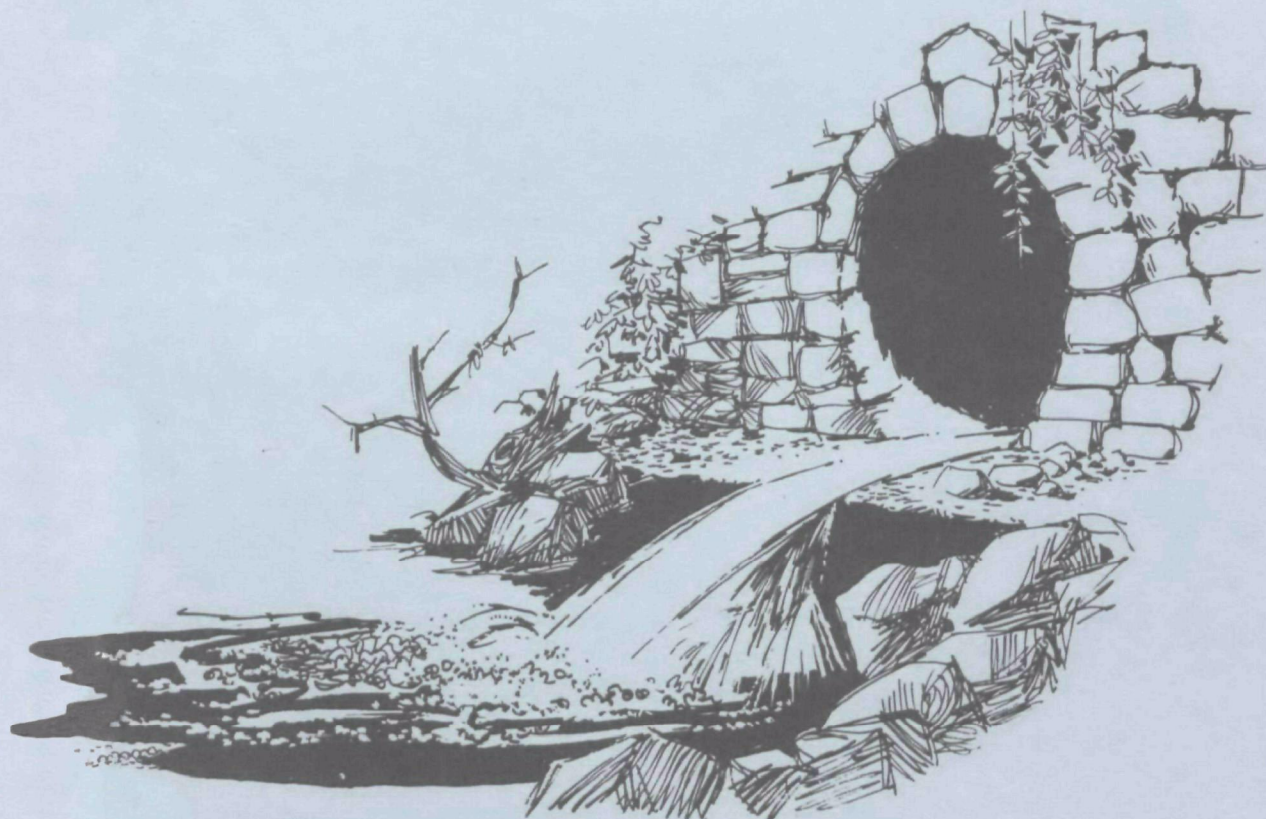


# Storm Water Management Model

*Volume IV—Program Listing*



## WATER POLLUTION CONTROL RESEARCH SERIES

The Water Pollution Control Research Reports describe the results and progress in the control and abatement of pollution of our Nation's waters. They provide a central source of information on the research, development and demonstration activities of the Water Quality Office of the Environmental Protection Agency, through in-house research and grants and contracts with the Federal, State and local agencies, research institutions, and industrial organizations.

Previously issued reports on the Storm and Combined Sewer Pollution Control Program:

|                 |                                                                                                          |
|-----------------|----------------------------------------------------------------------------------------------------------|
| 11023 FDB 09/70 | Chemical Treatment of Combined Sewer Overflows                                                           |
| 11024 FKJ 10/70 | In-Sewer Fixed Screening of Combined Sewer Overflows                                                     |
| 11024 EJC 10/70 | Selected Urban Storm Water Abstracts, First Quarterly Issue                                              |
| 11023 — 12/70   | Urban Storm Runoff and Combined Sewer Overflow Pollution                                                 |
| 11023 DZF 06/70 | Ultrasonic Filtration of Combined Sewer Overflows                                                        |
| 11024 EJC 01/71 | Selected Urban Runoff Abstracts, Second Quarterly Issue                                                  |
| 11020 FAQ 03/71 | Dispatching System for Control of Combined Sewer Losses                                                  |
| 11022 EFF 12/70 | Prevention and Correction of Excessive Infiltration and Inflow into Sewer Systems - A Manual of Practice |
| 11022 EFF 01/71 | Control of Infiltration and Inflow into Sewer Systems                                                    |
| 11022 DPP 10/70 | Combined Sewer Temporary Underwater Storage Facility                                                     |
| 11024 EQG 03/71 | Storm Water Problems and Control in Sanitary Sewers - Oakland and Berkeley, California                   |
| 11020 FAL 03/71 | Evaluation of Storm Standby Tanks - Columbus, Ohio                                                       |
| 11024 FJE 04/71 | Selected Urban Storm Water Runoff Abstracts, Third Quarterly Issue                                       |

To be continued on inside back cover...

# STORM WATER MANAGEMENT MODEL

Volume IV Program Listing

by

Metcalf & Eddy, Inc., Palo Alto, California  
University of Florida, Gainesville, Florida  
Water Resources Engineers, Inc., Walnut Creek, California

for the

ENVIRONMENTAL PROTECTION AGENCY

Contract No. 14-12-501 Project No. 11024EBI  
Contract No. 14-12-502 Project No. 11024DOC  
Contract No. 14-12-503 Project No. 11024EBJ

October 1971

EPA REVIEW NOTICE

This report has been reviewed by the Environmental Protection Agency and approved for publication. Approval does not signify that the contents necessarily reflect the views and policies of the Environmental Protection Agency, nor does mention of trade names or commercial products constitute endorsement or recommendation for use.



## ABSTRACT

A comprehensive mathematical model, capable of representing urban storm water runoff, has been developed to assist administrators and engineers in the planning, evaluation, and management of overflow abatement alternatives.

Hydrographs and pollutographs (time varying quality concentrations or mass values) were generated for real storm events and systems from points of origin in real time sequence to points of disposal (including travel in receiving waters) with user options for intermediate storage and/or treatment facilities. Both combined and separate sewerage systems may be evaluated. Internal cost routines and receiving water quality output assisted in direct cost-benefit analysis of alternate programs of water quality enhancement.

Demonstration and verification runs on selected catchments, varying in size from 180 to 5,400 acres, in four U.S. cities (approximately 20 storm events, total) were used to test and debug the model. The amount of pollutants released varied significantly with the real time occurrence, runoff intensity duration, pre-storm history, land use, and maintenance. Storage-treatment combinations offered best cost-effectiveness ratios.

A user's manual and complete program listing were prepared.

This report was submitted in fulfillment of Projects 11024 EBI, DOC, and EBJ under Contracts 14-12-501, 502, and 503 under the sponsorship of the Environmental Protection Agency.

The titles and identifying numbers of the final report volumes are:

| <u>Title</u>                                                         | <u>EPA Report No.</u> |
|----------------------------------------------------------------------|-----------------------|
| STORM WATER MANAGEMENT MODEL<br>Volume I - Final Report              | 11024 DOC 07/71       |
| STORM WATER MANAGEMENT MODEL<br>Volume II - Verification and Testing | 11024 DOC 08/71       |
| STORM WATER MANAGEMENT MODEL<br>Volume III - User's Manual           | 11024 DOC 09/71       |
| STORM WATER MANAGEMENT MODEL<br>Volume IV - Program Listing          | 11024 DOC 10/71       |

## CONTENTS

| <u>Section</u> |                       | <u>Page</u> |
|----------------|-----------------------|-------------|
|                | Typical JCL           | 1           |
| 1              | Executive Block       | 3           |
| 2              | Runoff Block          | 13          |
| 3              | Transport Block       | 41          |
| 4              | Storage Block         | 125         |
| 5              | Receiving Water Block | 201         |

# TYPICAL JCL

\*\*\*\*\*

THIS IS TYPICAL JCL FOR EXECUTING A RUN USING THE COMPLETE MODEL  
(JCL VARIATIONS MAY OCCUR ACCORDING TO MACHINE OR INSTALLATION)

\*\*\*\*\*

((THE FOLLOWING ARE THE ASSUMED TAPE/DISK ASSIGNMENTS FOR EACH BLOCK))

| BLOCK     | INPUT<br>FILE | OUTPUT<br>FILE |
|-----------|---------------|----------------|
| -----     | -----         | -----          |
| RUNOFF    | 0             | 8              |
| TRANSPORT | 8             | 9              |
| STORAGE   | 9             | 10             |
| RECEIVING | 10            | 11             |
| GRAPH     | 9             | 0              |

\*\*\*\*\*

//STORM EXEC FORTHCLG,PARM.FORT=MAP  
//FORT.SYSIN DD\*

\*\*\*\*\* FORTRAN PROGRAM INSERTED HERE \*\*\*\*\*

```
/*
//GO.FT08F001 DD DSN= C802.RUNOFF,UNIT=2314,DISP=(NEW,KEEP,DELETE),
//              SPACE=(TRK,(10,2),RLSE)
//GO.FT09F001 DD DSN= C802.TRANSPORT,UNIT=2314,DISP=(NEW,KEEP,DELETE),
//              SPACE=(TRK,(10,2),RLSE)
//GO.FT10F001 DD DSN= C802.STORAGE,UNIT=2314,DISP=(NEW,KEEP,DELETE),
//              SPACE=(TRK,(10,2),RLSE)
//GO.FT11F001 DD DSN= C802.RECEIVING,UNIT=2314,DISP=(NEW,KEEP,DELETE),
//              SPACE=(TRK,(10,2),RLSE)
//GO.FT01F001 DD UNIT=2314,DISP=NEW,SPACE=(TRK,(10,2),RLSE),
//              DCB=(RECFM=VS,LRECL=796,BLKSIZE=800)
//GO.FT02F001 CC UNIT=2314,DISP=NEW,SPACE=(TRK,(10,2),RLSE),DCB=*.FT01F001
//GO.FT03F001 CC UNIT=2314,DISP=NEW,SPACE=(TRK,(10,2),RLSE),DCB=*.FT01F001
//GO.FT13F001 CC UNIT=2314,DISP=NEW,SPACE=(TRK,(10,2),RLSE),DCB=*.FT01F001
//GO.SYSYSIN DD*
```

\*\*\*\*\* DATA CARDS INSERTED HERE \*\*\*\*\*

/\*

\*\*\*\*\*

## Section 1

### EXECUTIVE BLOCK

|                  | <u>Page</u> |
|------------------|-------------|
| Main Program     | 5           |
| Subroutine GRAPH | 7           |
| Subroutine CURVE | 9           |
| Subroutine PINE  | 11          |
| Subroutine PPLOT | 12          |

|     |                                                                        |          |
|-----|------------------------------------------------------------------------|----------|
|     | STORMWATER MAIN PROGRAM                                                | MAIN 1   |
|     |                                                                        | MAIN 2   |
|     | DIMENSION PNAME (6,2),TITLE1(10),STORM(4),RAIN(4)                      | MAIN 3   |
|     | COMMON /TAPES/ INCNT,IOUTCT,JIN(10),JOUT(10),NSCRAT(5)                 | MAIN 4   |
|     | COMMON CNAME(2)                                                        | MAIN 5   |
|     | DATA PNAME / 4HWATE, 4HTRAN, 4HRECE, 4HENDP                            | MAIN 6   |
|     | 1 : 4HSTOR, 4HGRAP, 4HRSHE, 4HSPOR, 4HIVIN, 4HROGR                     | MAIN 7   |
|     | 2 : 4HACE, 4HH                                                         | MAIN 8   |
|     | N5=5                                                                   | MAIN 9   |
|     | N6=6                                                                   | MAIN 10  |
|     | INCNT=0                                                                | MAIN 11  |
|     | IOUTCT=0                                                               | MAIN 12  |
|     | READ(05,399)TITLE1                                                     | MAIN 13  |
| 399 | FORMAT(10A4)                                                           | MAIN 14  |
|     | READ(05,400)NSERYS,ACRES,ADDWF,NDESYR,DESFLO,NSTRMS,QTRUNK             | MAIN 15  |
| 400 | FORMAT(I5,F10.1,F10.2,I5,F10.1,I5,F10.1)                               | MAIN 16  |
|     | WRITE(06,401)NSERYS,TITLE1,ACRES,ADDWF,NDESYR,DESFLO,QTRUNK            | MAIN 17  |
| 401 | FORMAT('1'///26X,'FEDERAL WATER QUALITY ADMINISTRATION',               | MAIN 18  |
|     | *10X,'CONTRACTS 14-12-501'//26X,'STORMWATER MANAGEMENT PROJECT',       | MAIN 19  |
|     | *27X,'14-12-502'//82X,'14-12-503'///26X,'METCALF & EDDY, INC'//        | MAIN 20  |
|     | *26X,'WATER RESOURCES ENGINEERS, INC'//26X,'UNIVERSITY OF',            | MAIN 21  |
|     | *' FLORIDA'///31X,'DEMONSTRATION SERIES NO.',I3/'0',30X,               | MAIN 22  |
|     | *10A4/31X,'COMBINED SEWER AREA OF',F8.2,'ACRES'//31X,'AVERAGE DAILY    | MAIN 23  |
|     | *DRY WEATHER FLOW =' ,F8.2,'CFS'//31X,I2,'-YEAR DESIGN FLOW =' ,F10.2, | MAIN 24  |
|     | *'CFS'//31X,'AVAILABLE MAX. TRUNK CAPACITY =' ,F10.2,'CFS'//31X,'STORM | MAIN 25  |
|     | *MS STUDIED: TOTAL RAINFALL, INCHES')                                  | MAIN 26  |
|     | DO 500 J=1,NSTRMS                                                      | MAIN 27  |
|     | READ(05,402)STORM,RAIN                                                 | MAIN 28  |
| 402 | FORMAT(4A4,4A4)                                                        | MAIN 29  |
|     | WRITE(06,403)STORM,RAIN                                                | MAIN 30  |
| 403 | FORMAT(30X,4A4,16X,4A4)                                                | MAIN 31  |
| 500 | CONTINUE                                                               | MAIN 32  |
|     | WRITE(06,106)                                                          | MAIN 33  |
|     | READ(N5,100) (JIN(J),JOUT(J),J=1,10)                                   | MAIN 34  |
|     | WRITE(N6,108) JIN,JOUT                                                 | MAIN 35  |
|     | READ(N5,100) (NSCRAT(I),I=1,5)                                         | MAIN 36  |
|     | WRITE(N6,108) (NSCRAT(I),I=1,5)                                        | MAIN 37  |
| 220 | CONTINUE                                                               | MAIN 38  |
|     | READ(N5,102) CNAME                                                     | MAIN 39  |
|     | WRITE(N6,106) CNAME                                                    | MAIN 40  |
|     | DO 240 I = 1,6                                                         | MAIN 41  |
|     | IF(CNAME(I).NE.PNAME(I,1).OR.CNAME(2).NE.PNAME(I,2)) GO TO 240         | MAIN 42  |
|     | GO TO (260,280,300,340,360,380),I                                      | MAIN 43  |
| 240 | CONTINUE                                                               | MAIN 44  |
|     | WRITE(N6,110) CNAME                                                    | MAIN 45T |
|     | STOP 1000                                                              | MAIN 45A |
| 260 | CALL RUNOFF                                                            | MAIN 46  |
|     | GO TO 220                                                              | MAIN 47  |
| 280 | CALL TRANS                                                             | MAIN 48  |
|     | GO TO 220                                                              | MAIN 49  |
| 300 | CALL RECEIV                                                            | MAIN 50  |
|     | GO TO 220                                                              | MAIN 51  |
| 340 | WRITE(N6,104)                                                          | MAIN 52  |
|     | STOP                                                                   | MAIN 53  |
| 360 | CALL STORAG                                                            | MAIN 54  |
|     | GO TO 220                                                              | MAIN 55  |
| 380 | CONTINUE                                                               | MAIN 56  |
|     | CALL GRAFH(1)                                                          | MAIN 57  |
|     | GO TO 220                                                              | MAIN 58  |
| 100 | FORMAT(20I4)                                                           | MAIN 59  |

|                                                  |          |
|--------------------------------------------------|----------|
| 102 FORMAT(2CA4)                                 | MAIN 60  |
| 104 FORMAT('1 STORMWATER SIMULATION ENDED')      | MAIN 61  |
| 106 FORMAT(1F1,2A4)                              | MAIN 62  |
| 108 FORMAT('0 TAPE ASSIGNMENTS'/(10I10))         | MAIN 63  |
| 110 FORMAT('C CCRRECT CNAME NOT FOUND - - ',2A4) | MAIN 63T |
| END                                              | MAIN 64  |



|                                                                            |          |
|----------------------------------------------------------------------------|----------|
| SUBROUTINE GRAPH(IC)                                                       | GRAP 1   |
| COMMON /TAPES/ INCNT, IOUTCT, JIN(10), JOUT(10), NSCRAT(5)                 | GRAP 2   |
| COMMON DUMMY(50)                                                           | GRAP 3   |
| COMMON X(201,5), Y(201,5), NLOC(100), YT(160,150), NPT(100)                | GRAP 4X  |
| 1, ITAB(101), IPLGT(200)                                                   | GRAP 5   |
| COMMON/LAB/ TITL(18), XLAB(11), YLAB(6), HORIZ(20), VERT(7,6), IT          | GRAP 6X  |
| IF(IC-1) 205, 210, 210                                                     | GRAP 7   |
| 205 READ(5,100) NPLCT, NPCV, NQP                                           | GRAP 8   |
| IF(NPLCT.EQ.0) RETURN                                                      | GRAP 9   |
| NLP=NCP+1                                                                  | GRAP 10  |
| READ(5,100) (IPLCT(M), M=1, NPLCT)                                         | GRAP 11  |
| WRITE(6,106) NPLCT, (IPLCT(N), N=1, NPLCT)                                 | GRAP 12  |
| 106 FORMAT(' HYDROGRAPHS WILL BE PLOTTED FOR THE FOLLOWING', I5,           | GRAP 13  |
| 1' PGINTS'/(10I1C))                                                        | GRAP 14  |
| IF(NPCV.LT.1) NPCV=1                                                       | GRAP 15  |
| NTAPE=JOUT(IOUTCT)                                                         | GRAP 16  |
| GO TO 220                                                                  | GRAP 17  |
| 210 CONTINUE                                                               | GRAP 18  |
| READ(5,100) NTAPE, NPCV, NQP, NPLCT                                        | GRAP 19  |
| NLP=NCP+1                                                                  | GRAP 20  |
| IF(NPCV.EQ.0) NPCV=5                                                       | GRAP 21  |
| WRITE(6,102) NTAPE, NPCV                                                   | GRAP 22  |
| IF(NPLCT.LT.1) GO TO 212                                                   | GRAP 23  |
| READ(5,100) (IPLCT(N), N=1, NPLCT)                                         | GRAP 24  |
| WRITE(6,106) NPLCT, (IPLCT(N), N=1, NPLCT)                                 | GRAP 25  |
| 212 CONTINUE                                                               | GRAP 26  |
| 100 FORMAT(16I5)                                                           | GRAP 27  |
| 102 FORMAT('1 OUTPUT WILL BE PLOTTED FOR TAPE', I5, ' IN BLOCKS OF         | GRAP 28  |
| 1', I5)                                                                    | GRAP 29  |
| READ(5,104) TITL                                                           | GRAP 30  |
| READ(5,104) HORIZ                                                          | GRAP 31  |
| DO 1 JJJ=1, NLP                                                            | GRAP 31A |
| READ(5,104) (VERT(II, JJJ), II=1, 7)                                       | GRAP 32X |
| 1 CONTINUE                                                                 | GRAP 32A |
| 220 CONTINUE                                                               | GRAP 33  |
| REWIND NTAPE                                                               | GRAP 34  |
| READ(NTAPE) TITL                                                           | GRAP 35  |
| READ(NTAPE) NSTEPS, NCURVE, NQUAL, TDELT, TZERO, TAREA                     | GRAP 36X |
| WRITE(6,7091) NSTEPS, NCURVE, NQUAL, TDELT, TZERO, TAREA                   | GRAP 37X |
| 7091 FORMAT(3I1C, 3F10.1)                                                  | GRAP 38X |
| NVAL=NCURVE*(NQUAL+1)                                                      | GRAP 39  |
| IF(NVAL.LE.150) GO TO 402                                                  | GRAP 40X |
| WRITE(6,400) NVAL                                                          | GRAP 41  |
| 400 FORMAT(' ', 'THE GRAPH SUBROUTINE IS ASKING FOR MORE OUTPUT FROM TGRAP | GRAP 42  |
| *APE THAN THE DIMENSION OF YT(160,150) ALLOWS.'/ ' ', 'A VALUE OF ', GRAP  | GRAP 43X |
| *14, 'HAS BEEN REQUESTED. THIS ERROR RESULTED IN A SYSTEM BLOW-UP AGRAP    | GRAP 44  |
| *T STANFORD CN FORTRAN H, JUNE 1970. DAS'////)                             | GRAP 45  |
| STOP                                                                       | GRAP 46  |
| 402 CONTINUE                                                               | GRAP 47  |
| READ(NTAPE) (NLOC(M), M=1, NCURVE)                                         | GRAP 48  |
| WRITE(6,7090) (NLOC(M), M=1, NCURVE)                                       | GRAP 49  |
| IF(NPLCT-1) 245, 225, 225                                                  | GRAP 50  |
| 225 K=1                                                                    | GRAP 51  |
| DO 240 M=1, NPLCT                                                          | GRAP 52  |
| DO 230 J=1, NCURVE                                                         | GRAP 53  |
| IF(IPLCT(M)-NLOC(J)) 230, 235, 230                                         | GRAP 54  |
| 230 CONTINUE                                                               | GRAP 55  |
| 235 ITAB(K)=J                                                              | GRAP 56  |
| K=K+1                                                                      | GRAP 57  |
| 240 CONTINUE                                                               | GRAP 58  |

|                                      |          |
|--------------------------------------|----------|
| ITAB(K)=0                            | GRAP 59  |
| GO TO 260                            | GRAP 60  |
| 245 DO 250 J=1,NCURVE                | GRAP 61  |
| 250 ITAB(J)=J                        | GRAP 62  |
| ITAB(NCURVE+1)=0                     | GRAP 63  |
| 260 CONTINUE                         | GRAP 64  |
| MC=NSTEPS/100                        | GRAP 65  |
| IF(MC*100.NE.NSTEPS) MC=MC+1         | GRAP 66  |
| N=1                                  | GRAP 67  |
| NR=0                                 | GRAP 68  |
| DO 300 K=1,NSTEPS,MC                 | GRAP 69  |
| N=N+1                                | GRAP 70  |
| DO 290 J=1,MC                        | GRAP 71  |
| NR=NR+1                              | GRAP 72  |
| IF(NR.GT.NSTEPS) GO TO 290           | GRAP 73  |
| READ(NTAPE) TIMES,(YT(N,M),M=1,NVAL) | GRAP 74  |
| 290 CONTINUE                         | GRAP 75  |
| DO 295 M=1,5                         | GRAP 76  |
| X(N,M)=TIMES/3600.                   | GRAP 77  |
| 295 CONTINUE                         | GRAP 78  |
| 300 CONTINUE                         | GRAP 79  |
| DO 350 J=1,NLP                       | GRAP 80  |
| IT=J                                 | GRAP 80A |
| K=1                                  | GRAP 81  |
| L=ITAB(1)                            | GRAP 82  |
| DO 320 M=1,NCURVE,NPCV               | GRAP 83  |
| IF(L.EC.C) GO TO 330                 | GRAP 84  |
| DO 310 MM=1,NPCV                     | GRAP 85  |
| IF(J-1) 301,301,302                  | GRAP 86  |
| 301 LX=L                             | GRAP 87  |
| GO TO 303                            | GRAP 88  |
| 302 LX=NCURVE+(L-1)*NCUAL+J-1        | GRAP 89  |
| 303 NPT(MM)=N                        | GRAP 90  |
| X(1,MM)=TZERO /3600.                 | GRAP 91  |
| Y(1,MM)=0.                           | GRAP 92  |
| DO 305 MM=2,N                        | GRAP 93  |
| 305 Y(MM,MM)=YT(MM,LX)               | GRAP 94  |
| K=K+1                                | GRAP 95  |
| L=ITAB(K)                            | GRAP 96  |
| IF(L) 315,315,310                    | GRAP 97  |
| 310 CONTINUE                         | GRAP 98  |
| NCV=NPCV                             | GRAP 99  |
| GO TO 318                            | GRAP100  |
| 315 NCV=MM                           | GRAP101  |
| 318 CONTINUE                         | GRAP102  |
| ILAB=ITAB(K-1)                       | GRAP103  |
| CALL CURVE(X,Y,NPT,NCV,NLOC(ILAB))   | GRAP104  |
| 320 CONTINUE                         | GRAP105  |
| 330 CONTINUE                         | GRAP106  |
| 350 CONTINUE                         | GRAP107  |
| 104 FORMAT(20A4)                     | GRAP108  |
| 7090 FORMAT(10I10)                   | GRAP109  |
| RETURN                               | GRAP110  |
| END                                  | GRAP111  |

