NWS-ERCP-5

NOAA Eastern Region Computer Programs
and Problems NWS ERCP - No. 5

CONVERSION OF ALEMBIC$ WORKBINS TO TELETYPE FORMAT

Scientific Services Division
Eastern Region Headquarters
October 1982

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).

3 PUPPY (AFOS Hydrologic Data Reporting Program). Daniel P. Provost, December 1981. (PB82 199720).

4 Special Search Computer Program. Alan P. Blackburn. April 1982.

NOAA Eastern Region Computer Programs and Problems - No. 5

CONVERSION OF ALEMBIC$ WORKBINS TO TELETYPE FORMAT

Alan Blackburn
National Weather Service Forecast Office
Buffalo, New York

Scientific Services Division
Eastern Region Headquarters
October 1982

UNITED STATES DEPARTMENT OF COMMERCE
Malcolm Baldrige, Secretary
National Oceanic and Atmospheric Administration
John V. Byrne, Administrator
National Weather Service
Richard E. Hallgren, Director

CONVERSION OF ALEMBIC$ WORKBINS TO TELETYPE FORMAT

Alan Blackburn
National Weather Service Forecast Office
Buffalo, New York

I. General Information

A. Summary

TAPE.SV is used to convert terminal forecasts (FTs) and route forecasts (TWEBs) into a format which can be punched on an asynchronous line. FTs and TWEBs must be in format used by ALEMBIC$.SV.

B. Environment

The program is written in FORTRAN and is designed to be run at a WSFO with the foreground up.

II. Application

A. Complete Program Description

When Gateway is down and SMCC asks that data be entered by paper tape, it is tedious and time-consuming to edit FTs and TWEBs into format for punching. TAPE.SV takes FTs and TWEBs from CCCWRKFTA and CCCWRKTWB, and returns the punchable product to the WRK file. It adds an appropriate header and ending and changes $ to {{< , changes - to >@<, and changes + to " . It also keeps lines to a maximum of 72 characters. The AFREAD subroutine is used to read the data from the WRK file. Additional lines of an FT are indented one space.

B. Machine Requirements

10752 bytes of disk space are required. The program takes about 20 seconds to run with one and one-half ADM pages of data.

C. Data Base

CCCWRKFTA or CCCWRKTWB are read from and written to. An RDOS file named PUNCHOUT is created on DP0.

III. Procedures

A. Initiation of the Program

File TAPE.SV must be on DP0 or linked to it. Run the program from an ADM with RUN:TAPE CCCWRKFTA or RUN:TAPE CCCWRKTWB, where CCC is the WSFO identifier.

B. Input Required

FTs and TWEBs beginning with a $ must be in the referenced WRK file and must not extend beyond column 77. (See Figure 1.)

C. Output

Output is in the referenced file and an RDOS file named PUNCHOUT. (See Figure 2.) The following error messages may be printed when the program aborts:

1. "CANT FIND FILE" The AFREAD subroutine can't open the reference file.

2. "CANT READ FILE" The AFREAD subroutine aborts when reading a line of text.

3. "LINE TOO LONG" Text extends beyond column 77.

4. "91Ø ERROR" There are no spaces in 72 consecutive columns.

D. Cautions or Restrictions on Program Use

The program looks for the end of a line by searching for three consecutive spaces, so anything beyond a three-space interval will be ignored. Therefore don't indent the second line of FTs by more than two spaces.

