOG

AAL ID:
REVISION NO.: 11.00

PROGRAM EXECUTION:

1. Start WATCHDOG at the dasher by typing WDOG. WATCHDOG can be executed automatically by adding WDOG as the last command of all AFOS start-up indirects.

2. More specific information can be found in Part III.

ERROR CONDITIONS:

1. If you attempt to run WDOG.SV or WD.SV on the wrong level, the message "ATTEMPT TO RUN WD.SV ON THE WRONG LEVEL" or "ATTEMPT TO RUN WDOG.SV ON THE WRONG LEVEL" will be output to the dasher. The program will then return control to the next lowest level.

2. A "FILE DOES NOT EXIST:WDOG.SV" or "END OF FILE:WD.SV" error on the dasher usually signifies a problem with WD.DT, WDSCHD, WDTIME, or WDLATE. The normal solution for these errors is to run WDSETUP and allow it to complete successfully before restarting WATCHDOG.

3. For all other errors, the error description followed by WD TERM and an "R" prompt will be received at the dasher.

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:

NATIONAL TECHNICAL INFORMATION SERVICE

U. S. DEPARTMENT OF COMMERCE

5285 PORT ROYAL ROAD

SPRINGFIELD, VA 22161