|                                        |         |
|----------------------------------------|---------|
| SUBROUTINE CURVE (X,Y,NPT,NCV,NPLOT)   | CURV 1  |
| DIMENSION X(201,10),Y(201,10),NPT(10)  | CURV 2  |
| COMMON/LAB/ TITLE(18),XLAB(11),YLAB(6) | CURV 3  |
| L,HORIZ(20),VERT(7,6),IT               | CURV 4X |
| XMAX=X(1,1)                            | CURV 5  |
| XMIN=XMAX                              | CURV 6  |
| YMIN=Y(1,1)                            | CURV 7  |
| YMAX=YMIN                              | CURV 8  |
| DO 205 L=1,NCV                         | CURV 9  |
| NPTM=NPT(L)                            | CURV 10 |
| IF(NPTM.EQ.0) GO TO 205                | CURV 11 |
| DO 204 N=1,NPTM                        | CURV 12 |
| IF(X(N,L).LT.XMIN) XMIN=X(N,L)         | CURV 13 |
| IF(X(N,L).GT.XMAX) XMAX=X(N,L)         | CURV 14 |
| IF(Y(N,L).LT.YMIN) YMIN=Y(N,L)         | CURV 15 |
| IF(Y(N,L).GT.YMAX) YMAX=Y(N,L)         | CURV 16 |
| 204 CONTINUE                           | CURV 17 |
| 205 CONTINUE                           | CURV 18 |
| RANGE=(YMAX-YMIN)/5.                   | CURV 19 |
| IF(RANGE.GT.0.) GO TO 2059             | CURV 20 |
| IF(YMAX.GT.0.) YMIN=0.                 | CURV 21 |
| IF(YMAX.LT.0.) YMAX=0.                 | CURV 22 |
| RANGE=(YMAX-YMIN)/5.                   | CURV 23 |
| 2059 CONTINUE                          | CURV 24 |
| A=ALOG10(RANGE)                        | CURV 25 |
| IF(A.LT.0.) GO TO 220                  | CURV 26 |
| N=A                                    | CURV 27 |
| RANGE=RANGE/(10.**N)                   | CURV 28 |
| L=RANGE+1.001                          | CURV 29 |
| 206 CONTINUE                           | CURV 30 |
| IF(L.EQ.2) GO TO 209                   | CURV 31 |
| IF(L.GT.4) GO TO 207                   | CURV 32 |
| L=4                                    | CURV 33 |
| 207 IF(L.GT.5) L=10                    | CURV 34 |
| 209 CONTINUE                           | CURV 35 |
| FRANG=L*10.**N                         | CURV 36 |
| GO TO 240                              | CURV 37 |
| 220 M=A-0.5999                         | CURV 38 |
| N=-M                                   | CURV 39 |
| RANGE=RANGE*10.**N                     | CURV 40 |
| L=RANGE+1.001                          | CURV 41 |
| 226 CONTINUE                           | CURV 42 |
| IF(L.EQ.2) GO TO 229                   | CURV 43 |
| IF(L.GT.4) GO TO 227                   | CURV 44 |
| L=4                                    | CURV 45 |
| 227 CONTINUE                           | CURV 46 |
| IF(L.GT.5) L=10                        | CURV 47 |
| 229 CONTINUE                           | CURV 48 |
| FRANG=L/10.**N                         | CURV 49 |
| 240 CONTINUE                           | CURV 50 |
| K=YMIN/FRANG                           | CURV 51 |
| IF(YMIN.LT.0.) K=K-1                   | CURV 52 |
| IF(YMAX.LE.(K+5)*FRANG) GO TO 250      | CURV 53 |
| L=L+1                                  | CURV 54 |
| IF(L.LT.11) GO TO 245                  | CURV 55 |
| L=2                                    | CURV 55 |
| N=N+1                                  | CURV 57 |
| IF(A.LE.0.) N=N-2                      | CURV 58 |
| 245 CONTINUE                           | CURV 59 |
| IF(A) 226,206,206                      | CURV 60 |

|                                |         |
|--------------------------------|---------|
| 250 YMIN=K*FRANG               | CURV 61 |
| YMAX=(K+5)*FRANG               | CURV 62 |
| XSCAL=100./(XMAX-XMIN)         | CURV 63 |
| YSCAL=50./(YMAX-YMIN)          | CURV 64 |
| XINT=(XMAX-XMIN)/10.           | CURV 65 |
| YINT=(YMAX-YMIN)/5.            | CURV 66 |
| XLAB(1) =XMIN                  | CURV 67 |
| DO 260 N=1,10                  | CURV 68 |
| XLAB(N+1)=XLAB(N)+XINT         | CURV 69 |
| 260 CONTINUE                   | CURV 70 |
| YLAB(6)=YMIN                   | CURV 71 |
| DO 270 N=1,5                   | CURV 72 |
| 270 YLAB(6-N)=YLAB(7-N)+YINT   | CURV 73 |
| CALL PPLCT(0,0,100,NPLCT)      | CURV 74 |
| K = 1                          | CURV 75 |
| DO 450 L=1,NCV                 | CURV 76 |
| IF(NPT(L).EQ.0) GO TO 440      | CURV 77 |
| XO=XSCAL*(X(1,L)-XMIN)         | CURV 78 |
| YO=YSCAL*(Y(1,L)-YMIN)         | CURV 79 |
| NPOINT = NPT(L)                | CURV 80 |
| DO 400 N = 2,NPOINT            | CURV 81 |
| XT = XSCAL*(X(N,L) - XMIN)     | CURV 82 |
| YT = YSCAL*(Y(N,L) - YMIN)     | CURV 83 |
| CALL PINE(XC,YO,XT,YT,K,NPLOT) | CURV 84 |
| XO = XT                        | CURV 85 |
| YO = YT                        | CURV 86 |
| 400 CONTINUE                   | CURV 87 |
| 420 CONTINUE                   | CURV 88 |
| 440 K = K + 1                  | CURV 89 |
| 450 CONTINUE                   | CURV 90 |
| CALL PPLCT(0,0,99,NPLCT)       | CURV 91 |
| RETURN                         | CURV 92 |
| END                            | CURV 93 |

|     |                                            |         |
|-----|--------------------------------------------|---------|
|     | SUBROUTINE PINE(X1,Y1,X2,Y2,NSYM,NCT)      | PINE 1  |
|     | AXA=X1                                     | PINE 2  |
|     | AXB=X2                                     | PINE 3  |
|     | AYA=Y1                                     | PINE 4  |
|     | AYB=Y2                                     | PINE 5  |
|     | N=1                                        | PINE 6  |
|     | IF(ABS(AXB-AXA).LT.ABS(AYB-AYA)) GO TO 290 | PINE 7  |
| C   |                                            | PINE 8  |
| C   | SET PARAMETERS FOR X DIRECTION             | PINE 9  |
| C   |                                            | PINE 10 |
|     | IF(AXB-AXA) 241,400,245                    | PINE 11 |
| 241 | CONTINUE                                   | PINE 12 |
|     | AXA=X2                                     | PINE 13 |
|     | AXB=X1                                     | PINE 14 |
|     | AYA=Y2                                     | PINE 15 |
|     | AYB=Y1                                     | PINE 16 |
| 245 | CONTINUE                                   | PINE 17 |
|     | IXA=AXA+.5                                 | PINE 18 |
|     | IXB=AXB+.5                                 | PINE 19 |
|     | IYA=AYA+.5                                 | PINE 20 |
|     | IYB=AYB+.5                                 | PINE 21 |
| 250 | CONTINUE                                   | PINE 22 |
|     | IF(IXA.LT.C.OR.IXA.GT.100) GO TO 260       | PINE 23 |
|     | IF(IYA.LT.C.OR.IYA.GT.50) GO TO 260        | PINE 24 |
|     | CALL PPLCT(IXA,IYA,NSYM,NCT)               | PINE 25 |
| 260 | CONTINUE                                   | PINE 26 |
|     | IXA=IXA+1                                  | PINE 27 |
|     | YA=(N*(AYB-AYA))/(AXB-AXA)                 | PINE 28 |
|     | IYA=AYA+YA+0.5                             | PINE 29 |
|     | N=N+1                                      | PINE 30 |
|     | IF(IXA.LE.IXB) GO TO 250                   | PINE 31 |
|     | GO TO 400                                  | PINE 32 |
| C   |                                            | PINE 33 |
| C   | SET PARAMETERS FOR Y DIRECTION             | PINE 34 |
| C   |                                            | PINE 35 |
| 290 | CONTINUE                                   | PINE 36 |
|     | IF(AYB.GT.AYA) GO TO 295                   | PINE 37 |
|     | AYB=Y1                                     | PINE 38 |
|     | AYA=Y2                                     | PINE 39 |
|     | AXB=X1                                     | PINE 40 |
|     | AXA=X2                                     | PINE 41 |
| 295 | CONTINUE                                   | PINE 42 |
|     | IXA=AXA+.5                                 | PINE 43 |
|     | IXB=AXB+.5                                 | PINE 44 |
|     | IYA=AYA+.5                                 | PINE 45 |
|     | IYB=AYB+.5                                 | PINE 46 |
| 300 | CONTINUE                                   | PINE 47 |
|     | IF(IXA.LT.C.OR.IXA.GT.100) GO TO 310       | PINE 48 |
|     | IF(IYA.LT.C.OR.IYA.GT.50) GO TO 310        | PINE 49 |
|     | CALL PPLCT(IXA,IYA,NSYM,NCT)               | PINE 50 |
| 310 | CONTINUE                                   | PINE 51 |
|     | IYA=IYA+1                                  | PINE 52 |
|     | XA=(N*(AXB-AXA))/(AYB-AYA)                 | PINE 53 |
|     | IXA=XA+AXA+0.5                             | PINE 54 |
|     | N=N+1                                      | PINE 55 |
|     | IF(IYA-IYB) 300,320,400                    | PINE 56 |
| 320 | IXA = IXB                                  | PINE 57 |
|     | GO TO 300                                  | PINE 58 |
| 400 | RETURN                                     | PINE 59 |
|     | END                                        | PINE 60 |

|                                                                |          |
|----------------------------------------------------------------|----------|
| SUBROUTINE PPLOT(IX,IY,K,NCT)                                  | PPLO 1   |
| DIMENSION A(51,101),SYM(9)                                     | PPLO 2   |
| COMMON /LAB/ TITLE(18),XLAB(11),YLAB(6)                        | PPLO 3   |
| 1,HORIZ(20),VERT(7,6),IT                                       | PPLO 4X  |
| DATA SYM / 4H****,4H++++, 4H''''', 4HXXXX, 4H...., 4H2222,     | PPLO 5   |
| 1 4H , 4HIIII, 4H--- /                                         | PPLO 6   |
| IF(K-99) 200,220,230                                           | PPLO 7   |
| 200 A(51-IY,IX+1)=SYM(K)                                       | PPLO 8   |
| RETURN                                                         | PPLO 9   |
| 220 CONTINUE                                                   | PPLO 10  |
| I=0                                                            | PPLO 11  |
| WRITE(6,103) TITLE,NCT                                         | PPLO 12  |
| DO 225 II=1,6                                                  | PPLO 13  |
| I=I+1                                                          | PPLO 14  |
| IF(YLAB(II).LE.100000.)WRITE(6,101)YLAB(II),(A(I,J),J=1,101)   | PPLO 15  |
| IF(YLAB(II).GT.100000.)WRITE(6,107)YLAB(II),(A(I,J),J=1,101)   | PPLO 16  |
| IF(II.EC.6) GO TO 228                                          | PPLO 17  |
| DO 224 JJ=1,9                                                  | PPLO 18  |
| I=I+1                                                          | PPLO 19  |
| IF(I.NE.28) GO TO 221                                          | PPLO 20  |
| WRITE(6,108) VERT(5,IT),VERT(6,IT),VERT(7,IT),(A(I,J),J=1,101) | PPLO 21X |
| GO TO 224                                                      | PPLO 22  |
| 221 IF(I.NE.24) GO TO 222                                      | PPLO 23  |
| WRITE(6,106) VERT(1,IT),VERT(2,IT),(A(I,J),J=1,101)            | PPLO 24X |
| GO TO 224                                                      | PPLO 25  |
| 222 IF(I.NE.26) GO TO 223                                      | PPLO 26  |
| WRITE(6,106) VERT(3,IT),VERT(4,IT),(A(I,J),J=1,101)            | PPLO 27X |
| GO TO 224                                                      | PPLO 28  |
| 223 WRITE(6,100) (A(I,J),J=1,101)                              | PPLO 29  |
| 224 CONTINUE                                                   | PPLO 30  |
| 225 CONTINUE                                                   | PPLO 31  |
| 228 CONTINUE                                                   | PPLO 32  |
| WRITE(6,102) XLAB                                              | PPLO 33  |
| WRITE(6,105) HORIZ                                             | PPLO 34  |
| 100 FORMAT(10X,101A1)                                          | PPLO 35  |
| 101 FORMAT(' ',F16.3,1X,101A1)                                 | PPLO 36  |
| 102 FORMAT(' ',F19.1,10F10.1)                                  | PPLO 37  |
| 103 FORMAT(11,20X,18A4,16/)                                    | PPLO 38  |
| 105 FORMAT(/30X,20A4)                                          | PPLO 39  |
| 106 FORMAT(3X,2A4,7X,101A1)                                    | PPLO 40  |
| 107 FORMAT(' ',1PE16.2,1X,101A1)                               | PPLO 41  |
| 108 FORMAT(3X,3A4,3X,101A1)                                    | PPLO 41A |
| 230 DO 250 I=1,50                                              | PPLO 42  |
| DO 240 J=1,101                                                 | PPLO 43  |
| 240 A(I,J)=SYM(7)                                              | PPLO 44  |
| A(I,1)=SYM(8)                                                  | PPLO 45  |
| 250 CONTINUE                                                   | PPLO 46  |
| DO 260 J=1,101                                                 | PPLO 47  |
| 260 A(51,J)=SYM(9)                                             | PPLO 48  |
| DO 270 I=1,101,10                                              | PPLO 49  |
| 270 A(51,I)=SYM(8)                                             | PPLO 50  |
| DO 290 I=11,41,10                                              | PPLO 51  |
| A(I,1)=SYM(9)                                                  | PPLO 52  |
| 290 CONTINUE                                                   | PPLO 53  |
| RETURN                                                         | PPLO 54  |
| END                                                            | PPLO 55  |

Section 2

RUNNOFF BLOCK

|                   | <u>Page</u> |
|-------------------|-------------|
| Subroutine RUNOFF | 15          |
| Subroutine GUTTER | 16          |
| Subroutine HCURVE | 20          |
| Subroutine HYDRO  | 22          |
| Subroutine RHYDRO | 25          |
| Subroutine WSHED  | 30          |
| Subroutine SFQUAL | 32          |

|   |                                                            |         |
|---|------------------------------------------------------------|---------|
|   | SUBROUTINE RUNOFF                                          | RUND 1  |
|   | COMMON /TAPES/ INCNT, IOUTCT, JIN(10), JOUT(10), NSCRAT(5) | RUND 2  |
| C | WRITE(6,102)                                               | RUND 3  |
|   | 102 FORMAT('1 ENTRY MADE TO RUNOFF MODEL')                 | RUND 4  |
|   | INCNT=INCNT+1                                              | RUND 5  |
|   | IOUTCT=ICUTCT+1                                            | RUND 6  |
|   | CALL HYDFC                                                 | RUND 7  |
|   | CALL GRAPH(C)                                              | RUND 8  |
|   | CALL SFQUAL                                                | RUND 9  |
|   | RETURN                                                     | RUND 10 |
|   | END                                                        | RUND 11 |
|   |                                                            | RUND 12 |



|                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                |                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                     |
|----------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|-----------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| <pre> SUBROUTINE GUTTER C C   THIS SUBROUTINE COMPUTES THE INSTANTANEOUS WATER DEPTH C   AND FLOW RATE FOR THE GUTTERS/PIPES C C***** SPECIFICATION STATEMENTS C COMMON /TAPES/ INCNT,IOUTCT,JIN(10),JOUT(10),NSCRAT(5) COMMON NW,NG,NIN,HISTOG,TRAIN,DELT,DELT2,NOW,NOG,NSTEP,TAREA, 1 TIME,TIME2,RI,RLOSS,SUMR,SUMI,SUMOFF,SUMST,TZERO,NING COMMON WFLOW(160),KWIDTH(160),WAREA(160),WSLOPE(160),WN(160), 1 WSTORE(160,3),WLMAX (160),WLMIN(160),DECAY(160),WDEPTH(160,3), 2 WCGN(160,3),NAMEW(160),FLIMP(160) COMMON GFLCH(160),GWIDTH(160),GLEN(160),GSLOPE(160),GS1(160), 1 GS2(160),GN(160),GDEPTH(160),GCUN(160),NPG(160),DFULL(160), 2 NGUT(160),SUMQW(160),PCTZER COMMON NGTOG(160,10),NGTOG(160,10),NWTOI(10),NGTOI(160) COMMON RAIN(160,10),NHJET(160),NRAIN,NPGAG,NHISTO,THISTO COMMON CSUR(160),DELD(160),QIN(160) COMMON IPRNT(160),ISAVE(160),NPRNT,NSAVE,OUTFLW(160),INTERV, 1 INCNT NOUT=JOUT(IOUTCT) NTQUAL=NSCRAT(1) C DO 400 N=1,NOG J=NGUT(N) C C***** INPLTS FROM ADJACENT WATERSHED AREAS C SUMQW(J)=C. DO 220 JK=1,NIN IF(NWTCG(J,JK).EQ.0) GO TO 240 NX=NWTCG(J,JK) 220 SUMQW(J)=SUMQW(J)+WFLOW(NX) C C***** INPUTS FROM UPSTREAM GUTTERS C 240 QIN(J)=SUMQW(J) DO 260 JK=1,NIN IF(NGTCG(J,JK).EQ.0) GO TO 280 NX=NGTCG(J,JK) 260 QIN(J)=QIN(J)+GFLCH(NX) C 280 DO=GDEPTH(J) IF(QIN(J).NE.0.) GO TO 290 IF(GDEPTH(J).EQ.C.) GO TO 391 290 IF(NPG(J).EQ.3) GO TO 391 IFLG=0 DELD(J)=0. DO 360 I=1,30 C C***** COMPUTE CHANGE IN DEPTH (NEWTON-RAPHSON) C C***** ESTIPATED FINAL DEPTH D1=GDEPTH(J)+DELD(J) IF(NPG(J).EQ.2) GO TO 295 C C   * * * * * C C**** TRAPEZDICAL GUTTER </pre> | <pre> GUTT 1 GUTT 2 GUTT 3 GUTT 4 GUTT 5 GUTT 6 GUTT 7 GUTT 8 GUTT 9 GUTT 10 GUTT 11 GUTT 12 GUTT 13 GUTT 14 GUTT 15 GUTT 16 GUTT 17 GUTT 18 GUTT 19 GUTT 20 GUTT 21 GUTT 22 GUTT 23 GUTT 24 GUTT 25 GUTT 26 GUTT 27 GUTT 28 GUTT 29 GUTT 30 GUTT 31 GUTT 32 GUTT 33 GUTT 34 GUTT 35 GUTT 36 GUTT 37 GUTT 38 GUTT 39 GUTT 40 GUTT 41 GUTT 42 GUTT 43 GUTT 44 GUTT 45 GUTT 46 GUTT 47 GUTT 48 GUTT 49 GUTT 50 GUTT 51 GUTT 52 GUTT 53 GUTT 54 GUTT 55 GUTT 56 GUTT 57 GUTT 58 GUTT 59 GUTT 60 </pre> |
|----------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|-----------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|

|        |                                                                   |          |
|--------|-------------------------------------------------------------------|----------|
| C      |                                                                   | GUTT 61  |
|        | IF(D1.LT.0.) D1=0.                                                | GUTT 62  |
| C      |                                                                   | GUTT 63  |
| C***** | VOLUME CHANGE (TRAPEZOIDAL SECTION)                               | GUTT 64  |
|        | DELV=GLEN(J)*DELD(J)*((GS1(J)+GS2(J))*(D0+0.5*DELD(J))+GWIDTH(J)) | GUTT 65  |
|        | DDELV=GLEN(J)*((GS1(J)+GS2(J))*D1+GWIDTH(J))                      | GUTT 66  |
| C      |                                                                   | GUTT 67  |
| C***** | CROSS-SECTIONAL AREA (TRAPEZOIDAL CROSS-SECTION)                  | GUTT 68  |
|        | AX0=0.5*(GS1(J)+GS2(J))*D0**2+GWIDTH(J)*D0                        | GUTT 69  |
|        | AX1=0.5*(GS1(J)+GS2(J))*D1**2+GWIDTH(J)*D1                        | GUTT 70  |
|        | DAX1=(GS1(J)+GS2(J))*D1+GWIDTH(J)                                 | GUTT 71  |
| C      |                                                                   | GUTT 72  |
| C***** | WETTED PERIMETER (TRAPEZOIDAL CROSS-SECTION)                      | GUTT 73  |
|        | WPO=SQRT(GS1(J)**2+1.)*D0+SQRT(GS2(J)**2+1.)*D0+GWIDTH(J)         | GUTT 74  |
|        | WP1=SQRT(GS1(J)**2+1.)*D1+SQRT(GS2(J)**2+1.)*D1+GWIDTH(J)         | GUTT 75  |
|        | DWP1=SQRT(GS1(J)**2+1.)*D1+SQRT(GS2(J)**2+1.)*D1+GWIDTH(J)        | GUTT 76  |
|        | GO TO 315                                                         | GUTT 77  |
| C      |                                                                   | GUTT 78  |
| C      | * * * * *                                                         | GUTT 79  |
| C      |                                                                   | GUTT 80  |
| C****  | CIRCULAR PIPE                                                     | GUTT 81  |
| C      |                                                                   | GUTT 82  |
|        | 295 IF(1.GT.1) GO TO 307                                          | GUTT 83  |
|        | D1=1.5707963                                                      | GUTT 84  |
|        | DELD(J)=D1-GDEPTH(J)                                              | GUTT 85  |
|        | 307 IF(D1.GT.0.) GO TO 308                                        | GUTT 86  |
|        | D1=0.                                                             | GUTT 87  |
|        | DELD(J)=-GDEPTH(J)                                                | GUTT 88  |
|        | 308 IF(D1.LE.DFULL(J)) GO TO 310                                  | GUTT 89  |
|        | D1=DFULL(J)                                                       | GUTT 90  |
|        | DELD(J)=D1-GDEPTH(J)                                              | GUTT 91  |
| C      |                                                                   | GUTT 92  |
| C***** | VOLUME CHANGE (PIPE)                                              | GUTT 93  |
|        | 310 DELV=GLEN(J)*(GWIDTH(J)**2/4.)*(DELD(J)-0.5*SIN(2.*D1))       | GUTT 94  |
|        | 1 +0.5*SIN(2.*D0))                                                | GUTT 95  |
|        | DDELV=GLEN(J)*(GWIDTH(J)**2/4.)*(1.-COS(2.*D1))                   | GUTT 96  |
| C      |                                                                   | GUTT 97  |
| C***** | CROSS-SECTIONAL AREA (PIPE)                                       | GUTT 98  |
|        | AX0=(GWIDTH(J)**2/4.)*(1.-0.5*SIN(2.*D0))                         | GUTT 99  |
|        | AX1=(GWIDTH(J)**2/4.)*(1.-0.5*SIN(2.*D1))                         | GUTT 100 |
|        | DAX1=(GWIDTH(J)**2/4.)*(1.-COS(2.*D1))                            | GUTT 101 |
| C      |                                                                   | GUTT 102 |
| C***** | WETTED PERIMETER (PIPE)                                           | GUTT 103 |
|        | WPO=GWIDTH(J)*D0                                                  | GUTT 104 |
|        | WP1=GWIDTH(J)*D1                                                  | GUTT 105 |
|        | DWP1=GWIDTH(J)*D1                                                 | GUTT 106 |
| C      |                                                                   | GUTT 107 |
| C      | * * * * *                                                         | GUTT 108 |
| C      |                                                                   | GUTT 109 |
| C***** | HYDRAULIC RADIUS (ALL CROSS-SECTIONS)                             | GUTT 110 |
|        | 315 IF(AX0.LT.0.) AX0=0.                                          | GUTT 111 |
|        | IF(AX1.LT.0.) AX1=0.                                              | GUTT 112 |
|        | IF(WPO.LE.0.) WPO=0.001                                           | GUTT 113 |
|        | IF(WP1.LE.0.) WP1=0.001                                           | GUTT 114 |
|        | RAD0=AX0/WPO                                                      | GUTT 115 |
|        | RAD1=AX1/WP1                                                      | GUTT 116 |
| C      |                                                                   | GUTT 117 |
| C***** | FLCW                                                              | GUTT 118 |
|        | FLOW0=GCCN(J)*(AX0**1.6666667)/(WPO**0.6666667)                   | GUTT 119 |
|        | FLOW1=GCCN(J)*(AX1**1.6666667)/(WP1**0.6666667)                   | GUTT 120 |

|                                                          |         |
|----------------------------------------------------------|---------|
| FLOW=0.5*(FLOW0+FLOW1)                                   | GUTT121 |
| DFLOW1=C.5*GCON(J)*(1.6666667*(RAD1**0.66666667)*DAX1    | GUTT122 |
| 1 -0.66666667*(RAD1**1.6666667)*DWPI)                    | GUTT123 |
| C                                                        | GUTT124 |
| C***** NEWTON-RAPHSON CORRECTION (ALL CROSS-SECTIONS)    | GUTT125 |
| F=DELV+DELT*(FLOW-QIN(J))-QSUR(J)                        | GUTT126 |
| DF=DDELV+DELT*DFLOW1                                     | GUTT127 |
| C                                                        | GUTT128 |
| IF(DF.GT.0.) GO TO 320                                   | GUTT129 |
| C***** ZERO SLOPE                                        | GUTT130 |
| DEL=0.01                                                 | GUTT131 |
| GO TO 340                                                | GUTT132 |
| C***** NON-ZERO SLOPE                                    | GUTT133 |
| 320 DEL=DELC(J)-F/DF                                     | GUTT134 |
| C***** CONVERGENCE CHECK (INDIVIDUAL GUTTER)             | GUTT135 |
| 340 IF(1.EQ.1) GO TO 360                                 | GUTT136 |
| IF(GDEPTH(J)+DEL.LT.DFULL(J)) GO TO 355                  | GUTT137 |
| IF(IFLG.EQ.1) GO TO 390                                  | GUTT138 |
| DEL=DFULL(J)-GDEPTH(J)                                   | GUTT139 |
| IFLG=1                                                   | GUTT140 |
| GO TO 360                                                | GUTT141 |
| 355 IFLG=0                                               | GUTT142 |
| IF(A3S(F).LT.0.1) GO TO 380                              | GUTT143 |
| 360 DELD(J)=DEL                                          | GUTT144 |
| WRITE(6,1000) TIME,J,GDEPTH(J),DELD(J)                   | GUTT145 |
| 1000 FORMAT(' CHECK RESULTS. NOT CONVERGED IN *GUTTER*', | GUTT146 |
| 1 F8.0,16,2E12.5)                                        | GUTT147 |
| C                                                        | GUTT148 |
| C***** NEW DEPTH AT END OF TIME INTERVAL                 | GUTT149 |
| C                                                        | GUTT150 |
| 380 DELD(J)=DEL                                          | GUTT151 |
| GDEPTH(J)=GDEPTH(J)+DELD(J)                              | GUTT152 |
| QSUR(J)=C.                                               | GUTT153 |
| C                                                        | GUTT154 |
| C***** AVERAGE FLOW DURING TIME INTERVAL                 | GUTT155 |
| C                                                        | GUTT156 |
| GFLOW(J)=FLOW                                            | GUTT157 |
| GO TO 400                                                | GUTT158 |
| C                                                        | GUTT159 |
| C***** SURCHARGE                                         | GUTT160 |
| C                                                        | GUTT161 |
| 390 GDEPTH(J)=DFULL(J)                                   | GUTT162 |
| GFLOW(J)=FLOW1                                           | GUTT163 |
| QSUR(J)=QSUR(J)+(QIN(J)-FLOW)*DELT                       | GUTT164 |
| GO TO 400                                                | GUTT165 |
| 391 GFLOW(J)=QIN(J)                                      | GUTT166 |
| 400 CONTINUE                                             | GUTT167 |
| C                                                        | GUTT168 |
| C                                                        | GUTT169 |
| C                                                        | GUTT170 |
| IF(NPRNT.LT.1) GO TO 510                                 | GUTT171 |
| INTCNT=INTCNT+1                                          | GUTT172 |
| IF(INTCNT.LT.INTERV) GO TO 510                           | GUTT173 |
| INTCNT=0                                                 | GUTT174 |
| DO 500 N=1,NPRNT                                         | GUTT175 |
| J=JPRNT(N)                                               |         |
| 500 OUTFLW(N)=QIN(J)                                     |         |
| NTIMEH=TIME/3600.                                        |         |
| TIMEN=TIME/60.-FLOAT(NTIMEH)*60.                         |         |
| WRITE(6,9010) NTIMEH,TIMEN,(OUTFLW(N),N=1,NPRNT)         |         |