IV. Complete Program Listing

Figure 1

```
WOUS00 KBUF 110000

$CAK 052222 35 SCT 6H SCT OCNL BKN CHC 3FH. 03Z -X 10 SCT 3FH VRBL
 C3X 3/4F. 08Z C2X 1/4F VRBL -X 8 SCT 2F. 14Z 5 SCT C12 BKN 4FH CHC BRF
 C5 X 3/4TRW+F. 16Z MVFR CIG TRW FH..
$TOL 052222 C25 BKN 5H BKN OCNL SCT CHC -X 10 SCT 3FH. 04Z -X 10 SCT 3F
 OCNL C4X 3/4F. 07Z C2X 1/4F VRBL -X 10 SCT 11/2F. 14Z 6 SCT C14 BKN
 4FH BKN V SCT. 16Z MVFR CIG FH..
$CMH 052222 C30 BKN 5H BKN V SCT CHC -X2FH. 02Z -X 10 SCT 3FH OCNL
 C3X 1/4F. 05Z C1X 1/8F VRBL -X C5 OVC 1F. 14Z 5 SCT C12 BKN 4FH BKN V
 SCT. 16Z MVFR CIG FH..
$MFD 052222 C25 BKN 5H BKN V SCT  CHC -X 3F AFT 00Z. 03Z -X 10 SCT 3FH
 OCNL C4X 1/2F. 06Z C0X 0F VRBL -X C6 OVC 1F. 14Z 4 SCT C12 BKN 4FH BKN
 V SCT. 16Z MVFR CIG FH. AMDTS NOT AVBL AFT 03Z..
$LUK 052222 35 SCT 5H SCT V BKN CHC -X 20 SCT 2FH AFT 00Z. 02Z -X 8 SCT
 2FH VRBL C2X 1/4F. 05Z C1X 1/4F VRBL -X C5 OVC 1F. 14Z -X 5 SCT C12
 OVC 4FH. 16Z MVFR CIG FH..
$CVG 052222 35 SCT 5H SCT V BKN CHC -X 20 SCT 2FH AFT 00Z. 02Z -X 8 SCT
 2FH VRBL C2X 1/4F. 05Z C1X 1/4F VRBL -X C5 OVC 1F. 14Z -X 5 SCT C12
 OVC 4FH. 16Z MVFR CIG FH..
$DAY 052222 C30 BKN 5H BKN V SCT CHC -X 2FH. 02Z -X 10 SCT 3FH OCNL
 C3X 1/4F. 05Z C1X1/8F VRBL -X C5 OVC 1F. 14Z 5 SCT C12 BKN 4FH BKN V
 SCT. 16Z MVFR CIG FH..
$FDY 052222 C25 BKN 5H BKN OCNL SCT CHC -X 10 SCT 3F. 04Z -X 10 SCT 3F
 OCNL C4X 3/4F. 07Z C2X 1/4F VRBL -X 10 SCT 11/2F.14Z 6 SCT C14 BKN 4FH
 BKN V SCT. 16Z MVFR CIG FH..
$CLE 052222 C20 BKN 5H BKN V SCT CHC -X 3F AFT 01Z. 03Z -X 10 SCT 3FH
 OCNL C7X 3/4F. 07Z C3X 1/4F VRBL -X 8 SCT 2F. 14Z 6 SCT C14 BKN 4FH
 BKN V SCT. 16Z MVFR CIG FH..
$YNG 052222 35 SCT 6H SCT OCNL BKN  CHC 3FH. 03Z -X 10 SCT 3FH VRBL
 C3X 3/4F. 08Z C2X 1/4F VRBL -X 8 SCT 2F. 14Z 5 SCT C12 BKN 4FH. 16Z
 MVFR CIG H..
$ZZV 052222 C30 BKN 5H BKN OCNL SCT CHC -X 25 SCT 2F. 02Z -X 20 SCT 3F
 VRBL C3X 1/2F. 07Z C2X 1/4F VRBL -X 6 SCT C10 OVC 2F. 14Z 3 SCT C8 OVC
 2F OCNL C12 BKN 4F. 16Z IFR CIG FH BCMG MVFR CIG F H BY 18Z..
```