|      |                                                                  |         |
|------|------------------------------------------------------------------|---------|
| 9010 | FORMAT(14,F6.2,10F10.2,/(10X,10F10.2))                           | GUTT181 |
|      | DO 505 N=1,NOG                                                   | GUTT182 |
|      | J=NGUT(N)                                                        | GUTT183 |
|      | IF(QSUR(J).GT.0) WRITE(6,9000) J,QSUR(J),GFLOW(J)                | GUTT184 |
| 9000 | FORMAT(' GUTTER',14,' SURCHARGED, SURCHARGE='F10.0,' CUFT,FLOW=' | GUTT185 |
|      | 1F10.1,' CFS')                                                   | GUTT186 |
|      | 505 CONTINUE                                                     | GUTT187 |
| C    |                                                                  | GUTT188 |
| C    |                                                                  | GUTT189 |
| C    | WRITE INLETS TO BE SAVED                                         | GUTT190 |
|      |                                                                  | GUTT191 |
| 510  | IF(NSAVE.LT.1) GO TO 610                                         | GUTT192 |
|      | DO 600 N=1,NSAVE                                                 | GUTT193 |
|      | J=ISAVE(N)                                                       | GUTT194 |
| 600  | OUTFLW(N)=QIN(J)                                                 | GUTT195 |
|      | WRITE(NCUT) TIME,(OUTFLW(N),N=1,NSAVE)                           | GUTT196 |
|      | WRITE(NTCLAL)(OUTFLW(N),N=1,NSAVE)                               | GUTT197 |
| 610  | CONTINUE                                                         | GUTT198 |
| C    |                                                                  | GUTT199 |
|      | RETURN                                                           | GUTT200 |
|      | END                                                              |         |

|                                                                 |      |     |
|-----------------------------------------------------------------|------|-----|
| SUBROUTINE HCURVE(NTYPE, INLET)                                 | HCUR | 1   |
| COMMON NW,NG,NIA,HISTCG,TRAIN,DELT,DELT2,NOW,NUG,NSTEP,TAREA,   | HCUR | 2   |
| 1 TIME,TIME2,R1,RLOSS,SUMR,SUM1,SUMOFF,SUMST,TZERO,NING         | HCUR | 3   |
| COMMON WFLOW(160),WWIDTH(160),WAREA(160),WSLOPE(160),WN(160),   | HCUR | 4   |
| 1 WSTORE(160,3),WLMAX(160),WLMIN(160),DECAY(160),WDEPTH(160,3), | HCUR | 5   |
| 2 WCON(160,3),NAMEW(160),PCIMP(160)                             | HCUR | 6   |
| COMMON GFLCW(160),GWIDTH(160),GLEN(160),GSLOPE(160),GS1(160),   | HCUR | 7   |
| 1 GS2(160),GN(160),GDEPTH(160),GCON(160),NPG(160),DFULL(160),   | HCUR | 8   |
| 2 NGUT(160),SUMQW(160),PCTZER                                   | HCUR | 9   |
| COMMON NWTG(160,10),NGTG(160,10),NWTOI(10),NGTOI(160)           | HCUR | 10  |
| COMMON FAIR(160,10),NFYET(160),NRAIN,NRGAG,NHISTO,THISTO        | HCUR | 11  |
| COMMON QSUR(160),DELD(160),QIN(160)                             | HCUR | 12  |
| COMMON IPRAT(160),ISAVE(160),NPRNT,NSAVE,OUTFLW(160),INTERV,    | HCUR | 13  |
| 1 INTCNT                                                        | HCUR | 14  |
| COMMON CLPMY(40)                                                | HCUR | 15  |
| COMMON HGRAPH(160),HTIME(160)                                   | HCUR | 16  |
| COMMON X(201,5),Y(201,5),NPT(5)                                 | HCUR | 17  |
| COMMON/LAB/ TITLE(18),XLAB(11),YLAB(6)                          | HCUR | 18  |
| 1,HORIZ(20),VERT(7,6),IT                                        | HCUR | 19X |
| DIMENSION VER(7,2),TITL(7,2)                                    | HCUR | 20X |
| DIMENSION TITEL(18),ORIZ(20)                                    | HCUR | 21  |
| DATA TITL /4HRAIN,4HFALL,4H HYE,4HTUGR,4HAPH ,4H ,4H ,          | HCUR | 22X |
| * 4HINLE,4HT HY,4HDROG,4HRAPII,4H ,4H ,4H /                     | HCUR | 23X |
| DATA TITEL / 16*4H ,4HBASI,4HNN NO /                            | HCUR | 24  |
| DATA ORIZ /8*4H ,4HTIME,4H IN ,4HHOUR,4HS ,8*4H /               | HCUR | 25  |
| DATA VER / 4HRAIN,4HFALL,4H I,4HNN ,4HIN /,4H HR,4H             | HCUR | 26X |
| 1, 4H RUN,4HOFF ,4H I,4HNN ,4H CF,4HS ,4H /                     | HCUR | 27X |
| DO 750 I=1,18                                                   | HCUR | 28  |
| 750 TITLE(I)=TITEL(I)                                           | HCUR | 29  |
| DO 760 I=1,20                                                   | HCUR | 30  |
| 760 HORIZ(I)=ORIZ(I)                                            | HCUR | 31  |
| C RAINFALL HYETOGRAPH OR INLET HYDROGRAPH                       | HCUR | 32  |
| IT=1                                                            | HCUR | 32A |
| DO 5 I=1,7                                                      | HCUR | 33X |
| J = I+2                                                         | HCUR | 34  |
| VERT(I,IT)=VER(I,NTYPE)                                         | HCUR | 35X |
| TITLE(J) = TITL(I,NTYPE)                                        | HCUR | 36  |
| 5 CONTINUE                                                      | HCUR | 37  |
| IF(NTYPE.GT.1) GO TO 380                                        | HCUR | 38  |
| NGAGP=5                                                         | HCUR | 39  |
| TMAX=FLCAT(NSTEP)*DELT+TZERO                                    | HCUR | 40  |
| I=0                                                             | HCUR | 41  |
| DO 350 J=1,NRGAG,NGAGP                                          | HCUR | 42  |
| DO 300 K=1,NGAGP                                                | HCUR | 43  |
| I=I+1                                                           | HCUR | 44  |
| IF(I.GT.NRGAG) GO TO 320                                        | HCUR | 45  |
| TIME=TZERO                                                      | HCUR | 46  |
| N=0                                                             | HCUR | 47  |
| DO 250 L=1,NHISTO                                               | HCUR | 48  |
| N=N+1                                                           | HCUR | 49  |
| X(N,K)=TIME/3600.                                               | HCUR | 50  |
| Y(N,K)=RAIN(L,I)*43200.                                         | HCUR | 51  |
| TIME=TIME+ HISTOG                                               | HCUR | 52  |
| IF(TIME.GT.TMAX) GO TO 300                                      | HCUR | 53  |
| IF(NHISTO.GT.50) GO TO 250                                      | HCUR | 54  |
| N=N+1                                                           | HCUR | 55  |
| X(N,K)=TIME/3600.                                               | HCUR | 56  |
| Y(N,K)=RAIN(L,I)*43200.                                         | HCUR | 57  |
| 250 CONTINUE                                                    | HCUR | 58  |
| N=N+1                                                           | HCUR | 59  |

|                                                                      |         |
|----------------------------------------------------------------------|---------|
| X(N,K)=TMAX/3600.                                                    | HCUR 60 |
| Y(N,K)=C.                                                            | HCUR 61 |
| 300 NPT(K)=N                                                         | HCUR 62 |
| K=NGAGP+1                                                            | HCUR 63 |
| 320 K=K-1                                                            | HCUR 64 |
| CALL CURVE(X,Y,NPT,K, INLET)                                         | HCUR 65 |
| JT=J+K-1                                                             | HCUR 66 |
| WRITE(6,9000) (K,K=J,JT)                                             | HCUR 67 |
| 9000 FORMAT(/30X,'RAINGAGE LEGEND',I8,4H = *,I8,4H = +,I8,4H = ',I8, | HCUR 68 |
| 14H = X,I8,4H = .)                                                   | HCUR 69 |
| 350 CONTINUE                                                         | HCUR 70 |
| RETURN                                                               | HCUR 71 |
| 380 X(1,1)=HTIME(1)                                                  | HCUR 72 |
| Y(1,1)=HGRAPH(1)                                                     | HCUR 73 |
| C CHOOSE THE SCALE DOWN FACTOR                                       | HCUR 74 |
| M=(NSTEP+99)/100                                                     | HCUR 75 |
| I = 1                                                                | HCUR 76 |
| DO 10 J = 1,NSTEP,M                                                  | HCUR 77 |
| I = I + 1                                                            | HCUR 78 |
| X(I,1)=HTIME(I)                                                      | HCUR 79 |
| Y(I,1)=HGRAPH(I)                                                     | HCUR 80 |
| 10 CONTINUE                                                          | HCUR 81 |
| 48 NPT(1)=1                                                          | HCUR 82 |
| CALL CURVE(X,Y,NPT,1, INLET)                                         | HCUR 83 |
| RETURN                                                               | HCUR 84 |
| END                                                                  | HCUR 85 |

|                  |                                                                  |      |    |
|------------------|------------------------------------------------------------------|------|----|
| SUBROUTINE HYDRC |                                                                  | HYDR | 1  |
| C                |                                                                  | HYDR | 2  |
| C                | ***** SPECIFICATION STATEMENTS                                   | HYDR | 3  |
| C                |                                                                  | HYDR | 4  |
|                  | COMMON NW,NG,NIN,HISTCG,TRAIN,DELT,DELT2,NOW,NOG,NSTEP,TAREA,    | HYDR | 5  |
|                  | 1 TIME,TIME2,RI,RLOSS,SUMR,SUMI,SUMOFF,SUMST,TZERO,NING          | HYDR | 6  |
|                  | COMMON WFLOW(160),WWIDTH(160),WAREA(160),WSLOPE(160),WN(160),    | HYDR | 7  |
|                  | 1 WSTORE(160,3),WLMAX (160),WLMIN(160),DECAY(160),WDEPTH(160,3), | HYDR | 8  |
|                  | 2 WCON(160,3),NAMEW(160),PCIMP(160)                              | HYDR | 9  |
|                  | COMMON GFLOW(160),GWIDTH(160),GLEN(160),GSLOPE(160),GS1(160),    | HYDR | 10 |
|                  | 1 GS2(160),GN(160),GDEPTH(160),GCON(160),NPG(160),DFULL(160),    | HYDR | 11 |
|                  | 2 NGUT(160),SUMQW(160),PCTZER                                    | HYDR | 12 |
|                  | COMMON NWTQG(160,10),NGTQS(160,10),NWTQI(160),NGTQI(160)         | HYDR | 13 |
|                  | COMMON RAIN(160,10),NHET(160),NRAIN,NRGAG,NHISTO,THISTO          | HYDR | 14 |
|                  | COMMON QSUR(160),DELD(160),QIN(160)                              | HYDR | 15 |
|                  | COMMON IFRAT(160),ISAVE(160),NPRNT,NSAVE,OUTFLW(160),INTERV,     | HYDR | 16 |
|                  | 1 INTCNT                                                         | HYDR | 17 |
|                  | COMMON TITLE(40)                                                 | HYDR | 18 |
|                  | COMMON HGRAPH(160),HTIME(160)                                    | HYDR | 19 |
| C                |                                                                  | HYDR | 20 |
| C                | ***** INITIALIZATION                                             | HYDR | 21 |
| C                |                                                                  | HYDR | 22 |
|                  | NW=160                                                           | HYDR | 23 |
|                  | NG=160                                                           | HYDR | 24 |
|                  | NING=160                                                         | HYDR | 25 |
|                  | NRANVL=200                                                       | HYDR | 26 |
|                  | NIN=10                                                           | HYDR | 27 |
|                  | INTCNT=0                                                         | HYDR | 28 |
|                  | DO 220 I=1,NW                                                    | HYDR | 29 |
|                  | WFLOW(I)=0.0                                                     | HYDR | 30 |
|                  | WWIDTH(I)=0.                                                     | HYDR | 31 |
|                  | WDEPTH(I,1)=0.                                                   | HYDR | 32 |
|                  | WDEPTH(I,3)=0.                                                   | HYDR | 33 |
| 220              | WDEPTH(I,2)=0.                                                   | HYDR | 34 |
|                  | DO 240 I=1,NG                                                    | HYDR | 35 |
|                  | NPG(I)=0                                                         | HYDR | 36 |
|                  | NGUT(I)=0.                                                       | HYDR | 37 |
|                  | QSUR(I)=0.0                                                      | HYDR | 38 |
|                  | DELD(I)=0.0                                                      | HYDR | 39 |
|                  | QIN(I)=0.0                                                       | HYDR | 40 |
|                  | GFLOW(I)=0.0                                                     | HYDR | 41 |
|                  | GDEPTH(I)=0.                                                     | HYDR | 42 |
| 240              | GLEN(I)=0.                                                       | HYDR | 43 |
|                  | DO 250 J=1,NING                                                  | HYDR | 44 |
| 250              | NGTOI(J)=0                                                       | HYDR | 45 |
|                  | DO 260 J=1,NIN                                                   | HYDR | 46 |
|                  | NWTQI(J)=0                                                       | HYDR | 47 |
|                  | DO 260 I=1,NG                                                    | HYDR | 48 |
|                  | NWTQG(I,J)=0                                                     | HYDR | 49 |
| 260              | NGTQI(I,J)=0                                                     | HYDR | 50 |
|                  | DO 280 I=1,NRANVL                                                | HYDR | 51 |
|                  | HGRAPH(I)=0.                                                     | HYDR | 52 |
|                  | HTIME(I)=0.                                                      | HYDR | 53 |
|                  | DO 280 J=1,10                                                    | HYDR | 54 |
| 280              | RAIN(I,J)=0.                                                     | HYDR | 55 |
|                  | SUMR = 0.0                                                       | HYDR | 56 |
|                  | SUMI = 0.0                                                       | HYDR | 57 |
|                  | SUMOFF = 0.0                                                     | HYDR | 58 |
|                  | SUMST = 0.0                                                      | HYDR | 59 |
| C                |                                                                  | HYDR | 60 |

|                                               |          |
|-----------------------------------------------|----------|
| C***** CALL INPUT SUBROUTINE                  | HYDR 61  |
| C                                             | HYDR 62  |
| CALL RHYCRO(INLET)                            | HYDR 63  |
| TIME=TZERO                                    | HYDR 64  |
| C                                             | HYDR 65  |
| C***** SET UP ORDERING ARRAY                  | HYDR 66  |
| DO 2200 I=1,NIN                               | HYDR 67  |
| IF(NGTCI(I).EQ.0) GO TO 2220                  | HYDR 68  |
| NSPOT=NG+1-I                                  | HYDR 69  |
| 2200 NGUT(NSPCT)=NGTOI(I)                     | HYDR 70  |
| C***** BUILD TREE STRUCTURE                   | HYDR 71  |
| 2220 DO 2260 I=1,NG                           | HYDR 72  |
| KSPOT=NG+1-I                                  | HYDR 73  |
| ISUB=NGLT(KSPOT)                              | HYDR 74  |
| IF(ISUB.LE.0) GO TO 2260                      | HYDR 75  |
| DO 2240 J=1,NIN                               | HYDR 76  |
| IF(NGTCG(ISUB,J).EQ.0) GO TO 2260             | HYDR 77  |
| NSPOT=NSPOT-1                                 | HYDR 78  |
| 2240 NGUT(NSPOT)=NGTOG(ISUB,J)                | HYDR 79  |
| 2260 CONTINUE                                 | HYDR 80  |
| C***** SHIFT TO START OF ARRAY                | HYDR 81  |
| NSPOT=0                                       | HYDR 82  |
| DO 2280 I=1,NG                                | HYDR 83  |
| IF(NGLT(I).EQ.0) GO TO 2280                   | HYDR 84  |
| NSPOT=NSPOT+1                                 | HYDR 85  |
| NGUT(NSPCT)=NGUT(I)                           | HYDR 86  |
| 2280 CONTINUE                                 | HYDR 87  |
| C                                             | HYDR 88  |
| C***** CALCULATE INLET HYDROGRAPH             | HYDR 89  |
| C                                             | HYDR 90  |
| M=(NSTEP+99)/100                              | HYDR 91  |
| I=1                                           | HYDR 92  |
| HTIME(I)=TZERO/3600.                          | HYDR 93  |
| DO 440 II=1,NSTEP,M                           | HYDR 94  |
| I=I+1                                         | HYDR 95  |
| DO 430 IJ=1,M                                 | HYDR 96  |
| TIME=TIME+DELT                                | HYDR 97  |
| TIME2=TIME-DELT2                              | HYDR 98  |
| HTIME(I)=TIME/3600.                           | HYDR 99  |
| C                                             | HYDR 100 |
| C***** WATERSHED ELEMENTS (OVERLAND FLOW)     | HYDR 101 |
| C                                             | HYDR 102 |
| CALL WSHED                                    | HYDR 103 |
| C                                             | HYDR 104 |
| C***** GUTTER ELEMENTS                        | HYDR 105 |
| C                                             | HYDR 106 |
| IF(NUG.EQ.0) GO TO 340                        | HYDR 107 |
| CALL GUTTER                                   | HYDR 108 |
| 340 CONTINUE                                  | HYDR 109 |
| C                                             | HYDR 110 |
| C***** HYDROGRAPH CONSTRUCTION                | HYDR 111 |
| C                                             | HYDR 112 |
| C***** WATERSHEDS CONNECTED DIRECTLY TO INLET | HYDR 113 |
| HGRAPH(I)=0.                                  | HYDR 114 |
| DO 360 JK=1,NIN                               | HYDR 115 |
| IF(NWTOI(JK).EQ.0) GO TO 380                  | HYDR 116 |
| NX=NWTCI(JK)                                  | HYDR 117 |
| 360 HGRAPH(I)=HGRAPH(I)+WFLOW(NX)             | HYDR 118 |
| C                                             | HYDR 119 |
| C***** GUTTERS CONNECTED TO INLET             | HYDR 120 |



|                                                                |         |
|----------------------------------------------------------------|---------|
| 380 IF(NOG.EQ.0) GO TO 420                                     | HYDR121 |
| DO 400 JK=1,NING                                               | HYDR122 |
| IF(NGTOI(JK).EQ.0) GO TO 420                                   | HYDR123 |
| NX=NGICI(JK)                                                   | HYDR124 |
| 400 HGRAPH(I)=HGRAPH(I)+GFLOW(NX)                              | HYDR125 |
| 420 CONTINUE                                                   | HYDR126 |
| C***** SUM FOR CONTINUITY CHECK                                | HYDR127 |
| SUMOFF=SUMOFF+HGRAPH(I)*DELT                                   | HYDR128 |
| C                                                              | HYDR129 |
| 430 CONTINUE                                                   | HYDR130 |
| 440 CONTINUE                                                   | HYDR131 |
| C                                                              | HYDR132 |
| C***** CONTINUITY CHECK                                        | HYDR133 |
| DO 460 N=1,NOW                                                 | HYDR134 |
| J=NAMEW(N)                                                     | HYDR135 |
| SUMST=SUMST+WDEPTH(J,1)*WAREA(J)*PCIMP(J)/10000.*(100.-PCTZER) | HYDR136 |
| 1+WDEPTH(J,2)*(100.-PCIMP(J))/100.                             | HYDR137 |
| 460 CONTINUE                                                   | HYDR138 |
| ERROR=(SUMR-SUMI-SUMOFF-SUMST)*100./SUMR                       | HYDR139 |
| WRITE(6,9000) SUMR,SUMI,SUMOFF,SUMST,ERROR                     | HYDR140 |
| 9000 FORMAT('1TOTAL RAINFALL (CU FT) ',F10.0//                 | HYDR141 |
| 1 ' TOTAL INFILTRATION (CU FT) ',F10.0//                       | HYDR142 |
| 2 ' TCTAL GUTTER FLOW AT INLET (CU FT) ',F10.0//               | HYDR143 |
| 3 ' TOTAL SURFACE STORAGE AT END OF STORM (CU FT)',F10.0//     | HYDR144 |
| 4 ' ERROR IN CONTINUITY, PERCENTAGE OF RAINFALL, ',F10.5)      | HYDR145 |
| C                                                              | HYDR146 |
| C***** OUTPLT                                                  | HYDR147 |
| C                                                              | HYDR148 |
| CALL HCURVE(1,INLET)                                           | HYDR149 |
| CALL HCURVE(2,INLET)                                           | HYDR150 |
| RETURN                                                         | HYDR151 |
| END                                                            | HYDR152 |

|                                                                          |         |
|--------------------------------------------------------------------------|---------|
| SUBROUTINE RHYDRO(INLET)                                                 | RHYD 1  |
| COMMON /TAPES/ INCNT, IOUTCT, JIN(10), JOUT(10), NSCRAT(5)               | RHYD 2  |
| COMMON NW, NG, NIN, HISTOG, TRAIN, DELT, DELT2, NOW, NUG, NSTEP, TAREA,  | RHYD 3  |
| 1 TIME, TIME2, RI, RLOSS, SUMR, SUMI, SUMOFF, SUMST, TZERO, NING         | RHYD 4  |
| COMMON WFLOW(160), WWIDTH(160), WAREA(160), WSLOPE(160), WN(160),        | RHYD 5  |
| 1 WSTORE(160,3), WLMAX (160), WLMIN(160), DECAY(160), WDEPTH(160,3),     | RHYD 6  |
| 2 WCON(160,3), NAMEW(160), PCIMP(160)                                    | RHYD 7  |
| COMMON GFLOW(160), GWIDTH(160), GLEN(160), GSLOPE(160), GS1(160),        | RHYD 8  |
| 1 GS2(160), GN(160), GDEPTH(160), GCON(160), NPG(160), DFULL(160),       | RHYD 9  |
| 2 NGUT(160), SUMQW(160), PCTZER                                          | RHYD 10 |
| COMMON NUTLG(160,10), NGTUG(160,10), NWTUI(10), NGTOI(160)               | RHYD 11 |
| COMMON RAIN(160,10), NHYET(160), NRAIN, NRGAG, NHISTO, THISTO            | RHYD 12 |
| COMMON QSOR(160), CELD(160), QIN(160)                                    | RHYD 13 |
| COMMON IPRNT(160), ISAVE(160), NPRNT, NSAVE, OUTFLW(160), INTERV,        | RHYD 14 |
| 1 INCNT                                                                  | RHYD 15 |
| 2, TITLE(40)                                                             | RHYD 16 |
| C                                                                        | RHYD 17 |
| C**** GENERAL INFORMATION                                                | RHYD 18 |
| C                                                                        | RHYD 19 |
| READ(5,1005) TITLE                                                       | RHYD 20 |
| 1005 FORMAT(20A4)                                                        | RHYD 21 |
| READ(5,1000) INLET, NSTEP, NHR, NMN, DELT, NRGAG                         | RHYD 22 |
| 1, PCTZER                                                                | RHYD 23 |
| 1000 FORMAT(2I5, I3, I2, F5.1, I5, F5.0)                                 | RHYD 24 |
| IF(PCTZER.EQ.0.) PCTZER=25.                                              | RHYD 25 |
| TZERO=26CC.*FLOAT(NHR)+60.*FLOAT(NMN)                                    | RHYD 26 |
| WRITE(6,1010) TITLE, INLET, NSTEP                                        | RHYD 27 |
| 1010 FORMAT(1H ,20A4/, 1H ,2CA4//                                        | RHYD 28 |
| 1, 'OINLET NUMBER', I5/'CNUMBER OF TIME STEPS', I5)                      | RHYD 29 |
| WRITE(6,1041) DELT                                                       | RHYD 30 |
| WRITE(6,1043) PCTZER                                                     | RHYD 31 |
| 1043 FORMAT(1F0,F4.1, ' PERCENT OF IMPERVIOUS AREA HAS ZERO DETENTION DE | RHYD 32 |
| 1PTH')                                                                   | RHYD 33 |
| C                                                                        | RHYD 34 |
| C**** RAINFALL INTENSITY HISTOGRAM                                       | RHYD 35 |
| C                                                                        | RHYD 36 |
| READ(5,1020) NHISTO, THISTO                                              | RHYD 37 |
| 1020 FORMAT(15,F5.0)                                                     | RHYD 38 |
| WRITE(6,1040) NHISTO, THISTO                                             | RHYD 39 |
| 1040 FORMAT( 'OFOR', I6, ' RAINFALL STEPS, THE TIME INTERVAL IS'         | RHYD 40 |
| 1, F7.2, ' MINUTES')                                                     | RHYD 41 |
| TRAIN=FLCAT(NHISTO)*THISTO+TZERO/60.                                     | RHYD 42 |
| DO 230 N=1, NRGAG                                                        | RHYD 43 |
| READ(5,1030) (RAIN(I,N), I=1, NHISTO)                                    | RHYD 44 |
| 1030 FORMAT(1CF5.0)                                                      | RHYD 45 |
| C                                                                        | RHYD 46 |
| C***** PRINT RAINFALL HISTORY                                            | RHYD 47 |
| WRITE(6,1042) N, (RAIN(I,N), I=1, NHISTO)                                | RHYD 48 |
| 1042 FORMAT( 'OFOR RAINGAGE NUMBER', I4, ' RAINFALL HISTORY IS'          | RHYD 49 |
| 1//((10F10.2))                                                           | RHYD 50 |
| 1041 FORMAT('CINTEGRATION TIME INTERVAL (MINUTES)', F8.2)                | RHYD 51 |
| DO 220 I=1, NHISTO                                                       | RHYD 52 |
| 220 RAIN(I,N)=RAIN(I,N)/43200.                                           | RHYD 53 |
| 230 CONTINUE                                                             | RHYD 54 |
| TRAIN = TRAIN*60.                                                        | RHYD 55 |
| HISTOG = THISTO*60.                                                      | RHYD 56 |
| DELT = DELT*60.                                                          | RHYD 57 |
| DELT2 = DELT/2.                                                          | RHYD 58 |
| C                                                                        | RHYD 59 |
| C**** WATERSHED DATA                                                     | RHYD 60 |

|        |                                                                       |         |
|--------|-----------------------------------------------------------------------|---------|
| C      | NOW=0                                                                 | RHYD 61 |
|        | NSTOP=0                                                               | RHYD 62 |
|        | TAREA=C.                                                              | RHYD 63 |
|        | WRITE(6,1050)                                                         | RHYD 64 |
|        | 1050 FORMAT('1SUBAREA GUTTER WIDTH AREA PERCENT SLOPE RERHYD 66       |         |
|        | 1SISTANCE FACTOR SURFACE STORAGE(IN) INFILTRATION RATE(IN/HR) RHYD 67 |         |
|        | 1GAGE'/                                                               | RHYD 68 |
|        | 2' NUMBER OR MANHOLE (FT) (AC) IMPERV. (FT/FT) IMPERV. RHYD 69        |         |
|        | 3 PERV. IMPERV. PERV. MAXIMUM MINIMUM DECAY RATE ND') RHYD 70         |         |
|        | DO 340 I=1,NW                                                         | RHYD 71 |
|        | READ(5,1060) JK,N,NGOTO,W1,W2,W3,W4,W5,W6,W7,W8                       | RHYD 72 |
|        | 1,W9,W10,W11                                                          | RHYD 73 |
|        | 1060 FORMAT(3I5,3F5.0,7F5.3,F10.5)                                    | RHYD 74 |
|        | IF(W4.EQ.0.) W4=0.03                                                  | RHYD 75 |
|        | IF(W5.EQ.0.) W5=0.013                                                 | RHYD 76 |
|        | IF(W6.EQ.0.) W6=0.25                                                  | RHYD 77 |
|        | IF(W7.EQ.0.) W7=0.062                                                 | RHYD 78 |
|        | IF(W8.EQ.0.) W8=0.184                                                 | RHYD 79 |
|        | IF(W9.EQ.0.) W9=3.0                                                   | RHYD 80 |
|        | IF(W10.EQ.0.) W10=0.52                                                | RHYD 81 |
|        | IF(W11.EQ.0.) W11=0.00115                                             | RHYD 82 |
|        | IF(JK.EQ.0) JK=1                                                      | RHYD 83 |
|        | IF(N.EQ.0) GO TO 360                                                  | RHYD 84 |
|        | NOW=NOW+1                                                             | RHYD 85 |
|        | NAMEW(NOW)=N                                                          | RHYD 86 |
| C      |                                                                       | RHYD 87 |
| C***** | PRINT WATERSHED DATA                                                  | RHYD 88 |
|        | WRITE(6,1070) N,NGOTO,W1,W2,W3,W4,W5,W6,W7,W8                         | RHYD 89 |
|        | 1,W9,W10,W11,JK                                                       | RHYD 90 |
|        | 1070 FORMAT(I5,I9,F11.0,F8.0,F9.1,5F10.3,F9.2,F8.2,F12.5,I5)          | RHYD 91 |
|        | IF(N.GT.NW) GO TO 320                                                 | RHYD 92 |
| C      |                                                                       | RHYD 93 |
| C***** | TRANSFER DATA AND CONVERT UNITS                                       | RHYD 94 |
|        | NHYET(N)=JK                                                           | RHYD 95 |
|        | WWIDTH(N)=W1                                                          | RHYD 96 |
|        | WAREA(N)=W2*43560.                                                    | RHYD 97 |
|        | PCIMP(N)=W3                                                           | RHYD 98 |
|        | WSLOPE(N)=W4                                                          | RHYD 99 |
|        | WSTORE(N,1)=W7/12.                                                    | RHYD100 |
|        | WSTORE(N,2)=W8/12.                                                    | RHYD101 |
|        | WLMAX(N)=W9/43200.                                                    | RHYD102 |
|        | WLMIN(N)=W10/43200.                                                   | RHYD103 |
|        | DECAY(N)=W11                                                          | RHYD104 |
|        | WCON(N,1)=-(1.486/W5)*SQRT(W4)*W1*100./(WAREA(N)*PCIMP(N))            | RHYD105 |
|        | WCON(N,2)=-(1.486/W6)*SQRT(W4)*W1*100./(WAREA(N)*(100.-PCIMP(N)))     | RHYD106 |
|        | WSTORE(N,3)=0.                                                        | RHYD107 |
|        | WCON(N,3)=WCON(N,1)                                                   | RHYD108 |
|        | TAREA=TAREA+W2                                                        | RHYD109 |
| C      |                                                                       | RHYD110 |
| C***** | SET LP CONNECTIVITY TABLES                                            | RHYD111 |
|        | IF(NGOTO.EQ.0) GO TO 260                                              | RHYD112 |
| C***** | GUTTER CONNECTION                                                     | RHYD113 |
|        | DO 240 J=1,NIN                                                        | RHYD114 |
|        | IF(NWTOG(NGOTO,J).GT.0) GO TO 240                                     | RHYD115 |
|        | NWTOG(NGOTO,J)=N                                                      | RHYD116 |
|        | GO TO 340                                                             | RHYD117 |
|        | 240 CONTINUE                                                          | RHYD118 |
| C***** | DIRECT INLET CONNECTION                                               | RHYD119 |
|        | 260 DO 280 J=1,NIN                                                    | RHYD120 |

```

      IF(NWTOI(J).GT.C) GO TC 280
      NWTOI(J)=N
      GO TO 340
280 CONTINUE
C
C***** ERROR IN DATA
      320 NSTOP=1
      WRITE(6,1090) JK,N,NW
      1090 FORMAT(' FOR INLET ',I4,5X,'THE ASSIGNED WATERSHED NO',I10,5X,
      1' IS LARGER THAN THE ALLOWED NO',I10)
      340 CONTINUE
      360 WRITE(6,1100)NOW,TAREA
      1100 FORMAT('OTAL NUMBER OF SUBCATCHMENTS,',I4/
      1'OTAL TRIBUTARY AREA (ACRES)',F8.2)
C
C***** GLTTER AND PIPE DATA
C
      NOG=0
      WRITE(6,1110)
      1110 FORMAT('1GUTTER      GUTTER      WIDTH      LENGTH      SLOPE
      1DE SLOPES      MANNING      OVERFLOW'//
      2      ' NUMBER      CONNECTION      (FT)      (FT)      (FT/FT)
      3L      R      N      (IN)'//)
      DO 480 I=1,NG
      READ(5,1115)JK,N,NGOTC,NP,G1,G2,G3,G4,G5,G6,G7
      1115 FORMAT(4I5,7F8.0)
      IF(N.EQ.C) GO TC 500
      NOG=NCG+1
C
C***** PRINT CUTTER/PIPE DATA
      WRITE(6,1120) N,NGOTO, G1,G2,G3,G4,G5,G6,G7
      1120 FORMAT(15,I12,F14.1,F10.0,F10.3,F9.1,F6.1,F11.3,F12.2)
      IF(NP.EQ.2) WRITE(6,1122)
      1122 FORMAT('+      *')
      IF(N.GT.NG) GO TO 440
C
C***** TRANSFER DATA AND CONVERT UNITS
      NPG(N)=NP
      GWIDTH(N)=G1
      GLEN(N)=G2
      GSLOPE(N)=G3
      GS1(N)=G4
      GS2(N)=G5
      GN(N)=G6
      DFULL(N)=G7
      IF(NP.EQ.2) DFULL(N)=2.62
      IF(DFULL(N).LE.0.) DFULL(N)=10.
      GCON(N)=(1.486/GN(N))*SQRT(GSLOPE(N))
C
C***** SET LP CONNECTIVITY TABLES
      IF(NGCTC.EQ.C) GO TO 400
C
C***** GLTTER CONNECTION
      DO 380 J=1,NIN
      IF(NGTCG(NGCTC,J).GT.0) GO TC 380
      NGTCG(NGCTC,J)=N
      GO TO 480
      380 CONTINUE
C
C***** INLET CONNECTION

```

```

RHYD121
RHYD122
RHYD123
RHYD124
RHYD125
RHYD126
RHYD127
RHYD128
RHYD129
RHYD130
RHYD131
RHYD132
RHYD133
RHYD134
RHYD135
RHYD136
RHYD137
RHYD138
RHYD139
RHYD140
RHYD141
RHYD142
RHYD143
RHYD144
RHYD145
RHYD146
RHYD147
RHYD148
RHYD149
RHYD150
RHYD151
RHYD152
RHYD153
RHYD154
RHYD155
RHYD156
RHYD157
RHYD158
RHYD159
RHYD160
RHYD161
RHYD162
RHYD163
RHYD164
RHYD165
RHYD166
RHYD167
RHYD168
RHYD169
RHYD170
RHYD171
RHYD172
RHYD173
RHYD174
RHYD175
RHYD176
RHYD177
RHYD178
RHYD179
RHYD180

```

|                                                                     |         |
|---------------------------------------------------------------------|---------|
| 400 DO 420 J=1,NIN                                                  | RHYD181 |
| IF(NGTCG(J).GT.C) GO TO 420                                         | RHYD182 |
| NGTOI(J)=N                                                          | RHYD183 |
| GO TO 480                                                           | RHYD184 |
| 420 CONTINUE                                                        | RHYD185 |
| C                                                                   | RHYD186 |
| C***** ERROR IN DATA                                                | RHYD187 |
| 440 NSTOP=1                                                         | RHYD188 |
| WRITE(6,1130) JK,N,NG                                               | RHYD189 |
| 1130 FORMAT(' FOR INLET',15,5X,'THE ASSIGNED GUTTER NO',110,5X,     | RHYD190 |
| 1'IS LARGER THAN ALLOWED NO',110)                                   | RHYD191 |
| 480 CONTINUE                                                        | RHYD192 |
| 500 WRITE(6,1150) NCG                                               | RHYD193 |
| C                                                                   | RHYD194 |
| C                                                                   | RHYD195 |
| C                                                                   | RHYD196 |
|                                                                     | RHYD197 |
| DO 540 N=1,NG                                                       | RHYD198 |
| IF(NWTOG(N,1).EQ.0.AND.NGTOG(N,1).EQ.0) GO TO 540                   | RHYD199 |
| IF(NPG(N).NE.0) GO TO 540                                           | RHYD200 |
| NPG(N)=3                                                            | RHYD201 |
| NCG=NCG+1                                                           | RHYD202 |
| DO 520 J=1,NING                                                     | RHYD203 |
| IF(NGTOI(J).GT.0) GO TO 520                                         | RHYD204 |
| NGTOI(J)=N                                                          | RHYD205 |
| GO TO 540                                                           | RHYD206 |
| 520 CONTINUE                                                        | RHYD207 |
| 540 CONTINUE                                                        | RHYD208 |
| 1150 FORMAT('TOTAL NUMBER OF GUTTERS/PIPES',14)                     | RHYD209 |
| WRITE(6,1151)                                                       | RHYD210 |
| 1151 FORMAT('ASTERISK (*) DENOTES CIRCULAR PIPE, DIAMETER=.WIDTH.') | RHYD211 |
| 600 CONTINUE                                                        | RHYD212 |
| C                                                                   | RHYD213 |
| C**** PRINT CONNECTIVITY SUMMARY                                    | RHYD214 |
| C                                                                   | RHYD215 |
| WRITE(6,1190)                                                       | RHYD216 |
| 1190 FORMAT('ARRANGEMENT OF SUBCATCHMENTS AND GUTTERS/PIPES'//      | RHYD217 |
| 1' GUTTER', 5X,'TRIBUTARY GUTTER/PIPE',40X,                         | RHYD218 |
| 2'TRIBUTARY SUBAREA')                                               | RHYD219 |
| DO 620 J=1,NG                                                       | RHYD220 |
| IF(GLEN(J).LE.0.) GO TO 620                                         | RHYD221 |
| DO 605 N=1,NIN                                                      | RHYD222 |
| IF(NGTCG(J,N)) 605,606,605                                          | RHYD223 |
| 605 CONTINUE                                                        | RHYD224 |
| 606 N=N-1                                                           | RHYD225 |
| IF(N) 607,607,608                                                   | RHYD226 |
| 607 WRITE(6,1200) J                                                 | RHYD227 |
| GO TO 609                                                           | RHYD228 |
| 608 WRITE(6,1200) J,(NGTOG(J,I),I=1,N)                              | RHYD229 |
| 1200 FORMAT(/110,5X,1015/(1015))                                    | RHYD230 |
| 609 DO 610 N=1,NIN                                                  | RHYD231 |
| IF(NWTOG(J,N)) 610,615,610                                          | RHYD232 |
| 610 CONTINUE                                                        | RHYD233 |
| 615 N=N-1                                                           | RHYD234 |
| IF(N) 620,620,616                                                   | RHYD235 |
| 616 WRITE(6,1230) (NWTOG(J,I),I=1,N)                                | RHYD236 |
| 1230 FORMAT(1H+,74X,1015)                                           | RHYD237 |
| 620 CONTINUE                                                        | RHYD238 |
| WRITE(6,1240)                                                       | RHYD239 |
| 1240 FORMAT('C INLET',6X,'TRIBUTARY GUTTER-PIPE-MANHOLE',           | RHYD240 |
| 132X,'TRIBUTARY SUBAREA')                                           |         |

|                                                                        |          |
|------------------------------------------------------------------------|----------|
| DO 628 N=1,NING                                                        | RHYD241  |
| IF(NGTCI(N)) 628,630,628                                               | RHYD242  |
| 628 CONTINUE                                                           | RHYD243  |
| 630 N=N-1                                                              | RHYD244  |
| IF(N) 634,634,633                                                      | RHYD245  |
| 633 WRITE(6,1200) INLET,(NGTOI(I),I=1,N)                               | RHYD246  |
| GO TO 635                                                              | RHYD247  |
| 634 WRITE(6,1200) INLET                                                | RHYD248  |
| 635 DO 636 N=1,NIN                                                     | RHYD249  |
| IF(NWTCI(N)) 636,638,636                                               | RHYD250  |
| 636 CONTINUE                                                           | RHYD251  |
| 638 N=N-1                                                              | RHYD252  |
| IF(N) 640,640,639                                                      | RHYD253  |
| 639 WRITE(6,1230) (AWTOI(I),I=1,N)                                     | RHYD254  |
| 640 CONTINUE                                                           | RHYD255  |
| C                                                                      | RHYD256  |
| C                                                                      | RHYD257  |
| C                                                                      | RHYD258  |
| C                                                                      | RHYD259  |
|                                                                        | RHYD260  |
| NTQUAL=NSCRAT(1)                                                       | RHYD261  |
| REWIND NTQUAL                                                          | RHYD262  |
| READ(5,1205) NSAVE                                                     | RHYD263  |
| IF(NSAVE.LT.1) GO TO 660                                               | RHYD264  |
| READ(5,1205) (ISAVE(N),N=1,NSAVE)                                      | RHYD265  |
| 1205 FORMAT(16I5)                                                      | RHYD266  |
| WRITE(6,1210) NSAVE,(ISAVE(N),N=1,NSAVE)                               | RHYD267  |
| 1210 FORMAT( '0HYDROGRAPHS WILL BE STORED FOR THE FOLLOWING',I5,       | RHYD268  |
| 1' POINTS',(10I10))                                                    | RHYD269  |
| NOUT=JCUT(IOUTCT)                                                      | RHYD270  |
| REWIND NCUT                                                            | RHYD271  |
| WRITE(NOUT) TITLE                                                      | RHYD272  |
| NQUAL=C                                                                | RHYD273X |
| WRITE(NCUT) NSTEP,NSAVE,NQUAL,DELT,TZERO,TAREA                         | RHYD274  |
| WRITE(NOUT) (ISAVE(N),N=1,NSAVE)                                       | RHYD275  |
| WRITE(NTQUAL) NSAVE,(ISAVE(N),N=1,NSAVE)                               | RHYD276  |
| 660 CONTINUE                                                           | RHYD277  |
| READ(5,1205) NPRNT,INTERV                                              | RHYD278  |
| IF(NPRNT.LT.1) GO TO 680                                               | RHYD279  |
| READ(5,1205) (IPRNT(N),N=1,NPRNT)                                      | RHYD280  |
| WRITE(6,1220) NPRNT,(IPRNT(N),N=1,NPRNT)                               | RHYD281  |
| 1220 FORMAT( '1HYDROGRAPHS ARE LISTED FOR THE FOLLOWING',I5,' POINTS', | RHYD282  |
| 1/' TIME ',10I10/(10X,10I10))                                          | RHYD283  |
| WRITE(6,1225)                                                          | RHYD284  |
| 1225 FORMAT(1F )                                                       | RHYD285  |
| 680 CONTINUE                                                           | RHYD286  |
| C                                                                      | RHYD287  |
| C***** EXIT IF ERROR HAS BEEN DETECTED                                 | RHYD288  |
| IF(NSTCP.EQ.1) STOP                                                    | RHYD289  |
| RETURN                                                                 | RHYD290  |
| END                                                                    |          |

|                                                                  |         |
|------------------------------------------------------------------|---------|
| SUBROUTINE WSHED                                                 | WSHE 1  |
| C                                                                | WSHE 2  |
| C THIS SUBROUTINE COMPUTES THE INSTANTANEOUS WATER DEPTH         | WSHE 3  |
| C AND FLCW RATE FOR THE WATERSHED AREAS                          | WSHE 4  |
| C                                                                | WSHE 5  |
| C***** SPECIFICATION STATEMENTS                                  | WSHE 6  |
| C                                                                | WSHE 7  |
| COMMON NW,AG,NIN,HISTOG,TRAIN,DELT,DELT2,NOW,NOG,NSTEP,TAREA,    | WSHE 8  |
| 1 TIME,TIME2,RI,RLOSS,SUMR,SUMI,SUMOFF,SUMST,TZERO,NING          | WSHE 9  |
| COMMON WFLOW(160),WWIDTH(160),WAREA(160),WSLOPE(160),WN(160),    | WSHE 10 |
| 1 WSTORE(160,3),WLMAX (160),WLMIN(160),DECAY(160),WDEPTH(160,3), | WSHE 11 |
| 2 WCON(160,3),NAMEW(160),PCIMP(160)                              | WSHE 12 |
| COMMON GFLOW(160),GWIDTH(160),GLEN(160),GSLOPE(160),GS1(160),    | WSHE 13 |
| 1 GS2(160),GN(160),GDEPTH(160),GCON(160),NPG(160),DFULL(160),    | WSHE 14 |
| 2 NGUT(160),SUMQH(160),PCTZER                                    | WSHE 15 |
| COMMON NWTG(160,10),AGTG(160,10),NWTI(10),NGTI(160)              | WSHE 16 |
| COMMON RAIN(160,10),NHET(160),NRAIN,NRGAG,NHISTO,THISTO          | WSHE 17 |
| COMMON QSUR(160),DELD(160),QIN(160)                              | WSHE 18 |
| COMMON IPRNT(160),ISAVE(160),NPRNT,NSAVE,OUTFLW(160),INTERV,     | WSHE 19 |
| 1 INTCNT                                                         | WSHE 20 |
| C                                                                | WSHE 21 |
| C***** SELECT AVERAGE RAINFALL DURING TIME INTERVAL              | WSHE 22 |
| C                                                                | WSHE 23 |
| IND=1.+(TIME2-TZERO)/HISTOG                                      | WSHE 24 |
| C                                                                | WSHE 25 |
| C***** BEGIN MAJOR LOOP FOR WSHED                                | WSHE 26 |
| C                                                                | WSHE 27 |
| DO 320 N=1,NOW                                                   | WSHE 28 |
| J=NAMEW(N)                                                       | WSHE 29 |
| RI=0.                                                            | WSHE 30 |
| NGAG=NHET(J)                                                     | WSHE 31 |
| IF(TIME2.LE.TRAIN) RI=RAIN(IND,NGAG)                             | WSHE 32 |
| DELR=0.                                                          | WSHE 33 |
| WFLOW(J)=0.                                                      | WSHE 34 |
| IF(WAREA(J).EQ.0.) GO TO 320                                     | WSHE 35 |
| DO 315 K=1,3                                                     | WSHE 36 |
| IF(K-2) 201,205,210                                              | WSHE 37 |
| 201 WAR=WAREA(J)*PCIMP(J)/10000. *(100.-PCTZER)                  | WSHE 38 |
| RLOSS=C.                                                         | WSHE 39 |
| GO TO 220                                                        | WSHE 40 |
| 205 WAR=WAREA(J)*{(100.-PCIMP(J))/100.                           | WSHE 41 |
| RLOSS=WLMIN(J)                                                   | WSHE 42 |
| GO TO 215                                                        | WSHE 43 |
| 210 WAR=WAREA(J)*PCIMP(J)/10000.*PCTZER                          | WSHE 44 |
| RLOSS=0.                                                         | WSHE 45 |
| GO TO 220                                                        | WSHE 46 |
| C                                                                | WSHE 47 |
| C***** COMPLETE AVERAGE INFILTRATION DURING TIME INTERVAL        | WSHE 48 |
| C                                                                | WSHE 49 |
| 215 EXPON=DECAY(J)*TIME2                                         | WSHE 50 |
| IF(EXPON.LT.60.) RLOSS=RLOSS+(WLMAX(J)-WLMIN(J))/EXP(EXPON)      | WSHE 51 |
| IF((RI-RLOSS)*DELT+WDEPTH(J,K) .GT.0.) GO TO 220                 | WSHE 52 |
| C                                                                | WSHE 53 |
| C***** INFILTRATION LOSS EXCEEDS AVAILABLE WATER                 | WSHE 54 |
| RLOSS=RI+WDEPTH(J,K)/DELT                                        | WSHE 55 |
| WDEPTH(J,K)=0.                                                   | WSHE 56 |
| WFLQ=0.                                                          | WSHE 57 |
| GO TO 210                                                        | WSHE 58 |
| C                                                                | WSHE 59 |
| C***** COMPUTE CHANGE IN DEPTH (NEWTON-RAPHSON)                  | WSHE 60 |

|        |                                                                       |         |
|--------|-----------------------------------------------------------------------|---------|
| C      | 220 IF((RI-RLCSS)*DELT+WDEPTH(J,K).LE.WSTORE(J,K)) GO TO 285          | WSHE 61 |
|        | DO 260 I=1,11                                                         | WSHE 62 |
|        | DO=WDEPTH(J,K)-WSTORE(J,K)+0.5*DELR                                   | WSHE 63 |
|        | IF(DO.LT.0.) DO=0.                                                    | WSHE 64 |
|        | F=DELR-DELT*(WCCN(J,K)*DO**1.6666667+(RI-RLOSS))                      | WSHE 65 |
|        | DF=1.-DELT*(0.83333333*WCON(J,K)*DO**0.6666667)                       | WSHE 66 |
|        | DEL=DELR-F/DF                                                         | WSHE 67 |
|        | IF(1.EC.1) GO TO 240                                                  | WSHE 68 |
|        | IF((ABS(DEL-DELR)).LT.(ABS(0.01*DELR))) GO TO 280                     | WSHE 69 |
|        | 240 DELR=DEL                                                          | WSHE 70 |
|        | 260 CONTINUE                                                          | WSHE 71 |
|        | IF(DELR.LT..0001*WDEPTH(J,K)) GO TO 280                               | WSHE 72 |
|        | WRITE(6,1000) TIME,J,WDEPTH(J,K),DELR                                 | WSHE 73 |
|        | 1000 FORMAT(' CHECK RESULTS. NO CONVERGENCE IN *WSHED*F8.0,16,2E12.5) | WSHE 74 |
|        | 280 DCORR=WDEPTH(J,K)+DEL                                             | WSHE 75 |
|        | DELR=0.                                                               | WSHE 76 |
| C      |                                                                       | WSHE 77 |
| C***** | AVERAGE FLOW DURING TIME INTERVAL                                     | WSHE 78 |
| C      |                                                                       | WSHE 79 |
|        | WFLO=(RI-RLOSS)*WAR-(DCORR-WDEPTH(J,K))*WAR/DELT                      | WSHE 80 |
|        | IF(WFLC.GT.0.) GO TO 290                                              | WSHE 81 |
|        | 285 WFLO=0.                                                           | WSHE 82 |
|        | DCORR=WDEPTH(J,K)+(RI-RLOSS)*DELT                                     | WSHE 83 |
| C      |                                                                       | WSHE 84 |
| C***** | TRANSFER DEPTH FOR NEXT TIME INTERVAL                                 | WSHE 85 |
| C      |                                                                       | WSHE 86 |
|        | 290 WDEPTH(J,K)=DCORR                                                 | WSHE 87 |
| C***** | SUM FOR CONTINUITY CHECK                                              | WSHE 88 |
|        | 310 SUMR=SUMR+RI*DELT*WAR                                             | WSHE 89 |
|        | SUMI=SUMI+RLOSS*DELT*WAR                                              | WSHE 90 |
|        | WFLOW(J)=WFLOW(J)+WFLO                                                | WSHE 91 |
|        | 315 CONTINUE                                                          | WSHE 92 |
|        | 320 CONTINUE                                                          | WSHE 93 |
| C      |                                                                       | WSHE 94 |
|        | RETURN                                                                | WSHE 95 |
|        | END                                                                   | WSHE 96 |
|        |                                                                       | WSHE 97 |