Figure 2

```
<@@@@@@<SSE@@EE<EE@@VSSV@@@@@@
<<<<<<<<<<<<ZCZC(( )<CLE Z FT)<((
CAK 052222 35 SCT 6H SCT OCNL BKN CHC 3FH. 03Z >@X 10 SCT 3FH VRBL
 C3X 3/4F. 08Z C2X 1/4F VRBL >@X 8 SCT 2F. 14Z 5 SCT C12 BKN 4FH CHC BRF
 C5 X 3/4TRW"F. 16Z MVFR CIG TRW FH..
(((TOL 052222 C25 BKN 5H BKN OCNL SCT CHC >@X 10 SCT 3FH. 04Z >@X 10 SCT
 3F OCNL C4X 3/4F. 07Z C2X 1/4F VRBL >@X 10 SCT 11/2F. 14Z 6 SCT C14 BKN
 4FH BKN V SCT. 16Z MVFR CIG FH..
(((CMH 052222 C30 BKN 5H BKN V SCT CHC >@X2FH. 02Z >@X 10 SCT 3FH OCNL
 C3X 1/4F. 05Z C1X 1/8F VRBL >@X C5 OVC 1F. 14Z 5 SCT C12 BKN 4FH BKN V
 SCT. 16Z MVFR CIG FH..
(((MFD 052222 C25 BKN 5H BKN V SCT  CHC >@X 3F AFT 00Z. 03Z >@X 10 SCT
 3FH OCNL C4X 1/2F. 06Z C0X 0F VRBL >@X C6 OVC 1F. 14Z 4 SCT C12 BKN 4FH
 BKN V SCT. 16Z MVFR CIG FH. AMDTS NOT AVBL AFT 03Z..
(((LUK 052222 35 SCT 5H SCT V BKN CHC >@X 20 SCT 2FH AFT 00Z. 02Z >@X 8
 SCT 2FH VRBL C2X 1/4F. 05Z C1X 1/4F VRBL >@X C5 OVC 1F. 14Z >@X 5 SCT
 C12 OVC 4FH. 16Z MVFR CIG FH..
(((CVG 052222 35 SCT 5H SCT V BKN CHC >@X 20 SCT 2FH AFT 00Z. 02Z >@X 8
 SCT 2FH VRBL C2X 1/4F. 05Z C1X 1/4F VRBL >@X C5 OVC 1F. 14Z >@X 5 SCT
 C12 OVC 4FH. 16Z MVFR CIG FH..
(((DAY 052222 C30 BKN 5H BKN V SCT CHC >@X 2FH. 02Z >@X 10 SCT 3FH OCNL
 C3X 1/4F. 05Z C1X1/8F VRBL >@X C5 OVC 1F. 14Z 5 SCT C12 BKN 4FH BKN V
 SCT. 16Z MVFR CIG FH..
(((FDY 052222 C25 BKN 5H BKN OCNL SCT CHC >@X 10 SCT 3F. 04Z >@X 10 SCT
 3F OCNL C4X 3/4F. 07Z C2X 1/4F VRBL >@X 10 SCT 11/2F.14Z 6 SCT C14 BKN
 4FH BKN V SCT. 16Z MVFR CIG FH..
(((CLE 052222 C20 BKN 5H BKN V SCT CHC >@X 3F AFT 01Z. 03Z >@X 10 SCT
 3FH OCNL C7X 3/4F. 07Z C3X 1/4F VRBL >@X 8 SCT 2F. 14Z 6 SCT C14 BKN
 4FH BKN V SCT. 16Z MVFR CIG FH..
(((YNG 052222 35 SCT 6H SCT OCNL BKN  CHC 3FH. 03Z >@X 10 SCT 3FH VRBL
 C3X 1/4F. 08Z C2X 1/4F VRBL >@X 8 SCT 2F. 14Z 5 SCT C12 BKN 4FH. 16Z
 MVFR CIG H..
(((ZZV 052222 C30 BKN 5H BKN OCNL SCT CHC >@X 25 SCT 2F. 02Z >@X 20 SCT
 3F VRBL C3X 1/2F. 07Z C2X 1/4F VRBL >@X 6 SCT C10 OVC 2F. 14Z 3 SCT C8
 OVC 2F OCNL C12 BKN 4F. 16Z IFR CIG FH BCMG MVFR CIG F H BY 18Z..
NNNN<<<<<<<<<<<<
```

```
C           TAPE.FR
C
C       WRITTEN AUGUST 12, 1982 BY ALAN BLACKBURN WSFO BUFFALO
C
C       THIS PROGRAM READS A FILE FROM THE AFOS DATA BASE AND PUTS IT
C       INTO A FORM PUNCHABLE BY ARC:PUNCH BY CHANGING $ OR
C       * INTO {{{ ; + INTO " ; AND - INTO >@<.  IT ALSO KEEPS LINES
C       TO A MAXIMUM OF 72 CHARACTERS.  IF THE FILE IS WRKFTA OR
C       WRKTWB IT PUTS ON THE APPROPRIATE HEADER.
C
C       OUTPUT IS IN THE ORIGINAL FILE AND AN RDOS FILE NAMED PUNCHOUT
C
C       LOAD LINE:
C         RLDR TAPE.RB AFREAD.LB BG.LB UTIL.LB FORT.LB AFOSE.LB
C
C       SAMPLE COMMAND:  RUN:TAPE BUFWRKFTA
C             THE PROGRAM CAN ONLY BE RENAMED TO 4 DIGIT NAMES.
C
      COMMON/TPCOM/IHDR(11),IHDRF(33),IHDRT(33)
C
      DIMENSION IBPTR(2),IPROD(5),IPRODU(10),IHDR2(33),IXU(170),IXP(85)
     2,IXU2(150),IU(80),IP(40)
C
      DATA IHDR/"BUFWRKFTA000",177777K,177777K,"50",142600K,6412K/