|                                                            |                                                                        |         |
|------------------------------------------------------------|------------------------------------------------------------------------|---------|
| SUBROUTINE SFQUAL                                          |                                                                        | SFQU 1  |
| C=====METCALF & EDDY ENGINEERS' SURFACE QUALITY MODEL===== |                                                                        | SFQU 2  |
| C                                                          | COMMON /TAPES/ INCNT,ICUTCT,JIN(10),JOUT(10),NSCRAT(5)                 | SFQU 3  |
|                                                            | COMMON NH,NG,NIN,HISTCG,TRAIN,DELT,DELT2,NOW,NOG,NSTEP,TAREA,          | SFQU 4  |
|                                                            | 1 TIME,TIME2,RI,RLOSS,SUMR,SUMI,SUMOFF,SUMST,NSHED,NING                | SFQU 5  |
|                                                            | COMMON KNUM(160),INPUT(160),ASUB(160),GUTTER(160),KLAND(160) ,         | SFQU 6  |
|                                                            | * TOTDD(160),REMD(160),                                                | SFQU 7  |
|                                                            | * PD(160),                                                             | SFQU 8  |
|                                                            | * RUNCFS(2,160),AVGFLO(160),                                           | SFQU 9  |
|                                                            | * POP(160),POPSS(160),                                                 | SFQU 10 |
|                                                            | * CBSUM(160),CBLBS(2,160),CBINC(160),                                  | SFQU 11 |
|                                                            | * CBASTP(160),ATOT(160),C(160)                                         | SFQU 12 |
|                                                            | * ,SFCOLI(160),IKOUNT(160),CCOLI(160),TCCOLI(160)                      | SFQU 13 |
|                                                            | COMMON JN(160),RUNTMP(160),IPCINT(160),BOD(160),SS(160)                | SFQU 14 |
|                                                            | COMMON TITLE(40)                                                       | SFQU 15 |
|                                                            | COMMON TPCPSS(160),TCBINC(160),TCBAST(160),TPOP(160)                   | SFQU 16 |
|                                                            | DATA NQAL/3/                                                           | SFQU 17 |
| C                                                          | READ NUMBER OF SUBAREAS , TIME INTERVAL IN MINUTES, AND STARTING TIMES | SFQU 18 |
| C                                                          | OF STORM.                                                              | SFQU 19 |
|                                                            | READ(C5,2CC)KTNUM,NINLTS,DT,KHOUR,KMIN,NTSTEP,NPRINT                   | SFQU 20 |
| 200                                                        | FORMAT(2I5,F5.0,4I5)                                                   | SFQU 21 |
|                                                            | IFPRNT=NSCRAT(3)                                                       | SFQU 22 |
|                                                            | REWIND IFPRNT                                                          | SFQU 23 |
|                                                            | WRITE(C6,601) KTNUM,NINLTS,DT,KHOUR,KMIN                               | SFQU 24 |
| 601                                                        | FORMAT('1NUMBER OF SUBAREAS, KTNUM =', I4, /                           | SFQU 25 |
|                                                            | * ' NUMBER OF INLETS, NINLTS =', I4, /                                 | SFQU 26 |
|                                                            | * ' TIME INTERVAL (MIN), DT =', F6.2, /                                | SFQU 27 |
|                                                            | * ' STORM START TIME (HR:MIN) =', I3, ':', I2)                         | SFQU 28 |
| C                                                          | READ IN DATA FOR ESTIMATING AVAILABLE BOD AND DUST AND DIRT            | SFQU 29 |
|                                                            | READ(C5,2C1) DRYDAY,CLFREQ,NUPASS                                      | SFQU 30 |
| 201                                                        | FORMAT (2F10.0,15)                                                     | SFQU 31 |
|                                                            | WRITE(C6,602) DRYDAY,CLFREQ,NUPASS                                     | SFQU 32 |
| 602                                                        | FORMAT('0DRYDAY =', F10.0, ' , CLFREQ=', F10.0, ' , NUPASS =', I5)     | SFQU 33 |
|                                                            | IF(DRYDAY-CLFREQ) 10,10,11                                             | SFQU 34 |
| 11                                                         | CLEAN=DRYDAY/CLFREQ                                                    | SFQU 35 |
|                                                            | NCLEAN=CLEAN                                                           | SFQU 36 |
|                                                            | IF(NOPASS-2) 13,14,15                                                  | SFQU 37 |
| 13                                                         | IF(CLFREQ.GT.15.) REFF=0.60                                            | SFQU 38 |
|                                                            | IF(CLFREQ.LE.15.) REFF=0.70                                            | SFQU 39 |
|                                                            | IF(CLFREQ.LE.7.) REFF=0.75                                             | SFQU 40 |
|                                                            | GO TO 10                                                               | SFQU 41 |
| 14                                                         | IF(CLFREQ.GT.15.) REFF=0.88                                            | SFQU 42 |
|                                                            | IF(CLFREQ.LE.15.) REFF=0.92                                            | SFQU 43 |
|                                                            | IF(CLFREQ.LE.7.) REFF=0.95                                             | SFQU 44 |
|                                                            | GO TO 10                                                               | SFQU 45 |
| 15                                                         | REFF=0.98                                                              | SFQU 46 |
| 10                                                         | CONTINUE                                                               | SFQU 47 |
| C                                                          | CATCHBASIN CONTRIBUTIONS                                               | SFQU 48 |
| C                                                          | READ CB DENSITY (AVE NO PER ACRE), CONCENTRATION OF BOD IN MG/L        | SFQU 49 |
| C                                                          | AND STORED VOLUME OF LIQUID IN GALLONS                                 | SFQU 50 |
|                                                            | READ(C5,2I0) CBDEN,CBBOD,CBVOL                                         | SFQU 51 |
| 210                                                        | FORMAT(3F10.0)                                                         | SFQU 52 |
|                                                            | WRITE(C6,603) CBDEN, CBBOD, CBVOL                                      | SFQU 53 |
| 603                                                        | FORMAT('CAVERAGE NO. CB/ACRE, CBDEN =', F10.0, /                       | SFQU 54 |
|                                                            | * ' CB CONTENTS BOD (MG/L), CBBOD =', F10.0, /                         | SFQU 55 |
|                                                            | * ' CB STORED VOLUME (GAL), CBVOL =', F10.0 )                          | SFQU 56 |
|                                                            | IF(CBDEN.EQ.0.0) GO TO 19                                              | SFQU 57 |
|                                                            | PQOCB=CBVOL*CBBOD*8.34/1000000.                                        | SFQU 58 |
|                                                            |                                                                        | SFQU 59 |
|                                                            |                                                                        | SFQU 60 |

|     |                                                                    |         |
|-----|--------------------------------------------------------------------|---------|
| C   | PQOCB=LBS BOD AVAIL IN EACH CB AT START OF STORM                   | SFQU 61 |
| C   | CB'S ARE EXPECTED TO HAVE A CONSTANT CONCENTRATION REGARDLESS OF   | SFQU 62 |
| C   | LAND USE                                                           | SFQU 63 |
| C   | IF DRYDAY IS GREATER THAN 1.0                                      | SFQJ 64 |
|     | IF(DRYDAY.GE.1) GO TO 19                                           | SFQU 65 |
|     | PQOCB=SQRT(DRYDAY)*PQOCB                                           | SFQU 66 |
|     | 19 CONTINUE                                                        | SFQU 67 |
| C   |                                                                    | SFQU 68 |
| C   |                                                                    | SFQJ 69 |
|     | WRITE(06,778)                                                      | SFQU 70 |
| 778 | FORMAT('CKNUM/INPUT/KLAND ASUB GUTTER')                            | SFQU 71 |
|     | DO 300 K=1,KNUM                                                    | SFQJ 72 |
|     | READ(05,202) KNUM(K),INPUT(K),KLAND(K),ASUB(K),GUTTER(K)           | SFQU 73 |
|     | WRITE(6,202) KNUM(K),INPUT(K),KLAND(K),ASUB(K),GUTTER(K)           | SFQU 74 |
| 202 | FORMAT (3I5,2F10.2)                                                | SFQU 75 |
| C   | GUTTER EQUALS GUTTER LENGTH IN 100'S OF FEET. DD=DUST AND DIRT     | SFQJ 76 |
| C   | ACCUMULATION PER DAY. FACTORS ARE FROM CHICAGO APWA STUDY.         | SFQJ 77 |
|     | IF(KLAND(K).EQ.1) DD =0.7*GUTTER(K)                                | SFQU 78 |
|     | IF(KLAND(K).EQ.2) DD =2.3*GUTTER(K)                                | SFQJ 79 |
|     | IF(KLAND(K).EQ.3) DD =3.3*GUTTER(K)                                | SFQU 80 |
|     | IF(KLAND(K).EQ.4) DD =4.6*GUTTER(K)                                | SFQU 81 |
|     | IF(KLAND(K).EQ.5) DD =1.5*GUTTER(K)                                | SFQU 82 |
|     | IF(DRYDAY-CLFREQ) 16,16,17                                         | SFQJ 83 |
| 16  | TOTDD(K)=DRYDAY*DD                                                 | SFQJ 84 |
|     | GO TO 18                                                           | SFQU 85 |
| C   | GS IS A FACTOR IN A GEOMETRIC SERIES                               | SFQU 86 |
| C   | TGS IS THE SUM OF THIS GEOMETRIC SERIES PLUS 1.0                   | SFQU 87 |
| 17  | TGS=1.00                                                           | SFQJ 88 |
|     | DO 99 IK=1,NCLEAN                                                  | SFQJ 89 |
|     | GS=(1.00-REFF)**IK                                                 | SFQU 90 |
|     | TGS=TGS+GS                                                         | SFQJ 91 |
| 99  | CONTINUE                                                           | SFQU 92 |
|     | TOTDD(K)=CLFREQ*DD*TGS                                             | SFQU 93 |
| C   | DRYDAYS MUST BE ADJUSTED TO ALLOW FOR CARRYOVER FOR EARLIER STORMS | SFQU 94 |
| C   | TOTDD IS AVAILABLE DUST AND DIRT AT START OF STORM IN POUNDS       | SFQJ 95 |
| C   | AVAILABLE BOD COMPUTATION FOLLOWS                                  | SFQU 96 |
| 18  | IF(KLAND(K).EQ.1) SFBCD =5.0*TOTDD(K)/1000.                        | SFQJ 97 |
|     | IF(KLAND(K).EQ.2) SFBCD =3.6*TOTDD(K)/1000.                        | SFQJ 98 |
|     | IF(KLAND(K).EQ.3) SFBCD =7.7*TOTDD(K)/1000.                        | SFQU 99 |
|     | IF(KLAND(K).EQ.4) SFBCD =3.0*TOTDD(K)/1000.                        | SFQJ100 |
|     | IF(KLAND(K).EQ.5) SFBCD =5.0*TOTDD(K)/1000.                        | SFQJ101 |
| C   | SFBCD IS AVAILABLE SOLUBLE BOD CONTAINED IN THE DUST AND DIRT AT   | SFQJ102 |
| C   | THE START OF THE STORM ON THE SURFACE IN POUNDS.                   | SFQJ103 |
|     | PO(K)=SFBCD                                                        | SFQJ104 |
|     | REMDD(K)=TOTDD(K)                                                  | SFQJ105 |
| C   | REMDD = REMAINING DUST AND DIRT FROM PREVIOUS TIME STEP            | SFQJ106 |
| 300 | CONTINUE                                                           | SFQJ107 |
|     | A=0.0004*DT                                                        | SFQJ108 |
|     | B=0.0025*DT                                                        | SFQJ109 |
|     | TMIN=CT                                                            | SFQJ110 |
| C   | = TIME INTERVAL IN MINUTES                                         | SFQJ111 |
|     | TSEC=TMIN*60.0                                                     | SFQJ112 |
|     | T=TMIN/60.0                                                        | SFQJ113 |
|     | CONVER=1000000.0/(3600.0*T*62.4)                                   | SFQJ114 |
|     | CONV2=60.0*283.2                                                   | SFQJ115 |
| C   | CONV2 CONVERTS FLOW IN CFS TO FLOW IN 100ML/MIN                    | SFQJ116 |
| C   |                                                                    | SFQJ117 |
| C   |                                                                    | SFQJ118 |
|     | INPT=NSCRAT(1)                                                     | SFQJ119 |
|     | REWIND INPT                                                        | SFQJ120 |

|                                                             |                                  |
|-------------------------------------------------------------|----------------------------------|
| READ(INPT) ACW,(JN(N),N=1,NOW)                              | SFQU121                          |
| IF(NOW.NE.NINLTS) STOP                                      | SFQJ122                          |
| DO 1260 N=1,NINLTS                                          | SFQU123                          |
| TPUP(N)=C.0                                                 | SFQU124                          |
| TPOPSS(N)=0.0                                               | SFQU125                          |
| TGBAST(N)=0.0                                               | SFQU126                          |
| TCRINC(N)=0.0                                               | SFQU127                          |
| J=1                                                         | SFQU128                          |
| DO 1245 K=1,KTNUM                                           | SFQU129                          |
| CCOLI(K)=0.0                                                | SFQU130                          |
| TCCOLI(K)=0.0                                               | SFQU131                          |
| IKOUNT(K)=0                                                 | SFQJ132                          |
| IF(JN(N)-INPUT(K))1240,1250,1240                            | SFQU133                          |
| 1240 IF(K.EC.KTNUM) GO TO 1245                              | SFQU134                          |
| IF(INPUT(K+1).NE.INPUT(K)) J=J+1                            | SFQU135                          |
| 1245 CONTINUE                                               | SFQU136                          |
| 1250 IPUINT(N)=J                                            | SFQU137                          |
| 1260 CONTINUE                                               | SFQU138                          |
| J=1                                                         | SFQU139                          |
| DO 1270 K=1,KTNUM                                           | SFQU140                          |
| JN(J)=INPUT(K)                                              | SFQU141                          |
| IF(K.EC.KTNUM)GO TO 1270                                    | SFQJ142                          |
| IF(INPUT(K+1).NE.INPUT(K)) J=J+1                            | SFQU143                          |
| 1270 CONTINUE                                               | SFQU144                          |
| TZERO=0.                                                    | SFQU145                          |
| TZERO=3600.*FLCAT(KHOUR)+60.*FLOAT(KMIN)                    | SFQU146                          |
| NOUT=JCUT(ICUTCT)                                           | SFQU147                          |
| REWIND NCUT                                                 | SFQU148                          |
| TIME=TZERO                                                  | SFQU149                          |
| READ(NOUT) TITLE                                            | SFQU150                          |
| REWIND NCUT                                                 | SFQU151                          |
| WRITE(NCUT)(TITLE(I),I=1,40)                                | SFQJ152                          |
| WRITE(NOUT) NTSTEP,NINLTS,NQUAL,TSEC,TZERO,TAREA            | SFQJ153X                         |
| WRITE(NOUT) (JN(K),K=1,NINLTS)                              | SFQU154                          |
| WRITE(06,779)                                               | SFQU155                          |
| 779 FORMAT('1 TIME',9X,'RUNOFF IN CFS AT SELECTED POINTS    | B03SFQU156                       |
| * AT SELECTED POINTS                                        | SUSPENDED SULIOS AT SELECTED POS |
| *INTS')                                                     | SFQJ157                          |
| WRITE(06,780)((JN(IA),IA=1,4),KK=1,3)                       | SFQU158                          |
| 780 FORMAT(' ',6X,12I10)                                    | SFQU159                          |
| NOLD=1                                                      | SFQU160                          |
| NEW=2                                                       | SFQU161                          |
| TIMES=TZERO                                                 | SFQU162                          |
| MKOUNT=0                                                    | SFQJ163                          |
| DO 444 KTSTEP=1,NTSTEP                                      | SFQU164                          |
| MKOUNT=MKOUNT+1                                             | SFQU165                          |
| IF(MKOUNT.GT.NPRINT) MKOUNT=1                               | SFQU166                          |
| TIMES=TIMES+TSEC                                            | SFQU167                          |
| J=1                                                         | SFQJ168                          |
| C READ RUNOFF VALUES AT INCREMENTS =DT                      | SFQU169                          |
| READ(INPT)(RUNTMP(N),N=1,NINLTS)                            | SFQJ170                          |
| DO 203 N=1,NINLTS                                           | SFQU171                          |
| K=IPUINT(N)                                                 | SFQU172                          |
| 203 RUNCFS(NEW,K)=RUNTMP(N)                                 | SFQU173                          |
| DO 222 I=1,NINLTS                                           | SFQU174                          |
| C CALCULATE AVERAGE RUNOFF IN TIME INTERVAL                 | SFQJ175                          |
| IF(KTSTEP.EQ.1) AVGFLO(I)=RUNCFS(NEW,I)/2.0                 | SFQU176                          |
| IF(KTSTEP.GT.1)AVGFLO(I)=(RUNCFS(NOLD,I)+RUNCFS(NEW,I))/2.0 | SFQJ177                          |
| IF(KTSTEP.EQ.1) ATOT(I)=0.0                                 | SFQU178                          |
| POPSS(I)=C.C                                                | SFQU179                          |
|                                                             | SFQJ180                          |

|     |                                                                 |          |
|-----|-----------------------------------------------------------------|----------|
|     | CBASTM(I)=0.0                                                   | SFQU181  |
|     | RUNCFS(NCLD,I)=RUNCFS(NEW,I)                                    | SFQU182  |
| 222 | CONTINUE                                                        | SFQU183  |
|     | DO 333 JJ=1,KTNUM                                               | SFQU184  |
|     | SFCOLI(JJ)=0.0                                                  | SFQU185  |
|     | IKOUNT(J)=IKCUNT(J)+1                                           | SFQU186  |
| C   | ...CONVERT RUNOFF FROM CFS TO INCHES/HOUR...                    | SFQU187  |
|     | IF(KTSTEP.GT.1) GO TO 224                                       | SFQU188  |
|     | JK=JJ+1                                                         | SFQU189  |
|     | DO 223 JKL=1,5                                                  | SFQU190  |
|     | IF(ATOT(J).EQ.0.0) ATOT(J)=ASUB(JJ)                             | SFQU191  |
|     | IF(JK.GT.KTNUM) GO TO 224                                       | SFQU192X |
|     | IF(INPUT(JK).NE.INPUT(JJ)) GO TO 224                            | SFQU193  |
|     | ATOT(J)=ATOT(J)+ASUB(JK)                                        | SFQU194  |
|     | JK=JK+1                                                         | SFQU195  |
| 223 | CONTINUE                                                        | SFQU196  |
| 224 | IF(ATOT(J).GT.0.0) GO TO 226                                    | SFQU197  |
|     | RUNOFF=0.0                                                      | SFQU198  |
|     | GO TO 225                                                       | SFQU199  |
| 226 | RUNOFF=AVGFLO(J)/ATOT(J)                                        | SFQU200  |
| 225 | IF(KTSTEP.EQ.1) C(JJ)=0.90                                      | SFQU201  |
|     | IF(KTSTEP.EQ.1) CBSUM(JJ)=0.0                                   | SFQU202  |
|     | IF(GUTTER(JJ).EQ.0.0) GO TO 22                                  | SFQU203  |
|     | ERT=EXP(-4.6*RUNOFF*T)                                          | SFQU204  |
|     | P=PU(JJ)*ERT                                                    | SFQU205  |
|     | IF(RUNOFF.LE.0.0) GO TO 22                                      | SFQU206  |
|     | C(JJ)=C(JJ)-0.0025*DT                                           | SFQU207  |
|     | IF(C(JJ).LT.0.25) C(JJ)=0.25                                    | SFQU208  |
|     | POP(JJ)=PC(JJ)-P                                                | SFQU209  |
|     | D=2.60-1.25*RUNOFF**0.5                                         | SFQU210  |
|     | IF(RUNOFF.GE.1.0) D=1.0                                         | SFQU211  |
|     | E=100.0*RUNOFF                                                  | SFQU212  |
|     | IF(E.GT.300.0) E=300.0                                          | SFQU212A |
|     | AVAIL=C.C57+1.4*RUNOFF**1.1                                     | SFQU213  |
|     | IF(AVAIL.GT.0.75) AVAIL=0.75                                    | SFQU213A |
| C   | POPSS(JJ)=AVAIL*REMDD(JJ)*(1.0-ERT)                             | SFQU214  |
|     | POPSS(JJ)=ASUB(JJ)*(A+E+B*E**D)*C(JJ)*REMDD(JJ)/TOTDD(JJ)       | SFQU215  |
| C   | POPSS(KTSTEP,JJ)=PCUNDS OF SS REMOVED DURING TIME DT            | SFQU216  |
| C   | SUBSCRIPT JJ IS SUBAREA (COMMON LAND USE) FROM WHICH SOLIDS ARE | SFQU217  |
| C   | REMOVED.                                                        | SFQU218  |
|     | F=POPSS(JJ)                                                     | SFQU219  |
|     | IF(F.GE.C(JJ)*REMDD(JJ)) POPSS(JJ)=C(JJ)*REMDD(JJ)              | SFQU220  |
|     | IF(KLAND(J).EQ.1)SFCOLI(JJ)=590000000*POPSS(JJ)/DT              | SFQU221  |
|     | IF(KLAND(J).EQ.2)SFCOLI(JJ)=1220000000*POPSS(JJ)/DT             | SFQU222  |
|     | IF(KLAND(J).EQ.3)SFCOLI(JJ)=770000000*POPSS(JJ)/DT              | SFQU223  |
|     | IF(KLAND(J).GE.4)SFCOLI(JJ)=454000000*POPSS(JJ)/DT              | SFQU224  |
| C   | SFCOLI IS TOTAL COLIFORMS IN RUNOFF IN MPN/MIN                  | SFQU225  |
|     | CCOLI(JJ)=SFCOLI(JJ)/(AVGFLO(J)*CONV2)                          | SFQU226  |
|     | TCCOLI(JJ)=TCCOLI(JJ)+CCOLI(JJ)                                 | SFQU227  |
|     | BODNS=0.05*POPSS(JJ)                                            | SFQU228  |
|     | POP(JJ)=POP(JJ)+BODNS                                           | SFQU229  |
| C   | CATCHBASIN CONTRIBUTIONS                                        | SFQU230  |
|     | IF(CBDEN.EQ.0.0) GO TO 20                                       | SFQU231  |
|     | CBNUM=CBDEN*ASUE(JJ)                                            | SFQU232  |
|     | POCB=CBNUM*PODCR                                                | SFQU233  |
|     | DRAIN=(RUNCFF*ASUB(JJ)*TSEC*7.48)/CBNUM                         | SFQU234  |
| C   | DRAIN = DRAINAGE IN GALLONS TO EACH CATCHBASIN                  | SFQU235  |
|     | CBSUM(JJ)=CBSUM(JJ)+DRAIN                                       | SFQU236  |
|     | CBCENT=(1.0-EXP(-CBSUM(JJ)/(1.5*CBVOL)))                        | SFQU237  |
| C   | CBCENT IS PERCENT OF POLLUTION REMOVED/100.                     | SFQU238  |

|                                                                          |         |
|--------------------------------------------------------------------------|---------|
| CBLBS(NEW,JJ)=CBCENT*POCB                                                | SFQU239 |
| IF(KTSTEP.EQ.1) CBINC(JJ)=CBLBS(NEW,JJ)                                  | SFQU240 |
| IF(KTSTEP.GT.1) CBINC(JJ)=CBLBS(NEW,JJ)-CBLBS(NOLD,JJ)                   | SFQU241 |
| GO TO 21                                                                 | SFQU242 |
| 20 CBINC(JJ)=C.0                                                         | SFQU243 |
| 21 CONTINUE                                                              | SFQU244 |
| PRESENT COMPUTATIONS ASSUME CBS EFFECT ONLY 800                          | SFQU245 |
|                                                                          | SFQU246 |
|                                                                          | SFQU247 |
| GO TO 23                                                                 | SFQU248 |
| 22 POP(JJ)=0.0                                                           | SFQU249 |
| IF(KTSTEP.GT.1) GO TO 98                                                 | SFQU250 |
| CBLBS(NEW,JJ)=0.0                                                        | SFQU251 |
| 98 POPSS(JJ)=0.0                                                         | SFQU252 |
| CBINC(JJ)=C.0                                                            | SFQU253 |
| IF(KTSTEP.EQ.1) GO TO 505                                                | SFQU254 |
| CBLBS(NEW,JJ)=CBLBS(NOLD,JJ)                                             | SFQU255 |
| 505 CBSUM(JJ)=CBSUM(JJ)                                                  | SFQU256 |
| GO TO 23                                                                 | SFQU257 |
| 29 POP(JJ)=C.0                                                           | SFQU258 |
| POPSS(JJ)=0.0                                                            | SFQU259 |
| CBINC(JJ)=0.0                                                            | SFQU260 |
| 23 CONTINUE                                                              | SFQU261 |
| CBLBS(NOLD,JJ)=CBLBS(NEW,JJ)                                             | SFQU262 |
| PO(JJ)=P                                                                 | SFQU263 |
| REMOD(JJ)=REMOD(JJ)-PCPSS(JJ)                                            | SFQU264 |
| CBASTM(JJ)=CBINC(JJ)+POP(JJ)                                             | SFQU265 |
| IF(JJ.EQ.1) GO TO 330                                                    | SFQU266 |
| IF(INPUT(JJ-1).EQ.INPUT(JJ)) GO TO 332                                   | SFQU267 |
| C THIS IS A NEW INLET                                                    | SFQU268 |
| POPSS(J)=POPSS(JJ)                                                       | SFQU269 |
| CBINC(J)=CBINC(JJ)                                                       | SFQU270 |
| POP(J)=PCP(JJ)                                                           | SFQU271 |
| CBASTM(J)=CBASTM(JJ)                                                     | SFQU272 |
| SFCOLI(J)=SFCOLI(JJ)                                                     | SFQU273 |
| TCCOLI(J)=TCCOLI(JJ)                                                     | SFQU274 |
| CCOLI(J)=CCOLI(JJ)                                                       | SFQU275 |
| GO TO 330                                                                | SFQU276 |
| 332 CONTINUE                                                             | SFQU277 |
| POPSS(J)=POPSS(J)+POPSS(JJ)                                              | SFQU278 |
| CBINC(J)=CBINC(J)+CBINC(JJ)                                              | SFQU279 |
| POP(J)=PCP(J)+PCP(JJ)                                                    | SFQU280 |
| CBASTM(J)=CBASTM(J)+CBASTM(JJ)                                           | SFQU281 |
| SFCOLI(J)=SFCOLI(J)+SFCOLI(JJ)                                           | SFQU282 |
| TCCOLI(J)=TCCOLI(J)+TCCOLI(JJ)                                           | SFQU283 |
| CCOLI(J)=CCOLI(J)+CCOLI(JJ)                                              | SFQU284 |
| 330 CONTINUE                                                             | SFQU285 |
| IF(KTSTEP.EQ.NTSTEP) KNUM(J)=INPUT(JJ)                                   | SFQU286 |
| IF(JJ.EQ.KTNUM) GO TO 333                                                | SFQU287 |
| IF(INPUT(JJ+1).NE.INPUT(JJ)) J=J+1                                       | SFQU288 |
| 333 CONTINUE                                                             | SFQU289 |
| DO 440 J=1,NINLTS                                                        | SFQU290 |
| BOD(J)=CBASTM(J)/DT                                                      | SFQU291 |
| IF(NPRINT.EQ.0) GO TO 440                                                | SFQU292 |
| IF(KTSTEP.EQ.1) GO TO 439                                                | SFQU293 |
| IF(MKOUNT.LT.NPRINT) GO TO 440                                           | SFQU294 |
| 439 WRITE(IFPRINT)RUNCFS(NEW,J),AVGFLO(J),POPSS(J),CBINC(J),POP(J),CBASS | SFQU295 |
| *TM(J),CCOLI(J)                                                          | SFQU296 |
| 440 SS(J)=POPSS(J)/DT                                                    | SFQU297 |
| WRITE(NOUT) TIMES,(RUNCFS(NEW,J),J=1,NINLTS),(BOD(J),SS(J),SFCOLI        | SFQU298 |

```
*J), J=1, NINLTS) SFQU299
DO 442 J=1, NINLTS SFQU300
TPOP(J)=TPCP(J)+PCP(J) SFQU301
TPOPPSS(J)=TPOPS(S(J))+POPS(S(J) SFQU302
TCBAST(J)=TCBAST(J)+CBASTM(J) SFQU303
TCBINC(J)=TCBINC(J)+CBINC(J) SFQU304
BOD(J)=BOD(J)*DT SFQU305
442 SS(J)=SS(J)*DT SFQU306
WRITE(6,7C93) TIMES, (RUNCFS(NEW,IA), IA=1,4), (BOD(IA), IA=1,4), SFQU307
1(SS(IA), IA=1,4) SFQU308
7C93 FORMAT(' ', F8.0, 12F10.2) SFQU309
444 CONTINUE SFQU310
KSKIP=C SFQU311
LL=1 SFQU312
KOUNT=0 SFQU313
TTPSS=0.0 SFQU314
TTCBNC=0.0 SFQU315
TTPOP=C.0 SFQU316
TTCBST=0.0 SFQU317
TSUMDD=0.0 SFQU318
TSMBD=C.C SFQU319
TSUMCB=C.0 SFQU320
TBOD=0.0 SFQU321
WRITE(6,614) NTSTEP SFQU322
614 FORMAT('1', 10X, 'TOTAL QUANTITIES TRANSFERRED TO THE SEWER FROM THE SFQU323
* INDICATED AREAS DURING', I4, ' TIME STEPS', //) SFQU324
WRITE(6,630) SFQU325
630 FORMAT('0', T20, ' KLAND', T38, 'TYPE OF LAND USE'/' ', T20, '-----', SFQU326
*T32, '-----'/'0', T23, '1', T33, 'SINGLE FAMILY RSFQJ327
*ESIDENTIAL'/'0', T23, '2', T33, 'MULTI-FAMILY RESIDENTIAL'/'0', T23, '3'SFQU328
*, T33, 'COMMERCIAL'/'0', T23, '4', T33, 'INDUSTRIAL'/'0', T23, '5', T33, 'UNSFQJ329
*DEVELOPED OR PARKLAND', ///) SFQU330
WRITE(6,605) SFQU331
605 FORMAT(' ', T89, 'A M O U N T T R A N S F E R R E D'/' ', T82, '---SFQU332
*-----'/' ', T124, 'AVERAGESFQU333
*/' ', T54, 'BOD PRIOR TO STORM, LBS.', T83, 'SUSPENDED', T123, 'COLIFORMSFQJ334
*M'/' ', T9, 'LAND', T40, 'DUST & DIRT', T52, '-----SFQU335
*----', T84, 'SOLIDS', T95, 'FIVE DAY BOD', T125, 'CONC.'/' ', T9SFQU336
*, 'USE', T19, 'AREA', T26, 'GUTTER LENGTH PRIOR TO', T65, 'NON-', T76, 'INSFQU337
*, T84, '(PCPSS) (CBINC) + (POP) = (CBASTM) (CCOLI)'/' ', T2, 'INSFQU338
*PUT (KLAND) ACRES HUNDRED FEET STORM, LBS. SOLUBLE SOLUBLSFQJ339
*E CATCHBASIN PCUNDS PCUNDS POUNDS POUNDS MPN/100ML'/' SFQU340
*, '-----SFQJ341
*-----SFQJ342
*--'//) SFQU343
DO 666 J=1, NINLTS SFQU344
SUMDD = 0.0 SFQU345
SUMBOD = 0.0 SFQU346
SUMCB = 0.0 SFQU347
IF(NPRINT.NE.0) GO TO 613 SFQU348
GO TO 615 SFQU349
613 WRITE(6,2C9) KNUM(J), NTSTEP SFQU350
209 FORMAT('1'/10X, 'TOTAL QUANTITIES REMOVED FROM THE AREA SERVING INPSFQU351
*UT NJ. ', I3, ' ', '/32X, 'DURING ', I3, ' TIME STEPS'//) SFQU352
615 CONTINUE SFQU353
DO 334 KL=LL, KTAUM SFQU354
KK=KL SFQU355
335 IF(KLAND(KK).EQ.1) SFBOD=5.0*TOTBOD(KK)/1000. SFQU356
IF(KLAND(KK).EQ.2) SFBOD=3.6*TOTBOD(KK)/1000. SFQU357
IF(KLAND(KK).EQ.3) SFBOD=7.7*TOTBOD(KK)/1000. SFQU358
```

```

      IF(KLAND(KK).EQ.4) SF80D=3.0*TOTDD(KK)/1000. SFQU359
      IF(KLAND(KK).EQ.5) SF80D=5.0*TOTDD(KK)/1000. SFQU360
      BODNS=TCTDD(KK)*0.05 SFQU361
      POCB=POCCB*CBDEN*ASUB(KK) SFQU362
      SUMDD=SUMDC+TOTDD(KK) SFQU363
      SUMBOD=SUMBOD+BCDNS+SF80D SFQU364
      SUMCB=SUMCB+POCB SFQU365
341 CONTINUE SFQU366
      IF(KOUNT.NE.KTNUM) KOUNT=KK+1 SFQU367
      IF(KK.EQ.KTNUM) GO TO 334 SFQU368
      IF(INPUT(KK+1).EQ.INPUT(KK)) WRITE(6,631)INPUT(KK),KLAND(KK),ASUB(SFQU369
      *KK),GUTTER(KK),TOTDD(KK),SF80D,BODNS,POCB SFQU370
631 FORMAT(' ',I5,I7,T17,F8.2,T25,I11.2,T39,F11.2,T52,F8.2,T62,F8.2,T7SFQU371
      *2,F8.2) SFQU372
      IF(INPLT(KK+1).NE.KNUM(J)) GO TO 340 SFQU373
334 CONTINUE SFQU374
340 CONTINUE SFQU375
      LL=KOUNT SFQU376
      IF(NPRINT.EQ.0) GO TO 660 SFQU377
      WRITE(06,610) SFQU378
610 FORMAT('0',I9X,'SUSPENDED SOLIDS',6X,'FIVE-DAY BIOCHEMICAL OXYGEN SFQU379
      *DEMAND',5X,'CELLIFORM CONC.'/' ',I1X,'TIME',4X,'RUNCFS',3X,'(POPSS)' SFQU380
      *,2X,'(CONCSS)',3X,'(CBINC)',I1X,'+',3X,'(POP)',I1X,'= (CBASTM) (C' SFQU381
      *ONBOD)',8X,'(CCOLI)'/' ',I2X,'CFS',4X,'LBS/DT',6X,'MG/L',4X,'LBS/DS' SFQU382
      *T',4X,'LBS/DT',4X,'LBS/DT',6X,'MG/L',7X,'MPN/100ML') SFQU383
      IHOURL=KHCUR SFQU384
      IMIN=KMIN SFQU385
      REWIND IFPRNT SFQU386
      IF(KSKIP.EQ.0) GO TO 557 SFQU387
      DO 556 M=1,KSKIP SFQU388
      READ(IFPRNT) SKIP1,SKIP2,SKIP3,SKIP4,SKIP5,SKIP6,SKIP7 SFQU389
556 CONTINUE SFQU390
557 CONTINUE SFQU391
      ISKIP=NINLTS-1 SFQU392
      NUSTEP=NTSTEP/NPRINT SFQU393
      DO 555 KTSTEP=1,NUSTEP SFQU394
      READ(IFPRNT)RUNCFS(NEW,J),AVGFLO(J),POPSS(J),CBINC(J),POP(J),CBAST SFQU395
      *M(J),CCOLI(J) SFQU396
      IF(KTSTEP.EQ.NUSTEP) GO TO 554 SFQU397
      DO 553 N=1,ISKIP SFQU398
      READ(IFPRNT) SKIP1,SKIP2,SKIP3,SKIP4,SKIP5,SKIP6,SKIP7 SFQU399
553 CONTINUE SFQU400
554 CONTINUE SFQU401
      IF(KTSTEP.EQ.1) GO TO 303 SFQU402
      IMIN=IMIN+(TMIN*NPRINT) SFQU403
      IF(IMIN.GE.60) IHCUR=IHOURL+1 SFQU404
      IF(IMIN.GE.60) IMIN=IMIN-60 SFQU405
      IF(IHOURL.GE.25) IHOURL=1 SFQU406
303 IF(AVGFLC(J).GT.0.0) GO TO 301 SFQU407
      CONBOD=0.0 SFQU408
      CONCSS=C.C SFQU409
      GO TO 302 SFQU410
301 CONTINUE SFQU411
      CONBOD=CONVER*CBASTM(J)/AVGFLO(J) SFQU412
      CONCSS=CONVER*POPSS(J)/AVGFLO(J) SFQU413
302 CONTINUE SFQU414
      WRITE(06,208) IHOURL,IMIN,RUNCFS(NEW,J),POPSS(J),CONCSS,CBINC(J),POS SFQU415
      *P(J),CBASTM(J),CONBOD,CCOLI(J) SFQU416
208 FORMAT(' ',I2,IH:',I2,7F10.2,1PE10.2) SFQU417
      IF(KTSTEP.NE.NUSTEP) GO TO 555 SFQU418

```

```

WRITE (6,611) TPOPSS(J),TCBINC(J),TPOPI(J),TCBAST(J)
611 FORMAT('0',' PCUNDS REMOVED',F10.2,10X,3F10.2)
555 CONTINUE
KSKIP=KSKIP+1
GO TO 665
660 PCNTSS=TPOPSS(J)/SUMDD*100.0
PCTBOD=TPCP(J)/SUMBOD*100.0
PCNTCB=TCBINC(J)/SUMCB*100.0
TPCBOD=TCBAST(J)/(SUMCB+SUMBOD)*100.0
CCOLI(J)=TCCOLI(J)/IKCUNT(J)
C SUM THE REMOVALS FOR ALL AREAS
TTPSS=TTPSS+TPOPSS(J)
TTCBNC=TTCBNC+TCBINC(J)
TTPOP=TTPOP+TPOPI(J)
TTCBST=TCBAST(J)+TTCBST
C SUM THE CRIGINAL LOAD FOR ALL AREAS
TSUMDD=TSUMDD+SUMDD
TSMBD=TSMBD+SUMBOD
TSUMCB=TSUMCB+SUMCB
TBOD=TBOD+SUMCB+SUMBOD
WRITE (6,612) INPUT(KK),KLAND(KK),ASUB(KK),GUTTER(KK),TOTD(KK),SFBOSFQJ439
*,BODNS,FCCB,TPCPS(J),TCBINC(J),TPOPI(J),TCBAST(J),CCOLI(J),PCNTSSSFQJ440
*,PCNTCB,PCTBOD,TPCBOD
612 FORMAT(' ',I5,I7,T17,F8.2,T25,F11.2,T39,F11.2,T52,F8.2,T62,F8.2,T7SFQJ442
*,F8.2,T82,F9.2,T92,F8.2,T102,F8.2,T112,F8.2,T122,1PE9.2/' ',T82, SFQJ443
*,OPF8.1,' %',F7.1,' % ',F7.1,' % ',F7.1,' % '/')
665 CONTINUE
666 CONTINUE
777 CONTINUE
IF(NPRINT.GT.0) GO TO 888
C FIND THE TCTAL REMOVALS FOR ALL AREAS, IN PERCENT
TPCTSS=TTPSS/TSUMDD*100.0
TPCTBOD=TTPCB/TSMBD*100.0
TPCTCB=TTCBNC/TSUMCB*100.0
TPTBOD=TTCBST/TBOD*100.0
NEXDAY = 0
ENDTIM=(FLCAT(KHOUR)+(FLOAT(KMIN))/60.)+(DT*FLOAT(NTSTEP))/60.
IF(ENDTIM.GE.25) NEXDAY=NEXDAY+1
IF(ENDTIM.GE.25) ENDTIM=ENDTIM-24
IF(ENDTIM.GE.25) NEXDAY=NEXDAY+1
IF(ENDTIM.GE.25) ENDTIM=ENDTIM-24
WRITE (6,616) NTSTEP,TTPSS,TTCBNC,TTPOP,TTCBST,ENDTIM,NEXDAY,TPCTSSSFQJ460
*,TPCTCB,TPCTBOD,TPTBOD
616 FORMAT(' ',-----SFQJ462
*,-----SFQJ463
*,-----/'0',5X,'TOTAL TRANSFERRED FROM ALL INLET AREAS DURINGSFQJ464
*G ',I3,' TIMESTEPS',T82,F9.2,T92,F8.2,T102,F8.2,T112,F8.2,/' ',5X,SFQJ465
*'ENDING AT ',F4.1,' HRS., ',I1,' DAY(S) LATER',T82,F8.1,' %',F7.1,SFQJ466
*'% ',F7.1,' % ',F7.1,' % ',//)
888 CONTINUE
WRITE (6,211)
211 FORMAT(1H1)
RETURN
END

```



# Section 3

## TRANSPORT BLOCK

|                   | <u>Page</u> |
|-------------------|-------------|
| Subroutine TRANS  | 43          |
| Function DEPTH    | 54          |
| Function DPSI     | 56          |
| Subroutine DWLOAD | 58          |
| Subroutine FILTH  | 61          |
| Subroutine FINDA  | 67          |
| Subroutine FIRST  | 68          |
| Subroutine INFIL  | 71          |
| Subroutine INITAL | 74          |
| Subroutine NEWTON | 76          |
| Subroutine PRINT  | 77          |
| Function PSI      | 80          |
| Subroutine QUAL   | 82          |
| Function RADH     | 85          |
| Subroutine ROUTE  | 87          |
| Subroutine SLOP   | 93          |
| Function VEL      | 95          |
| Subroutine TSTRDT | 96          |
| Subroutine TSTORG | 104         |
| Subroutine TSTCST | 110         |
| Subroutine TPLUGS | 112         |
| Subroutine TSROUT | 116         |
| Subroutine TINTRP | 118         |
| Function ACOS     | 119         |
| BLOCK DATA        | 120         |

|        |                                                                        |          |
|--------|------------------------------------------------------------------------|----------|
|        | SUBROUTINE TRANS                                                       | TRAN 1   |
| C      |                                                                        | TRAN 2   |
| C***** | UNIVERSITY OF FLORIDA TRANSPORT MODEL*****                             | TRAN 3   |
| C      |                                                                        | TRAN 4   |
| C***** | SPECIFICATION STATEMENTS                                               | TRAN 5   |
| C      |                                                                        | TRAN 6   |
|        | DIMENSION QI(160),QO(160),SURGE1(160),SURGE2(160),WELL1(160),          | TRAN 7   |
| 1      | WELL2(160),PUMF(160),G(400),QO1(160),QO2(160),OUTIN(10,5),             | TRAN 8   |
| 2      | OUT(10,5),OUTTAP(5,5)                                                  | TRAN 9   |
|        | COMMON /TAPES/ INCNT, IOUTCT, JIN(10), JOUT(10), NSCRAT(5)             | TRAN 10  |
|        | COMMON/DRWF/ CVDWF(7), DVBD(7), DVSS(7), HVDWF(24), HVBD(24),          | TRAN 11  |
| 1      | HVSS(24), HVCOLI(24), KTNUM, KDAY, K HOUR, KMIN                        | TRAN 12  |
|        | COMMON/TABLES/ KDEPTH(25), KCLASS(25), PSIMAX(15), ALFMAX(15),         | TRAN 13  |
| 1      | NN(25), MM(25), ANORM(15,51), QNORM(15,51),                            | TRAN 14  |
| 2      | DNORM(15,51), AFACT(15), RFACT(15)                                     | TRAN 15  |
|        | COMMON/NAMES/ NAME(4,25), GND, YES, BLANK                              | TRAN 16  |
|        | COMMON A(160,2,2), Q(160,2,2), CPOLL(160,2,2,3), QMAX(160),            | TRAN 17  |
| 1      | CFULL(160), AFULL(160), DXDT(160), C1(160), SLOPE(160),                | TRAN 18  |
| 2      | DIST(160), GEOM1(160), ROUGH(160), NUE(160), NUE(160,3),               | TRAN 19  |
| 3      | INUE(160,3), NTYPE(160), JR(160), NCLASS, NE, NDT, EPSIL,              | TRAN 20  |
| 4      | TIME, DT, M, KFULL, N, NOS, NPOLL, NPRINT, ITER,                       | TRAN 21  |
| 5      | CDWF(160), IOLD(160), P1(160), RNOFF(160), QINFIL(160),                | TRAN 22  |
| 6      | WCWF(160,3), PLUTO(160,3), IR(160), P2(160), NIN(1000),                | TRAN 23  |
| 7      | P5(160), P6(160), P7(160), SCF(160), BARREL(160),                      | TRAN 24  |
| 8      | TITLE(40), NPE(20), NYN(20), NORDER(70), GEOM2(160),                   | TRAN 25  |
| 9      | GECH3(160), P4(160), SCOUR(160), KSTORE(160)                           | TRAN 26  |
|        | COMMON BCDIN( 2,150), SSIN( 2,150), BODOUT, SSOUT, COLIN(2,150),       | TRAN 27X |
| *      | QINST, CCLST, CINSTL( 2), QOUSTL( 2), STORL( 2), QOUTO( 2), STORO( 2), | TRAN 28  |
| *      | NSTOR, KSTCR, IPRINT( 2), IPOL( 2), IFLOOD( 2), ICOST( 2), DEPMAX( 2), | TRAN 29  |
| *      | ATERM( 2,11), AQ2DT2( 2,11), BDEPTH( 2,11), BSTOR( 2,11), COLOUT,      | TRAN 30X |
| *      | DUMSTR(11), DUMDEP(11),                                                | TRAN 31  |
| *      | KTSTEP, VOLIN( 2,150), VOLOUT( 2,150), STOR, CUMIN( 2), CUMOUT( 2),    | TRAN 32  |
| *      | SBOD( 2), SSS( 2), SCOL( 2),                                           | TRAN 33X |
| *      | ISTMOD( 2), ISTTYP( 2), ISTOUT( 2),                                    | TRAN 34  |
| *      | QPUMP( 2), DSTART( 2), DSTOP( 2),                                      | TRAN 35  |
| *      | DTON( 2), STORMX( 2), DTPUMP( 2), DTMORE( 2), STORF( 2), APLAN( 2),    | TRAN 36  |
| *      | CLAND( 2), CSTOR( 2), CPS( 2), CTOTAL( 2), CPCUYD( 2), CPACRE( 2),     | TRAN 37  |
| *      | LP, JP, LPREV( 2), LABEL, DETENT(150), FRAC(150), OUT1(10,20)          | TRAN 38X |
|        | EQUIVALENCE (QO(1), Q(1,2,2)), (QI(1), Q(1,1,2))                       | TRAN 39  |
|        | EQUIVALENCE (PUMP(1), DIST(1)), (SURGE1(1), P1(1)), (SURGE2(1), P2(1)) | TRAN 40  |
|        | EQUIVALENCE (QO1(1), QMAX(1)), (QO2(1), QFULL(1))                      | TRAN 41  |
|        | EQUIVALENCE (WELL1(1), SLOPE(1)), (WELL2(1), ROUGH(1))                 | TRAN 42  |
|        | DIMENSION JN(5)                                                        | TRAN 43  |
|        | DIMENSION FLODEP(150)                                                  | KINGTEMP |
| C      |                                                                        | TRAN 44  |
| C      |                                                                        | TRAN 45  |
| C***** | READ DATA DESCRIBING USER SUPPLIED CONDUITS, IF ANY.                   | TRAN 46  |
| C      |                                                                        | TRAN 47  |
|        | READ(5,901) NCLASS, KPRINT                                             | TRAN 48  |
|        | NKLAS=NCLASS +12                                                       | TRAN 49  |
|        | IF (NKLASS.LE.0) GO TO 41                                              | TRAN 50  |
|        | READ (5,903) ((NAME(J,I), J=1,4), I=13, NKLAS)                         | TRAN 51  |
|        | READ (5,901) (NN(I), I=13, NKLAS)                                      | TRAN 52  |
|        | READ (5,901) (MM(I), I=13, NKLAS)                                      | TRAN 53  |
|        | READ (5,900) (ALFMAX(I), I=13, NKLAS)                                  | TRAN 54  |
|        | READ (5,900) (PSIMAX(I), I=13, NKLAS)                                  | TRAN 55  |
|        | READ (5,900) (AFACT(I), I=13, NKLAS)                                   | TRAN 56  |
|        | READ (5,900) (RFACT(I), I=13, NKLAS)                                   | TRAN 57  |
| C      | READ PARAMETERS FOR DEPTH-AREA RELATIONSHIPS.                          | TRAN 58  |
|        | DO 18 I=13, NKLAS                                                      | TRAN 59  |
|        | KOEPH(I) = 2                                                           |          |

|                                                                      |         |
|----------------------------------------------------------------------|---------|
| NNN = NN(I)                                                          | TRAN 60 |
| READ (5,900) (DNORM(I,J),J=1,NNN)                                    | TRAN 61 |
| NNN = NNN+1                                                          | TRAN 62 |
| IF (NNN.GT.51) GO TO 18                                              | TRAN 63 |
| DO 17 J=NNN,51                                                       | TRAN 64 |
| 17 DNORM(I,J)=0.0                                                    | TRAN 65 |
| 18 CONTINUE                                                          | TRAN 66 |
| C READ PARAMETERS AND PERFORM INITIALIZING CALCULATIONS ON Q-A CURVE | TRAN 67 |
| DO 40 I=13,NKLAS                                                     | TRAN 68 |
| KLASS(I) = 2                                                         | TRAN 69 |
| C CONDUIT WITH TABULAR C-A RELATIONSHIP                              | TRAN 70 |
| MMM = MM(I)                                                          | TRAN 71 |
| DALPHA=1.0/FLCAT(MMM-1)                                              | TRAN 72 |
| READ(5,900)(QNORM(I,J),J=1,MMM)                                      | TRAN 73 |
| ANORM(I,1)=0.0                                                       | TRAN 74 |
| DO 33 J=2,MMM                                                        | TRAN 75 |
| 33 ANORM(I,J)=ANORM(I,J-1)+DALPHA                                    | TRAN 76 |
| MMM = MMM+1                                                          | TRAN 77 |
| IF (MMM.GT.51) GO TO 40                                              | TRAN 78 |
| DO 39 J=MMM,51                                                       | TRAN 79 |
| ANORM(I,J) = 0.0                                                     | TRAN 80 |
| 39 QNORM(I,J)=0.0                                                    | TRAN 81 |
| 40 CONTINUE                                                          | TRAN 82 |
| C                                                                    | TRAN 83 |
| C*****WRITE DATA DESCRIBING DIFFERENT TYPES OF SEWER ELEMENTS(IF     | TRAN 84 |
| C DESIRED)                                                           | TRAN 85 |
| 41 CONTINUE                                                          | TRAN 86 |
| IF(KPRINT.NE.1) GO TO 45                                             | TRAN 87 |
| WRITE (6,970)                                                        | TRAN 88 |
| DO 43 I=1,15                                                         | TRAN 89 |
| MMM = MM(I)                                                          | TRAN 90 |
| IF (MM(I).LT.NN(I)) MMM = NN(I)                                      | TRAN 91 |
| 43 WRITE (6,971) I,(NAME(J,I),J=1,4),ALFMAX(I),PSIMAX(I),AFAC(I),    | TRAN 92 |
| 1 RFACT(I),KDEPTH(I),KLASS(I),(J,ANORM(I,J),QNORM(I,J),DNORM(I,J),   | TRAN 93 |
| 2 J=1,MMM)                                                           | TRAN 94 |
| WRITE(6,972)                                                         | TRAN 95 |
| WRITE(6,973) (I,KDEPTH(I),KLASS(I),(NAME(K,I),K=1,4),I=16,25)        | TRAN 96 |
| C                                                                    | TRAN 97 |
| C*****READ IN DATA FOR EACH SEWER SYSTEM                             | TRAN 98 |
| C                                                                    | TRAN 99 |
| C READ IN TITLE FOR DATA.                                            | TRAN100 |
| C                                                                    | TRAN101 |
| 45 READ(5,903) (TITLE(I),I=1,20)                                     | TRAN102 |
| C                                                                    | TRAN103 |
| C READ IN EXECUTION DATA.                                            | TRAN104 |
| C                                                                    | TRAN105 |
| READ (5,901) NE,NDT,NINPUT,NNYN,NNPE,NCUTS,NPRINT,NPOLL,NITER        | TRAN106 |
| C                                                                    | TRAN107 |
| C NE = TOTAL NUMBER OF ELEMENTS                                      | TRAN108 |
| C NDT = NUMBER OF TIME STEPS                                         | TRAN109 |
| C NINPUT = NUMBER OF INPUT ELEMENTS                                  | TRAN110 |
| C NNYN = NUMBER OF ELEMENTS FOR WHICH INFLOW                         | TRAN111 |
| C PRINTOUT IS DESIRED                                                | TRAN112 |
| C NNPE = NUMBER OF ELEMENTS FOR WHICH OUTFLOW                        | TRAN113 |
| C PRINTOUT IS DESIRED                                                | TRAN114 |
| C NCUTS = NUMBER OF ELEMENTS FOR WHICH OUTFLOW                       | TRAN115 |
| C HYDROGRAPHS AND POLLUTOGRAPHS ARE TO BE                            | TRAN116 |
| C PROVIDED ON TAPE FOR INTERFACING                                   | TRAN117 |
| C NPRINT = 0 NO TRACING MESSAGES GENERATED                           | TRAN118 |
| C 1 TRACING MESSAGES ARE GENERATED IN ROUTE                          | TRAN119 |