      DATA IHDRF/"<@@@@@@<SSE@@EE<EE@@VSSV@@@@@@",6412K,
     2"<<<<<<<<<<<<ZCZC{{}KBUF Z FT}{{{",6412K/
      DATA IHDRT/"<@@@@@@EE<EE@@KT TK@@<YYR@@@@@",6412K,
     2"<<<<<<<<<<<ZCZC{{}KBUF Z TWEB}{{{",6412K/
C
C.......GET NAME OF PRODUCT FROM ICE2
      CALL GCHN(IC,IER)
      CALL OPENN(IC,"ICE2.CM",0,IER)
      I1=0
      IBPTR(1)=0
      IBPTR(2)=5
      CALL SPOS(IC,IBPTR,IER)
      IPROD(5)=0
      CALL RDS(IC,IPROD,9,IER)
      CALL KLOSE(IC,IER)
C
C.......REWRITE HEADERS
      DO 10 J=1,4
10    IHDR(J)=IPROD(J)
      CALL UNPACK(IPROD,10,IPRODU)
      IHDR(5)=IOR(60K,ISHFT(IPRODU(9),8))
C.......CHECK IF FTA
      IF(IPRODU(7).EQ.70.AND.IPRODU(8).EQ.84.AND.IPRODU(9).EQ.65)
     2GOTO 100
C.......CHECK IF TWB
      IF(IPRODU(7).EQ.84.AND.IPRODU(8).EQ.87.AND.IPRODU(9).EQ.66)
     2GOTO 200
      GOTO 300
C.......SET FTA HEADERS
100   I1=1
      IHDRF(27)=IPROD(1)
      IHDRF(28)=IOR(IAND(177400K,IPROD(2)),40K)
      GOTO 300
C.......SET TWB HEADERS
200   I1=2
      IHDRT(26)=IPROD(1)
```

```
      IHDRT(27)=IOR(IAND(177400K,IPROD(2)),40K)
C
300   CONTINUE
C.....GET PUNCHOUT READY
      CALL GCHN(IC2,IER)
      CALL DFILW("PUNCHOUT",IER)
      CALL CRAND("PUNCHOUT",IER)
      CALL OPENN(IC2,"PUNCHOUT",0,IER)
C.....WRITE HEADERS TO PUNCHOUT
      CALL WRS(IC2,IHDR,22,IER)
      IF(I1.EQ.0)GOTO 306
      IF(I1.EQ.1)GOTO 304
      CALL WRS(IC2,IHDRT,66,IER)
      GOTO 306
304   CALL WRS(IC2,IHDRF,66,IER)
C
C.....READ PRODUCT
306   CALL AFREAD(1,IPROD,$900)
307   CALL AFREAD(2,IP,$700,$800)
      CALL UNPACK(IP,80,IU)
C.....FIND THE # OF CHARACTERS ON THE LINE THAT HAS BEEN READ
      DO 120 L=1,78
C.....LOOK FOR 3 CONSECUTIVE SPACES
      IF(IU(L).EQ.32.AND.IU(L+1).EQ.32.AND.IU(L+2).EQ.32)GOTO 130
120   CONTINUE
      GOTO 920
130   IF(L.EQ.1)GOTO 307
      N=L-1
C.....N IS # OF CHARACTERS ON THE LINE THAT HAS BEEN READ
C.....N+NX IS # OF CHARACTERS ON THE EXPANDED LINE
      NX=0
C.....LOOK FOR $ OR *
      IF(IU(1).NE.36.AND.IU(1).NE.42)GOTO 411
      NX=NX-1
C
C.....EXPAND AND EDIT THE LINE
312   CONTINUE
      IF(N.LT.2)GOTO 410
      DO 320 J=2,N
      IF(IU(J).EQ.43)GOTO 330    ; PLUS SIGN
      IF(IU(J).EQ.45)GOTO 340    ; MINUS SIGN
      IXU(J+NX)=IU(J)
320   CONTINUE
      GOTO 400
330   IXU(J+NX)=34               ; " WRITTEN FOR +
      GOTO 320
340   IXU(J+NX)=62
      IXU(J+NX+1)=64             ; >@< WRITTEN FOR -
      IXU(J+NX+2)=60
      NX=NX+2
      GOTO 320
400   CONTINUE
      IF((N+NX).GT.72)GOTO 450   ; IS LINE TOO LONG?
C
C.....FINISH AND WRITE LINE
      IXU(N+NX+1)=15K
      IXU(N+NX+2)=12K
      CALL PACK(IXU,(N+NX+2),IXP)
      CALL WRS(IC2,IXP,(N+NX+2),IER)
      NX=0
```

```
C
C.......READ NEXT LINE
410    CALL AFREAD(2,IP,$700,$800)
411    CALL UNPACK(IP,80,IU)
       DO 220 J=1,78
       IF(IU(J).EQ.32.AND.IU(J+1).EQ.32.AND.IU(J+2).EQ.32)GOTO 230
220    CONTINUE
       GOTO 920
230    IF(J.EQ.1)GOTO 410
       N=J-1
C.......CHECK IF FT IS INDENTED