|   |                                                                          |          |
|---|--------------------------------------------------------------------------|----------|
| C | 2 TRACING MESSAGES ARE GENERATED IN ROUTE AND                            | TRAN120  |
| C | TRANS                                                                    | TRAN121  |
| C | NPOLL = NUMBER OF POLLUTANTS TO BE ROUTED                                | TRAN122  |
| C | NITER = NUMBER OF ITERATIONS IN ROUTING SCHEME                           | TRAN123  |
| C |                                                                          | TRAN124  |
|   | READ(5,900)DT, EPSIL, DWDCAYS                                            | TRAN125  |
| C |                                                                          | TRAN126  |
| C | DT = LENGTH OF TIME STEP IN SECONDS                                      | TRAN127  |
| C | EPSIL = CONVERGENCE CRITERIA IN ROUTING                                  | TRAN128  |
| C | DWDCAYS = NUMBER OF DRY WEATHER DAYS PRIOR TO STORM                      | TRAN129  |
| C |                                                                          | TRAN130  |
| C | ASSIGN DEFAULT VALUES                                                    | TRAN131  |
|   | IF(NITER.LE.0) NITER=4                                                   | TRAN132  |
|   | IF (EPSIL.LE.0.0) EPSIL = 0.0001                                         | TRAN133  |
| C |                                                                          | TRAN134  |
|   | READ(5,901) NCNTRL, NINFIL, NFILTH, JPRINT, JPLOT                        | TRAN135  |
| C |                                                                          | TRAN136  |
| C | NCNTRL = 1 IF INPUT INFORMATION IS ON TAPE                               | TRAN137  |
| C | 0 IF INPUT INFORMATION IS NOT ON TAPE                                    | TRAN138  |
| C | NINFIL = 1 IF INFILTRATION ROUTINE IS TO BE CALLED                       | TRAN139  |
| C | 0 IF INFILTRATION ROUTINE IS NOT TO BE CALLED                            | TRAN140  |
| C | NFILTH = 1 IF DRY WEATHER FLOW ROUTINE IS TO BE CALLED                   | TRAN141  |
| C | 0 IF DRY WEATHER FLOW ROUTINE IS NOT TO BE CALLED                        | TRAN142  |
| C | JPRINT = 1 IF PRINTING ROUTINE IS TO BE CALLED                           | TRAN143  |
| C | 0 IF PRINTING ROUTINE IS NOT TO BE CALLED                                | TRAN144  |
| C | JPLOT = 1 IF PLOTTING ROUTINE IS TO BE CALLED FROM WITHIN                | TRAN145  |
| C | 0 IF PLOTTING ROUTINE NOT CALLED FROM WITHIN                             | TRAN146  |
| C |                                                                          | TRAN147  |
| C | READ IN ELEMENT DATA AND INITIALIZE VARIABLES.                           | TRAN148  |
| C |                                                                          | TRAN149  |
|   | I = 1                                                                    | TRAN150  |
|   | KSTOR = 0                                                                | TRAN151  |
|   | NSTOR = 0                                                                | TRAN152  |
|   | ICOST(1)=0                                                               | TRAN152A |
|   | ICOST(2)=0                                                               | TRAN152B |
|   | DO 47 K=1, NE                                                            | TRAN153  |
|   | READ(5,902)NOE(K), (NUE(K,J), J=1,3), NTYPE(K), DIST(K), GEOM1(K), SLOPE | TRAN154  |
|   | 1(K), ROUGH(K), GEOM2(K), BARREL(K), GEOM3(K)                            | TRAN155  |
| C |                                                                          | TRAN156  |
| C | ASSIGN DEFAULT VALUES                                                    | TRAN157  |
| C |                                                                          | TRAN158  |
|   | IF(BARREL(K).LT.1.0) BARREL(K)=1.0                                       | TRAN159  |
|   | IF(GEOM1(K).LE.0.0) GEOM1(K)=0.0                                         | TRAN160  |
|   | IF (NTYPE(K).LE.15.AND.SLOPE(K).LE.0.0) SLOPE(K)=0.1                     | TRAN161X |
|   | IF(ROUGH(K).LE.0.0.AND.NTYPE(K).LE.15) ROUGH(K)=0.0130                   | TRAN162  |
| C | ASSUME MANHOLE IF TYPE NOT SPECIFIED                                     | TRAN163X |
|   | IF(NTYPE(K).LE.0) NTYPE(K)=16                                            | TRAN164X |
| C |                                                                          | TRAN165  |
|   | KSTCRE(K) = 0                                                            | TRAN166  |
|   | IF(NTYPE(K) .NE. 19) GO TO 54                                            | TRAN167  |
|   | KSTOR = KSTOR + 1                                                        | TRAN168  |
|   | KSTCRE(K) = KSTOR                                                        | TRAN169  |
|   | NSTOR = KSTOR                                                            | TRAN170  |
|   | 54 CONTINUE                                                              | TRAN171  |
|   | QFULL(K) = 0.0                                                           | TRAN172  |
|   | QMAX(K)=0.0                                                              | TRAN173  |
|   | AFULL(K)=0.0                                                             | TRAN174  |
|   | NNEED = NCE(K)                                                           | TRAN175  |
|   | 47 NIN(NNEED) = K                                                        | TRAN176  |
| C |                                                                          | TRAN177  |

|                                                                         |         |
|-------------------------------------------------------------------------|---------|
| C*****READ DATA FOR STORAGE ELEMENTS                                    | TRAN178 |
| C                                                                       | TRAN179 |
| IF (NSTOR .GT. 2) GO TO 9000                                            | TRAN180 |
| IF (NSTOR .GT. 0) CALL TSTRDT                                           | TRAN181 |
| C                                                                       | TRAN182 |
| C*****SEQUENCE ELEMENT DATA                                             | TRAN183 |
| C                                                                       | TRAN184 |
| WRITE(6,903) (TITLE(I),I=1,20)                                          | TRAN185 |
| CALL SLGP                                                               | TRAN186 |
| C                                                                       | TRAN187 |
| C*****CALCULATE CONSTANTS AND FLOW PARAMETERS FOR EACH ELEMENT          | TRAN188 |
| C                                                                       | TRAN189 |
| CALL FIRST                                                              | TRAN190 |
| C                                                                       | TRAN191 |
| C*****WRITE DESCRIPTION OF RUN AND SEWER SYSTEM                         | TRAN192 |
| C                                                                       | TRAN193 |
| WRITE (6,903) (TITLE(I),I=1,20)                                         | TRAN194 |
| WRITE (6,915) NE, NDT, DT                                               | TRAN195 |
| WRITE (6,916)                                                           | TRAN196 |
| DO 48 I=1,NE                                                            | TRAN197 |
| NT = NTYPE(I)                                                           | TRAN198 |
| 48 WRITE (6,920) NOE(I), NT, (NAME(J, NT), J=1,4), SLOPE(I), DIST(I),   | TRAN199 |
| TROUGH(I), GEOM1(I), GEOM2(I), GEOM3(I), BARREL(I), AFULL(I), QFULL(I), | TRAN200 |
| ZQMAX(I), SCF(I)                                                        | TRAN201 |
| WRITE(6,913) EPSIL, NITER                                               | TRAN202 |
| C                                                                       | TRAN203 |
| C*****INITIALIZATION                                                    | TRAN204 |
| C                                                                       | TRAN205 |
| NEE=NE+1                                                                | TRAN206 |
| DO 50 K=1,NEE                                                           | TRAN207 |
| IOLD(K)=1                                                               | TRAN208 |
| QINFIL(K)=0.0                                                           | TRAN209 |
| QDWF(K)=0.0                                                             | TRAN210 |
| SCOUR(K)=0.0                                                            | TRAN211 |
| IR(K)=NINPUT+1                                                          | TRAN212 |
| DO 50 L=1,NPOLL                                                         | TRAN213 |
| WDWF(K,L)=0.0                                                           | TRAN214 |
| 49 PLUTC(K,L)=0.0                                                       | TRAN215 |
| DO 50 I=1,2                                                             | TRAN216 |
| DO 50 II=1,2                                                            | TRAN217 |
| CPOLL(K,I,II,L)=0.0                                                     | TRAN218 |
| A(K,II,I)=0.0                                                           | TRAN219 |
| 50 Q(K,II,I)=0.0                                                        | TRAN220 |
| CO(NE+1)=0.0                                                            | TRAN221 |
| QI(NE+1)=0.0                                                            | TRAN222 |
| NTYPE(NE+1)=16                                                          | TRAN223 |
| NPOLS=NPOLL+1                                                           | TRAN224 |
| KDAY=1                                                                  | TRAN225 |
| KHOOR=1                                                                 | TRAN226 |
| KMIN=0                                                                  | TRAN227 |
| TIME=0.0                                                                | TRAN228 |
| RNUFF(NINPUT+1)=0.0                                                     | TRAN229 |
| DO 51 I=1,NPOLL                                                         | TRAN230 |
| PLUTO(NINPUT+1,I)=0.0                                                   | TRAN231 |
| 51 CPOLL(NE+1,2,2,I)=0.0                                                | TRAN232 |
| DO 52 I=1,7                                                             | TRAN233 |
| DVDWF(I)=1.0                                                            | TRAN234 |
| CVBOD(I)=1.0                                                            | TRAN235 |
| 52 DVSS(I)=1.0                                                          | TRAN236 |
| DO 53 I=1,24                                                            | TRAN237 |

|                                                                       |          |
|-----------------------------------------------------------------------|----------|
| HVDWF(I)=1.0                                                          | TRAN238  |
| HVBUD(I)=1.0                                                          | TRAN239  |
| 53 HVSS(I)=1.0                                                        | TRAN240  |
| C                                                                     | TRAN241  |
| C*****INTERFACING MECHANISM FOR QUANTITY AND QUALITY OF RUNOFF        | TRAN242  |
| C                                                                     | TRAN243  |
| INCNT=INCNT+1                                                         | TRAN244  |
| IOUTCT=IOUTCT+1                                                       | TRAN245  |
| NTRIN=JIN(INCNT)                                                      | TRAN246  |
| NTROUT=JOUT(IOUTCT)                                                   | TRAN247  |
| C BYPASS TAPE OPERATIONS IF DESIRED                                   | TRAN248  |
| IF(INCNTRL.NE.1) GO TO 59                                             | TRAN249  |
| C                                                                     | TRAN250  |
| C*****READ LEADING INFORMATION ON INPUT TAPE                          | TRAN251  |
| C                                                                     | TRAN252  |
| REWIND ATRIN                                                          | TRAN253  |
| READ(NTRIN)(TITLE(I),I=1,40)                                          | TRAN254  |
| READ(NTRIN) NDUM1,NINPUT,NPOLL,NDUM2,TZERO,TAREA                      | TRAN255X |
| C                                                                     | TRAN256  |
| C READ ELEMENT NOS. AT WHICH HYDROGRAPHS AND POLLUTOGRAPHS ARE        | TRAN257  |
| C ENTERED. THIS MUST ALSO BE THE ORDER IN WHICH HYDROGRAPH AND        | TRAN258  |
| C POLLUTOGRAPH ORDINATES APPEAR AT EACH TIME STEP.                    | TRAN259  |
| C                                                                     | TRAN260  |
| READ(NTRIN) (NORDER(I),I=1,NINPUT)                                    | TRAN261  |
| C                                                                     | TRAN262  |
| C*****WRITE LEADING INFORMATION ON OUTPUT TAPE                        | TRAN263  |
| C                                                                     | TRAN264  |
| 59 CONTINUE                                                           | TRAN265  |
| REWIND NTRCUT                                                         | TRAN265A |
| WRITE(NTRCUT)(TITLE(I),I=1,40)                                        | TRAN266  |
| WRITE(NTRCUT) NDT,NOUTS,NPOLL,DT,TZERO,TAREA                          | TRAN267X |
| C                                                                     | TRAN268  |
| C*****READ ELEMENT NUMBERS FOR WHICH OUTPUT IS TO BE PROVIDED ON TAPE | TRAN269  |
| C                                                                     | TRAN270  |
| READ(5,9C1)(JN(I),I=1,NOUTS)                                          | TRAN271  |
| WRITE(6,9I7) (JN(I),I=1,NOUTS)                                        | TRAN272  |
| WRITE(NTRCUT)(JN(N),N=1,NOUTS)                                        | TRAN273  |
| C                                                                     | TRAN274  |
| DO 60 I=1,NINPUT                                                      | TRAN275  |
| NNEED=NORDER(I)                                                       | TRAN276  |
| NNEED=MIN(NNEED)                                                      | TRAN277  |
| 60 IR(NNEED)= I                                                       | TRAN278  |
| C                                                                     | TRAN279  |
| C*****PERFORM SCRATCH TAPE OPERATIONS                                 | TRAN280  |
| C                                                                     | TRAN281  |
| C READ IN ELEMENT NUMBERS FOR WHICH INPUT POLLUTOGRAPHS AND           | TRAN282  |
| C HYDROGRAPHS TO BE STORED AT ALL TIME STEPS.                         | TRAN283  |
| C                                                                     | TRAN284  |
| READ(5,9C1)(NYN(I),I=1,NNYN)                                          | TRAN285  |
| C                                                                     | TRAN286  |
| C READ IN ELEMENT NUMBERS FOR WHICH OUTPUT POLLUTOGRAPHS AND          | TRAN287  |
| C HYDROGRAPHS TO BE STORED AT ALL TIME STEPS.                         | TRAN288  |
| C                                                                     | TRAN289  |
| READ(5,9C1)(NPE(I),I=1,NNPE)                                          | TRAN290  |
| C                                                                     | TRAN291  |
| C*****STORE PRINT INFORMATION ON SCRATCH TAPES                        | TRAN292  |
| C                                                                     | TRAN293  |
| DO 62 III=1,2                                                         | TRAN294  |
| NTX=NSCRAT(III)                                                       | TRAN295  |
| REWIND NTX                                                            | TRAN296  |

|                                                                      |         |
|----------------------------------------------------------------------|---------|
| IF(III.GT.1) GO TO 61                                                | TRAN297 |
| WRITE(NTX) NDT,NNYN,NPCLL,NNPE,DT,JPRINT,JPLOT                       | TRAN298 |
| WRITE(NTX) (NYN(I),I=1,NNYN)                                         | TRAN299 |
| GO TO 62                                                             | TRAN300 |
| 61 WRITE(NTX) NDT,ANPE,NPCLL,NNPE,DT,JPRINT,JPLOT                    | TRAN301 |
| WRITE(NTX) (NPE(I),I=1,NNPE)                                         | TRAN302 |
| 62 CONTINUE                                                          | TRAN303 |
| C                                                                    | TRAN304 |
| C*****DETERMINE AVERAGE DAILY DWF AND INFILTRATION                   | TRAN305 |
| C                                                                    | TRAN306 |
| C DATA IS READ FROM BOTH OF THESE SUBROUTINES                        | TRAN307 |
| C IF(NINFIL.EQ.1) CALL INFIL                                         | TRAN308 |
| C                                                                    | TRAN309 |
| C IF(NFILTH.EQ.1) CALL FILTH                                         | TRAN310 |
| C                                                                    | TRAN311 |
| C*****INITIALIZE FLOWS, AREAS, CONCENTRATIONS AND DEPOSITION         | TRAN312 |
| C                                                                    | TRAN313 |
| C N = 0                                                              | TRAN314 |
| C CALL DKLOAE(DWDAYS)                                                | TRAN315 |
| C                                                                    | TRAN316 |
| C CALL INITAL                                                        | TRAN317 |
| C                                                                    | TRAN318 |
| C*****BEGIN MAIN LOOPS OF PROGRAM                                    | TRAN319 |
| C*****OUTER LOOP ON TIME, INNER LOOP ON ELEMENT NUMBER               | TRAN320 |
| C                                                                    | TRAN321 |
| C DO 200 N=1,NDT                                                     | TRAN322 |
| C UPDATE TIME OF DAY                                                 | TRAN323 |
| C TIME=TIME+DT                                                       | TRAN324 |
| C KMINS=KMINS+INT(DT)/60                                             | TRAN325 |
| C IF(KMINS.GT.60) K HOUR=K HOUR+1                                    | TRAN326 |
| C IF(KMINS.GT.60) KMINS=KMINS-60                                     | TRAN327 |
| C IF(KHOUR.GT.24) KDAY=KDAY+1                                        | TRAN328 |
| C IF(KHOUR.GT.24) KHOUR=1                                            | TRAN329 |
| C IF(KDAY.GT.7) KDAY=1                                               | TRAN330 |
| C                                                                    | TRAN331 |
| C*****READ INPUT TAPE AT EACH TIMESTEP                               | TRAN332 |
| C                                                                    | TRAN333 |
| C IF(NCNTRL.NE.1) GO TO 63                                           | TRAN334 |
| C READ INPLT HYDROGRAPH AND POLLUTOGRAPH ORDINATES AT EACH TIME STEP | TRAN335 |
| C                                                                    | TRAN336 |
| C READ(NTRIN) DTIN,(RNGFF(I),I=1,NINPUT),((PLUTO(I,J),J=1,NPOLL),    | TRAN337 |
| 1I=1,NINPUT)                                                         | TRAN338 |
| C                                                                    | TRAN339 |
| C 63 CONTINUE                                                        | TRAN340 |
| C                                                                    | TRAN341 |
| C*****CONVERT INFLOW POLLUTOGRAPHS TO LBS/SEC OR MPN/SEC             | TRAN342 |
| C                                                                    | TRAN343 |
| C DO 68 ND=1,NINPUT                                                  | TRAN344 |
| C DO 68 NJ=1,NPCLL                                                   | TRAN345 |
| C 68 PLUTO(ND,NJ)=PLUTO(ND,NJ)/60.                                   | TRAN346 |
| C                                                                    | TRAN347 |
| C*****BEGIN INNER LOOP ON ELEMENT NUMBER                             | TRAN348 |
| C                                                                    | TRAN349 |
| C DO 150 I=1,NE                                                      | TRAN350 |
| C M=CURRENT ELEMENT NUMBER,(INTERNAL NUMBER).                        | TRAN351 |
| C M=JRI(I)                                                           | TRAN352 |
| C                                                                    | TRAN353 |
| C STORE INPUT HYDROGRAPHS AND POLLUTOGRAPHS FOR DESIRED ELEMENTS     | TRAN354 |
| C                                                                    | TRAN355 |
| C DO 70 J=1,NNYN                                                     | TRAN356 |

|                                                                   |          |
|-------------------------------------------------------------------|----------|
| IF(NOE(M)-NYN(J)) 70,69,70                                        | TRAN357  |
| 69 NNEED=IR(M)                                                    | TRAN358  |
| OUTIN(J,1) = RNCFF(NNEED)                                         | TRAN359  |
| DO 81 JJ=2,NPOLS                                                  | TRAN360  |
| C CONVERT INLET POLLUTOGRAPH TO LBS/MIN OR MPN/MIN FOR PRINT ONLY | TRAN361  |
| 81 OUTIN(J,JJ)= PLUTO(NNEED,JJ-1) * 60.                           | TRAN362  |
| GO TO 71                                                          | TRAN363  |
| 70 CONTINUE                                                       | TRAN364  |
| C                                                                 | TRAN365  |
| 71 INPUT=M                                                        | TRAN366  |
| C                                                                 | TRAN367  |
| C*****CORRECT DWF FOR TEMPORAL VARIATIONS                         | TRAN368  |
| C                                                                 | TRAN369  |
| C CORRECT SEWAGE FOR DAILY VARIATION                              | TRAN370  |
| C WDW1(INPUT,1) IS B O D IN LBS/SEC                               | TRAN371  |
| C WDW2(INPUT,2) IS SS IN LBS/SEC                                  | TRAN372  |
| C WDW3(INPUT,3) IS COLIFORM IN MPN/SEC                            | TRAN373  |
| CDWF=QCWF(INPUT)*DVDWF(KDAY)                                      | TRAN374  |
| WDW1=WCWF(INPUT,1)*DVBOC(KDAY)*DVDWF(KDAY)                        | TRAN375  |
| WDW2=WCWF(INPUT,2)*DVSS(KDAY)*DVDWF(KDAY)                         | TRAN376  |
| WDW3=WDWF(INPUT,3)*1.0*DVDWF(KDAY)                                | TRAN377  |
| C CORRECT SEWAGE FOR HOURLY VARIATION                             | TRAN378  |
| DUMY1=CDWF*HVDWF(KHOUR)                                           | TRAN379  |
| DUMY2=WDW1*HVBOC(KHOUR)*HVDWF(KHOUR)                              | TRAN380  |
| DUMY3=WCWF2*HVSS(KHOUR)*HVDWF(KHOUR)                              | TRAN381  |
| DUMY5=WDWF3*HVCOLI(KHOUR)*HVDWF(KHOUR)                            | TRAN382  |
| C                                                                 | TRAN383  |
| C*****SUM UPSTREAM FLOWS                                          | TRAN384  |
| C                                                                 | TRAN385  |
| TOTAL=0.0                                                         | TRAN386  |
| DO 80 J=1,3                                                       | TRAN387  |
| NNEED=INUE(M,J)                                                   | TRAN388  |
| NTU = NTYPE(NNEED)                                                | TRAN389  |
| IF (NTU.LE.17.OR.NTU.GE.23) GO TO 79                              | TRAN390  |
| C HERE IF UPSTREAM ELEMENT IS OF FLOW DIVIDER TYPE.               | TRAN391  |
| L = GECM3(NNEED)                                                  | TRAN392  |
| QQ = QC2(NNEED)                                                   | TRAN393  |
| IF (NOE(M).EQ.L) QQ = QQ1(NNEED)                                  | TRAN394  |
| TOTAL = TOTAL+QQ                                                  | TRAN395  |
| GO TO 80                                                          | TRAN396  |
| 79 TOTAL = TOTAL+QQ(NNEED)                                        | TRAN397  |
| 80 CONTINUE                                                       | TRAN398  |
| C                                                                 | TRAN399  |
| C*****SURCHARGE ROUTINE                                           | TRAN400  |
| C                                                                 | TRAN401  |
| KFULL=2                                                           | TRAN402  |
| NT = NTYPE(M)                                                     | TRAN403  |
| K = KCLASS(NT)                                                    | TRAN404  |
| GO TO (90,90,110),K                                               | TRAN405  |
| C IF ELEMENT IS A CONDUIT,CHECK FOR SURCHARGING.                  | TRAN406  |
| 90 QI(M) = TOTAL/BARREL(M)                                        | TRAN407  |
| QI(M)=CI(M)+QINFIL(M)                                             | TRAN407A |
| IF(QI(M).LE.QMAX(M)) GO TO 95                                     | TRAN408  |
| C STORE SURCHARGE IN UPSTREAM ELEMENT.                            | TRAN409  |
| C SURCHARGING CONDUIT IS ASSUMED TO HAVE MANHOLE AT UPSTREAM END. | TRAN410  |
| C ASSUME ELEMENT WILL FLOW FULL AT UPSTREAM END.                  | TRAN411  |
| QFULL(M) = P1(M)*SQRT(SLOPE(M))                                   | TRAN412  |
| NNEED=INLF(M,1)                                                   | TRAN413  |
| SURGE2(NNEED) = (CI(M)-QFULL(M))*DT*BARREL(M)                     | TRAN414  |
| WRITE (6,960) TIME,N,NOE(M),SURGE2(NNEED),NUE(M,1)                | TRAN415  |



|                                                                     |          |
|---------------------------------------------------------------------|----------|
| QI(M)=CFULL(M)                                                      | TRAN416  |
| KFULL= 1                                                            | TRAN417  |
| ITER = 0                                                            | TRAN418  |
| GO TO 115                                                           | TRAN419  |
| C NO LONGER UPSTREAM SURCHARGE.                                     | TRAN420  |
| 95 NNEED=IAUE(M,1)                                                  | TRAN421  |
| SURGE2(NNEED)=0.0                                                   | TRAN422  |
| ITER=NITER                                                          | TRAN423  |
| GO TO 115                                                           | TRAN424  |
| C                                                                   | TRAN425  |
| C*****SUM INFLCWS TO NON-CONDUITS.                                  | TRAN426  |
| C                                                                   | TRAN427  |
| 110 NNEED= IR(M)                                                    | TRAN428  |
| QI(M)=TCTAL+RNCFF(NNEED)+SURGE2(M)/DT+DUMY1                         | TRAN429  |
| IF(QI(M).LT.0.0) QI(M)=0.0                                          | TRAN430  |
| ITER = 0                                                            | TRAN431  |
| C                                                                   | TRAN432  |
| C*****ROUTE FLOW THROUGH ELEMENT                                    | TRAN433  |
| C                                                                   | TRAN434  |
| 115 CALL ROUTE(NITER)                                               | TRAN435  |
| IF(NT.EQ.19) GO TO 116                                              | TRAN436  |
| ITER = ITER+1                                                       | TRAN437  |
| IF(ITER.GT.NITER) GO TO 115                                         | TRAN438  |
| C                                                                   | TRAN439  |
| C*****MULTIPLY FLCH BY NUMBER OF BARRELS                            | TRAN440  |
| C                                                                   | TRAN441  |
| 116 QI(M) = CI(M)*BARREL(M)                                         | TRAN442  |
| QO(M) = CC(M)*BARREL(M)                                             | TRAN443  |
| IF (NT.EQ.19) GO TO 6600                                            | TRAN444  |
| C IF(M.NE.115) GO TO 1010                                           | KINGTEMP |
| C RATIO=A(M,2,2)/AFULL(M)                                           | KINGTEMP |
| C FLODEP(N)=DEPTH(RATIO)*GECM1(M)                                   | KINGTEMP |
| C1010 CONTINUE                                                      | KINGTEMP |
| C                                                                   | TRAN445  |
| C*****ROUTE QUALITY PARAMETERS THROUGH ELEMENT                      | TRAN446  |
| C                                                                   | TRAN447  |
| CALL QUAL(DUMY2,DUMY3,0.0,DUMY5)                                    | TRAN448  |
| C                                                                   | TRAN449  |
| C*****STORE DESIRED OUTFLOWS TO BE GENERATED ONTO OUTPUT TAPE       | TRAN450  |
| C                                                                   | TRAN451  |
| 6600 DO 6666 J=1,NOUTS                                              | TRAN452  |
| IF(NOE(M)-JN(J)) 6666,6610,6666                                     | TRAN453  |
| 6610 OUTTAP(J,1)=QO(M)                                              | TRAN454  |
| DO 6620 JJ=2,NPOLS                                                  | TRAN455  |
| C CONVERT POLLUTOGRAPHS TO LBS/MIN OR MPN/MIN FOR INTERFACING       | TRAN456  |
| 6620 OUTTAP(J,JJ)=CPOLL(M,2,2,JJ-1) * QO(M)*60.                     | TRAN457  |
| GO TO 117                                                           | TRAN458  |
| 6666 CONTINUE                                                       | TRAN459  |
| C                                                                   | TRAN460  |
| C*****STORE DESIRED OUTFLOWS TO BE GENERATED ON SCRATCH TAPE        | TRAN461  |
| C FOR PRINTING PURPOSES                                             | TRAN462  |
| C                                                                   | TRAN463  |
| 117 DO 124 J=1,NNPE                                                 | TRAN464  |
| IF(NOE(M)-NPE(J))124,120,124                                        | TRAN465  |
| 120 OUT(J,1) = QO(M)                                                | TRAN466  |
| DO 118 JJ=2,NPOLS                                                   | TRAN467  |
| C CONVERT CUTFLOW POLLUTOGRAPH TO LBS/MIN OR MPN/MIN FOR PRINT ONLY | TRAN468  |
| 118 OUT(J,JJ) = CPOLL(M,2,2,JJ-1) * QO(M) * 60.                     | TRAN469  |
| GO TO 125                                                           | TRAN470  |
| 124 CONTINUE                                                        | TRAN471  |