       IF(NX.LE.0)GOTO 250
       IF(I1.NE.1)GOTO 250
       IF(IU(1).NE.32)GOTO 250
       NX=NX-1
       GOTO 312
250    CONTINUE
C.......LOOK FOR $ OR *
       IF(IU(1).EQ.36.OR.IU(1).EQ.42)GOTO 420
C.......LOOK FOR NNNN
       IF(IU(1).EQ.78.AND.IU(2).EQ.78.AND.IU(3).EQ.78.AND.IU(4).EQ.78)
      2GOTO 700
       DO 412 J=1,N
       IF(IU(J).EQ.43)GOTO 414
       IF(IU(J).EQ.45)GOTO 416
       IXU(J+NX)=IU(J)
412    CONTINUE
       GOTO 400
414    IXU(J+NX)=34
       GOTO 412
416    IXU(J+NX)=62
       IXU(J+NX+1)=64
       IXU(J+NX+2)=60
       NX=NX+2
       GOTO 412
C
C.......START OF NEXT FT OR TWEB
420    IF(NX.GT.0)GOTO 500              ; TEXT IS LEFTOVER
425    IXU(1)=123                       ; PUT ((( AT START OF LINE
       IXU(2)=123
       IXU(3)=60
       NX=2
       GOTO 312
C
C.......LOOK FOR LAST SPACE BEFORE CHARACTER #72 AND PUT THE REST OF
C          OF THE TEXT AT THE BEGINNING OF THE NEXT LINE.
450    N2=73
452    IF(IXU(N2).EQ.32)GOTO 454
       N2=N2-1                          ; N2 WILL BE # OF SPACE CHAR.
       IF(N2.EQ.0)GOTO 910
       GOTO 452
454    IF(I1.NE.1)GOTO 470
       K=N+NX
       DO 460 J=N2,K                    ; PRODUCT IS FT -- INDENT 1 SPACE
460    IXU2(J-N2+1)=IXU(J)              ; PUT LEFTOVERS IN IXU2
       NX=(N+NX)-N2+2                   ; NEW NX TO ADD TO NEXT LINE
       GOTO 480
470    K1=N2+1
       K=N+NX
       DO 475 J=K1,K                    ; PRODUCT IS NOT FT
```

```
475     IXU2(J-N2)=IXU(J)
        NX=(N+NX)-(N2+1)+2
480     IXU2(NX)=32                          ; SPACE AT END OF LEFTOVERS
        IF(IXU2(NX-1).NE.46)GOTO 485
        IXU2(NX+1)=32                        ; ADD SPACE AFTER PERIOD
        NX=NX+1
C
C.......FINISH AND WRITE LINE
485     IXU(N2)=15K
        IXU(N2+1)=12K
        CALL PACK(IXU,(N2+1),IXP)
        CALL WRS(IC2,IXP,(N2+1),IER)
        DO 490 J=1,NX
490     IXU(J)=IXU2(J)
        IF((NX+1).GT.72)GOTO 450
        GOTO 410
C
C.......FINISH THE LEFTOVERS BEFORE STARTING A NEW LINE
500     IXU(NX+1)=15K
        IXU(NX+2)=12K
        CALL PACK(IXU,(NX+2),IXP)
        CALL WRS(IC2,IXP,(NX+2),IER)
        GOTO 425
C
C.......OUT OF TEXT -- FINISH THE PRODUCT
700     CONTINUE
C.......PUT NNNN ON THE END
        IXP(1)=47116K
        IXP(2)=47116K
        DO 710 J=1,6
710     IXP(J+2)=36074K
        IXP(9)=6412K
        CALL WRS(IC2,IXP,18,IER)
C.......PUT OCTAL 203 ON THE END
        IXP(1)=101603K
        CALL WRS(IC2,IXP,2,IER)
        CALL KLOSE(IC2,IER)
C
C.......STORE THE PRODUCT
        CALL FSTORE("PUNCHOUT",0,IER)
        CALL FORKO("TAPE",IPROD,IER)
        STOP
C
C.......ERRORS
800     CALL FORKE("TAPE","CANT READ FILE",IER)
825     CALL KLOSE(IC2,IER)
        STOP
900     CALL FORKE("TAPE","CANT OPEN FILE",IER)
        GOTO 825
910     CALL FORKE("TAPE","ERROR 910",IER)
        GOTO 825
920     CALL FORKE("TAPE","LINE TOO LONG",IER)
        GOTO 825
        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