|                                                                           |  |          |
|---------------------------------------------------------------------------|--|----------|
| C                                                                         |  | TRAN472  |
| C*****REPLACE VALUES AT OLD TIME STEP BY VALUES AT NEW ONE                |  | TRAN472  |
| C                                                                         |  | TRAN474  |
| 125 A(M,1,1)=A(M,1,2)                                                     |  | TRAN475  |
| A(M,2,1)=A(M,2,2)                                                         |  | TRAN476  |
| DO 126 IP=1,NPOLL                                                         |  | TRAN477  |
| CPOLL(M,1,1,IP)=CPOLL(M,1,2,IP)                                           |  | TRAN478  |
| 126 CPOLL(M,2,1,IP)=CPOLL(M,2,2,IP)                                       |  | TRAN479  |
| Q(M,1,1)=Q(M,1,2)/BARREL(M)                                               |  | TRAN480  |
| 150 Q(M,2,1)=Q(M,2,2)/BARREL(M)                                           |  | TRAN481  |
| C                                                                         |  | TRAN482  |
| C*****STORE DESIRED INPUT HYDROGRAPHS AND POLLUTOGRAPHS ON SCRATCH        |  | TRAN483  |
| C TAPE.                                                                   |  | TRAN484  |
| NTX=NSCRAT(1)                                                             |  | TRAN485  |
| DO 140 I=1,NNYN                                                           |  | TRAN486  |
| 140 WRITE(NTX) (OUTIN(I,J),J=1,NPOLS)                                     |  | TRAN487  |
| C                                                                         |  | TRAN488  |
| C*****STORE DESIRED OUTPUT HYDROGRAPHS AND POLLUTOGRAPHS ON SCRATCH       |  | TRAN489  |
| C TAPE.                                                                   |  | TRAN490  |
| NTX=NSCRAT(2)                                                             |  | TRAN491  |
| DO 155 I=1,NNPE                                                           |  | TRAN492  |
| 155 WRITE(NTX) (OUT(I,J),J=1,NPOLS)                                       |  | TRAN493  |
| C                                                                         |  | TRAN494  |
| C*****GENERATE OUTPUT TAPE TO BE USED BY INTERFACING MODEL                |  | TRAN495  |
| C                                                                         |  | TRAN495  |
| TIMX=TIME+TZERO                                                           |  | TRAN496T |
| WRITE(NTROUT) TIMX, (CUTTAP(J,1),J=1,NOUTS),((OUTTAP(J,KK),KK=2,NP        |  | TRAN497T |
| *OLS),J=1,NOUTS)                                                          |  | TRAN498  |
| C                                                                         |  | TRAN499  |
| IF(NPRINT.LT.2) GO TO 200                                                 |  | TRAN500  |
| WRITE (6,980)                                                             |  | TRAN501  |
| WRITE (6,981) TIME,(QFULL(J),J=1,NE)                                      |  | TRAN502  |
| 200 CONTINUE                                                              |  | TRAN503  |
| C                                                                         |  | TRAN504  |
| C*****PRINT FINAL SOLIDS DEPOSITION                                       |  | TRAN505  |
| C                                                                         |  | TRAN506  |
| WRITE(6,940)                                                              |  | TRAN507  |
| WRITE(6,942)                                                              |  | TRAN508  |
| DO 333 I=1,NE                                                             |  | TRAN509  |
| M=JR(I)                                                                   |  | TRAN510  |
| NT=NTYPE(M)                                                               |  | TRAN511  |
| IF(KLASS(NT).GT.2) GO TO 333                                              |  | TRAN512  |
| WRITE(6,944) NOE(M),\$CCUR(M)                                             |  | TRAN513  |
| 333 CONTINUE                                                              |  | TRAN514  |
| C THE FOLLOWING WRITE STATEMENTS ARE TO WRITE THE FLOW DEPTH FOR          |  | KINGTEMP |
| C ELEMENT 115 OF KINGMAN LAKE AREA.                                       |  | <INGTEMP |
| C WRITE(06,1000)                                                          |  | KINGTEMP |
| C1000 FORMAT('1','DEPTH OF FLOW FOR ELEMENT 115 (DOWNSTREAM) FOR EACH TIK |  | INGTEMP  |
| C *MESTEP'/' ' 'STEP 1 2 3 4 5                                            |  | KINGTEMP |
| C * 6 7 8 9 10'//)                                                        |  | KINGTEMP |
| C WRITE(C6,1001)(FLODEP(I),I=1,NDT)                                       |  | <INGTEMP |
| C1001 FORMAT(' ',10F10.1)                                                 |  | KINGTEMP |
| C WRITE(C6,1002)                                                          |  | KINGTEMP |
| C1002 FORMAT('1')                                                         |  | KINGTEMP |
| C                                                                         |  | TRAN515  |
| C*****PRINT HEADINGS                                                      |  | TRAN516  |
| C                                                                         |  | TRAN517  |
| WRITE(6,903)(TITLE(I),I=1,20)                                             |  | TRAN518  |
| WRITE(6,910)TIME,DT                                                       |  | TRAN519  |
| WRITE(6,925) (NORDER(I),I=1,NINPUT)                                       |  | TRAN520  |

|   |                                                                         |         |
|---|-------------------------------------------------------------------------|---------|
| C |                                                                         | TRAN521 |
| C | *****PRINT TOTAL HYDROGRAPHS AND POLLUTOGRAPHS FOR DESIRED ELEMENTS     | TRAN522 |
| C | IF(JPRINT.NE.1) GO TO 170                                               | TRAN523 |
|   | CALL PRINT                                                              | TRAN524 |
| C | 170 CONTINUE                                                            | TRAN525 |
| C | CALL TSTCST                                                             | TRAN526 |
|   | GO TO 9999                                                              | TRAN527 |
| C | -----ERROR MESSAGES-----                                                | TRAN528 |
|   | 9000 WRITE(6,690) NSTOR                                                 | TRAN529 |
|   | 690 FORMAT('0 *** THE NUMBER OF STORAGE UNITS, NSTOR =', I5,            | TRAN530 |
|   | * ', MUST NOT EXCEED 2')                                                | TRAN531 |
|   | 9999 RETURN                                                             | TRAN532 |
| C | *****FORMAT STATEMENTS                                                  | TRAN533 |
| C | 900 FORMAT(8F10.5)                                                      | TRAN534 |
|   | 901 FORMAT(16I5)                                                        | TRAN535 |
|   | 902 FORMAT(5I4,7F8.3)                                                   | TRAN536 |
|   | 903 FORMAT(20A4)                                                        | TRAN537 |
|   | 910 FORMAT(' TOTAL SIMULATION TIME='F7.1,' SECONDS. TIME STEP='F6.1,    | TRAN538 |
|   | 1 ' SECONDS.')                                                          | TRAN539 |
|   | 911 FORMAT(' EXTERNAL/' ELEMENT TIME STEP/' NUMBER',3X,10(16,4X))       | TRAN540 |
|   | 912 FORMAT ( /16,4X,10F10.3,/, (1CX,10F10.3) )                          | TRAN541 |
|   | 913 FORMAT ( ///' EPSILON='F8.6,5X,'NO. OF ITERATIONS IN ROUTING ROUTE' | TRAN542 |
|   | INE =',I5)                                                              | TRAN543 |
|   | 915 FORMAT(T10,'NUMBER OF ELEMENTS=',I5,/T10,'NUMBER OF TIME INT=',     | TRAN544 |
|   | 115/15X,'TIME INTERVAL='F7.1,' SECONDS.')                               | TRAN545 |
|   | 916 FORMAT ('OELEMENT PARAMETERS/' EXT. TYPE DESCRIPTION'7X,'SLOPE      | TRAN546 |
|   | 1 DISTANCE MANNING GEOM1 GEOM2 GEOM3 NUMBER AFULL QFULL                 | TRAN547 |
|   | 2 QMAX SUPER-CRITICAL/' ELE.'24X,'(FT/FT) (FT) ROUGHNESS                | TRAN548 |
|   | 3S (FT) (FT) OF (SQ.FT) (CFS) (CFS) FLOW WHEN                           | TRAN549 |
|   | 4N LESS/' NUM.'74X,'BARRELS'27X,'THAN 95% FULL?')                       | TRAN550 |
|   | 917 FORMAT ('HYDROGRAPHS AND POLLUTOGRAPHS PROVIDED TO SUBSEQUENT PRO   | TRAN551 |
|   | 1GRAMS FOR THE FOLLOWING ELEMENTS:/'(10I5))                             | TRAN552 |
|   | 920 FORMAT (2I5,3X,4A4,F8.5,F9.2,F9.4,F9.3,2F7.3,F6.1,3F9.3,5X,A4)      | TRAN553 |
|   | 921 FORMAT(///43X,'SELECTED INLET HYDROGRAPHS - CFS')                   | TRAN554 |
|   | 922 FORMAT(///43X,'SELECTED INLET POLLUTOGRAPHS - LBS/DT ')             | TRAN555 |
|   | 923 FORMAT(///42X,'SELECTED OUTFLOW HYDROGRAPHS - CFS')                 | TRAN556 |
|   | 924 FORMAT(///42X,'SELECTED OUTFLOW POLLUTOGRAPHS - LBS/DT ')           | TRAN557 |
|   | 925 FORMAT(///' INFLOW POLLUTOGRAPHS AND HYDROGRAPHS AT THE FOLLOWING   | TRAN558 |
|   | 1EXTERNAL ELEMENT NUMBERS:/'(20I5))                                     | TRAN559 |
|   | 926 FORMAT (///40X,'OVERFLOWS FROM OVERFLOW STRUCTURES')                | TRAN560 |
|   | 940 FORMAT('1',30X,'BED OF SOLIDS IN SEWER AT END OF STORM'///)         | TRAN561 |
|   | 942 FORMAT(' ',31X,'ELEMENT',20X,'SOLIDS IN',/,32X,'NUMBER',23X,'BOTTO  | TRAN562 |
|   | 1M'/,62X,'(LBS)'///)                                                    | TRAN563 |
|   | 944 FORMAT(' ',30X,16,20X,F10.5)                                        | TRAN564 |
|   | 950 FORMAT(16F5.2)                                                      | TRAN565 |
|   | 955 FORMAT('C',40X,'POLLUTANT',I5)                                      | TRAN566 |
|   | 960 FORMAT ('OTIME='F7.0,' SECONDS, TIME STEP='I5,'. ELEMENT'I4,' SURC  | TRAN567 |
|   | 1HARGING. SURCHARGE OF',F11.2, 'CU. FT. STORED AT UPSTREAM ELEMENT'     | TRAN568 |
|   | 2 I4)                                                                   | TRAN569 |
|   | 970 FORMAT ('ONTYPE DESCRIPTION ALFMAX PSIMAX AFACT REACT KTRAN         | TRAN570 |
|   | 1DEPTH KLAS INDEX ANORM QNORM DNORM')                                   | TRAN571 |
|   | 971 FORMAT (I4,4X,4A4,F7.4,3F8.4,I5,I7,I9,F8.3,2F9.5,/(67X,18,F8.3,     | TRAN572 |
|   | 1 2F9.5))                                                               | TRAN573 |
|   | 972 FORMAT ('0 NON-CONDUITS'/// NTYPE KDEPTH KLAS DESCRIPTION')         | TRAN574 |
|   | 973 FORMAT('C',16,17,18,4X,4A4)                                         | TRAN575 |
|   |                                                                         | TRAN576 |
|   |                                                                         | TRAN577 |
|   |                                                                         | TRAN578 |
|   |                                                                         | TRAN579 |
|   |                                                                         | TRAN580 |

```
980 FORMAT ('      TIME      QFULLS FROM ELEMENT 1 TO NE.')
```

```
981 FORMAT (F9.0,15F8.2/(9X,15F8.2))
```

```
983 FORMAT(//43X,'SELECTD. VELOCITIES - FT./SEC.')
```

```
      END
```

```
TRAN581
```

```
TRAN582X
```

```
TRAN583
```

```
TRAN584
```

```

      FUNCTION CN DEPTH( ALPHA)                                DEPT 1
C      COMPUTES ACRMALIZED DEPTH IN CONDUIT GIVEN NORMALIZED AREA, ALPHA. DEPT 2
      DIMENSION QI(160),CO(160)                               DEPT 3
      COMMON/TABLES/KDEPTH(25),KLASS(25),PSIMAX(15),ALFMAX(15), DEPT 4
1      NN(25),MM(25),ANORM(15,51),QNORM(15,51),             DEPT 5
2      DNORM(15,51),AFAC(15),RFAC(15)                         DEPT 6
      COMMON A(160,2,2),Q(160,2,2),CPOLL(160,2,2,3),QMAX(150), DEPT 7
1      QFULL(160),AFULL(160),DXDT(160),C1(160),SLOPE(160),    DEPT 8
2      CIST(160),GEOM1(160),ROUGH(160),NOE(160),NUE(160,3),  DEPT 9
3      INUE(160,3),NTYPE(160),JR(160),NKLASS,NE,NDT,EPSIL,    DEPT 10
4      TIME,DT,M,KFULL,N,NOS,NPULL,NPRINT,ITER,             DEPT 11
5      CCWF(160),IOLC(160),P1(160),RNOFF(160),QINFIL(160),    DEPT 12
6      WCWF(160,3),PLUFO(160,3),IR(160),P2(160),NIN(1000),  DEPT 13
7      P5(160),P6(160),P7(160),SCF(160),BARREL(160),         DEPT 14
8      TITLE(40),NPE(20),NYN(20),NORDER(70),GEOM2(160),      DEPT 15
9      GEOM3(160),P4(160),SCOUR(160),KSTORE(160)             DEPT 16
      COMMON BCDIN( 2,150),SSIN( 2,150),BDDOUT,SSOUT,COLIN(2,150), DEPT 17X
*      QINST,COUST,QINSTL( 2),COUSTL( 2),STURL( 2),QUUTO( 2),STURO( 2), DEPT 18
*      NSTOR,KSTOR,IPRINT( 2),IPOL( 2),IFLOOD( 2),ICOST( 2),DEPMAX( 2), DEPT 19
*      ATERM( 2,11),AO2DT2( 2,11),BDEPTH( 2,11),BSTOR( 2,11),COLOUT, DEPT 20X
*      DUMSTR(11),DUMDEP(11),                                  DEPT 21
*      KSTEP,VCLIN( 2,150),VOLOUT( 2,150),STOR,CUMIN( 2),CUMOUT( 2), DEPT 22
*      SBOD( 2),SSS( 2),SCOL( 2),                             DEPT 23X
*      ISTMO( 2),ISTTYP( 2),ISTOUT( 2),                        DEPT 24
*      QPUMP( 2),DSTART( 2),DSTOP( 2),                         DEPT 25
*      DTON( 2),STORMX( 2),DTPUMP( 2),DTMORE( 2),STORF( 2),APLAN( 2), DEPT 26
*      CLANC( 2),CSTCR( 2),CPS( 2),CTOTAL( 2),CPCUYD( 2),CPACRE( 2), DEPT 27
*      LP,JP,LPREV( 2),LABEL,DETENT(150),FRAC(150),OUT1(10,200) DEPT 28X
      EQUIVALENCE (QO(1),Q(1,2,2)),(QI(1),Q(1,1,2))          DEPT 29
      IF(ALPHA.EQ.0.0) GO TO 80                                DEPT 30
      NT = NTYPE(M)                                             DEPT 31
C      KDEPTH(NT) = 1 FOR CONDUIT WITH A FUNCTIONAL D-A RELATIONSHIP. DEPT 32
C      KDEPTH(NT) = 2 FOR CONDUIT WITH A TABULAR D-A RELATIONSHIP. DEPT 33
C      KDEPTH(NT) = 3 FOR ELEMENT OTHER THAN CONDUIT.          DEPT 34
      IF (KDEPTH(NT).EQ.2) GO TO 50                             DEPT 35
      IF (NT.EQ.2) GO TO 80                                     DEPT 36
      IF (NT.EQ.10) GO TO 100                                   DEPT 37
      IF (NT.EQ.11) GO TO 110                                   DEPT 38
      IF (NT.EQ.12) GO TO 120                                   DEPT 39
C      ROUTINE FOR TABULAR D-A CURVE.                           DEPT 40
C      LINEAR INTRPOLATION BETWEEN TABULAR POINTS IS USED.    DEPT 41
50  DALPHA = ANORM(NT,2)-ANORM(NT,1)                           DEPT 42
      I = ALPHA/DALPHA+1.0                                       DEPT 43
      DEPTH = CNORM(NT,I)+(ALPHA-ANCRM(NT,I))/DALPHA*(DNORM(NT,I+1)- DEPT 44
1      DNORM(NT,I))                                              DEPT 45
      RETURN                                                      DEPT 46
C      IN RECTANGULAR CONDUIT, NORMALIZED DEPTH EQUALS NORMALIZED AREA. DEPT 47
80  DEPTH = ALPHA                                                DEPT 48
      RETURN                                                      DEPT 49
C      FUNCTIONAL FORM FOR MODIFIED BASKET-HANDLE.             DEPT 50
100 AA = ALPHA*AFULL(M)                                          DEPT 51
      IF (AA.GT.GEOM3(M)) GO TO 105                             DEPT 52
      DEPTH = AA/GEOM2(M)/(GEOM1(M)+GEOM2(M)/2.0)              DEPT 53
      RETURN                                                      DEPT 54
105 ALF = (AA-GEOM3(M)+P5(M)/2.0)/P5(M)                         DEPT 55
      I = ALF/C.C2+1.0                                           DEPT 56
      DD = DNORM(1,I)+(ALF-ANORM(1,I))/0.02*(DNORM(1,I+1)-DNORM(1,I)) DEPT 57
      DEPTH = ((DD-0.5)*GEOM2(M)+GEOM1(M))/(GEOM1(M)+GEOM2(M)/2.0) DEPT 58
      RETURN                                                      DEPT 59
C      FUNCTIONAL FORM FOR RECTANGULAR, TRIANGULAR BOTTOM.      DEPT 60

```

|     |                                                                    |         |
|-----|--------------------------------------------------------------------|---------|
| 110 | AA = ALPHA*AFULL(M)                                                | DEPT 61 |
|     | AB = GECM3(M)*GECM2(M)/2.0                                         | DEPT 62 |
|     | IF (AA-AB) 111,113,115                                             | DEPT 63 |
| 111 | DEPTH = GECM3(M)/GECM1(M)*SQRT(AA/AB)                              | DEPT 64 |
|     | RETURN                                                             | DEPT 65 |
| 113 | DEPTH = GECM3(M)/GECM1(M)                                          | DEPT 66 |
|     | RETURN                                                             | DEPT 67 |
| 115 | DEPTH = (GECM3(M)+(AA-AB)/GECM2(M))/GECM1(M)                       | DEPT 68 |
|     | RETURN                                                             | DEPT 69 |
| C   | FUNCTIONAL FORM FOR RECTANGULAR, ROUND BOTTOM.                     | DEPT 70 |
| 120 | AA = ALPHA*AFULL(M)                                                | DEPT 71 |
|     | IF (AA.GT.P6(M)) GO TO 125                                         | DEPT 72 |
|     | ALF = ALPHA*AFULL(M)/(D+1+15965*GECM3(M)*GECM3(M))                 | DEPT 73 |
|     | I = ALF/0.02+1.0                                                   | DEPT 74 |
|     | DEPTH = DNCFM(1,I)+(DNCRM(1,I+1)-DNORM(1,I))/0.02*(ALF-ANDRM(1,I)) | DEPT 75 |
|     | DEPTH = DEPTH*2.0*GECM3(M)/(P2(M)*DIST(M))                         | DEPT 76 |
|     | RETURN                                                             | DEPT 77 |
| 125 | DO = P2(M)*DIST(M)-GECM1(M)                                        | DEPT 78 |
|     | DEPTH = (DO+(AA-P6(M))/GECM2(M))/(P2(M)*DIST(M))                   | DEPT 79 |
|     | RETURN                                                             | DEPT 80 |
|     | END                                                                | DEPT 81 |

```

FUNCTIONAL DPSI(ALPHA)
C FINDS DERIVATIVE OF FUNCTIONAL Q-A CURVE GIVEN A/AFULL (ALPHA).
C*****NOTE, FOR NT=2,10,11,12, SUB PSI MUST HAVE BEEN CALLED PRIOR TO
C DPSI.
C THIS WILL ALWAYS BE THE CASE IF DPSI IS CALLED ONLY FROM NEWTON.
COMMON /PSIDPS/ AA,AB,D1,D2,AAA,CATH,ALF,I,R
COMMON/TABLES/ KDEPTH(25),KLASS(25),PSIMAX(15),ALFMAX(15),
1 NN(25),MM(25),ANORM(15,51),QNORM(15,51),
2 DNORM(15,51),AFAC(15),RFAC(15)
COMMON A(160,2,2),Q(160,2,2),CPOLL(160,2,2,3),QMAX(150),
1 QFULL(160),AFULL(160),DXDT(160),C1(160),SLOPE(160),
2 DIST(160),GEOM1(160),ROUGH(160),NOE(160),NUE(160,3),
3 INUE(160,3),NTYPE(160),JR(160),BKCLASS,NE,NDT,EPSIL,
4 TIME,DT,M,KFULL,N,NOS,NPOLL,NPRINT,ITER,
5 QCWF(160),IOLD(160),P1(160),RNOFF(160),QINFIL(160),
6 WDAF(160,3),PLUTO(160,3),IR(160),P2(160),NIN(1000),
7 P5(160),P6(160),P7(160),SCF(160),BARREL(160),
8 TITLE(40),NPE(20),NYN(20),NURDER(70),GEUM2(160),
9 GECM3(160),P4(160),SCOUR(160),KSTORE(160)
COMMON BODIN( 2,150),SSIN( 2,150),BODOUT,SSOUT,COLIN(2,150),
* QINST,COLST,QINSTL( 2),QOUSTL( 2),STORL( 2),QOUTO( 2),STORO( 2),
* NSTOR,KSTOR,IPRINT( 2),IPOL( 2),IFLOOD( 2),ICOST( 2),DEPMAX( 2),
* ATERM( 2,11),AOZDT2( 2,11),BDEPTH( 2,11),BSTER( 2,11),COLOUT,
* DUMSTR(11),CUMDEP(11),
* KSTEP,VOLIN( 2,150),VOLOUT( 2,150),STOR,CUMIN( 2),CUMOUT( 2),
* SBOD( 2),SSS( 2),SCOL( 2),
* ISTMCC( 2),ISTTYP( 2),ISTOUT( 2),
* QPUMP( 2),DSTART( 2),DSTOP( 2),
* DTON( 2),STORMX( 2),DTPUMP( 2),DTMORE( 2),STORF( 2),APLAN( 2),
* CLANC( 2),CSTOR( 2),CPS( 2),CTOTAL( 2),CPCUYD( 2),CPACRE( 2),
* LP,JP,LPREV( 2),LABEL,DETENT(150),FRAC(150),OUTI(10,200)
NT = NTYPE(M)
IF(NT.EQ.2) GO TO 30
IF (NT.EQ.10) GO TO 100
IF (NT.EQ.11) GO TO 110
IF (NT.EQ.12) GO TO 120
C INCLUDE TABULAR DPSI CALC. IN CASE DPSI IS CALLED BY KLASS=2
C CONDUIT.
MMM = MM(NT)
CALPHA = ANORM(NT,2)-ANORM(NT,1)
I = ALPHA/CALPHA+1.0
IF (I.EQ.MMM) I=I-1
DPSI = (QNGRM(NT,I+1)-QNORM(NT,I))/(ANORM(NT,I+1)-ANORM(NT,I))
RETURN
C SPECIAL FUNCTIONAL FORM FOR RECTANGULAR CONDUITS.
30 CONTINUE
IF(ALPHA.LE.0.0) GO TO 45
IF (ALPHA.GT.ALFMAX(NT)) GO TO 35
IF (ALPHA.LT.0.0001) GO TO 40
DPSI = CATH*(P6(M)*ALPHA/AAA+1.666667)
RETURN
35 DPSI = (1.0-P4(M))/(1.0-ALFMAX(NT))
RETURN
40 IF (ALPHA.LE.1.E-30) GO TO 45
DPSI = 1.666667*(ALPHA/P7(M))**0.6666667
RETURN
45 DPSI = 1.E-30
RETURN
C FUNCTIONAL FORM FOR MODIFIED BASKET-HANDLE.
100 IF (ALPHA.EQ.0.0) GO TO 45

```

|                                                                  |          |
|------------------------------------------------------------------|----------|
| IF (AA.GT.GEOM3(M)) GO TO 105                                    | DPSI 61  |
| IF (ALF.LT.0.0001) GO TO 103                                     | DPSI 62  |
| DPSI = CATH*(-1.333333*R*ALF/AAA+1.666667)*AFULL(M)/GEOM3(M)     | DPSI 63X |
| RETURN                                                           | DPSI 64  |
| 103 DPSI = P6(M)*1.666667*(ALF*P7(M))*0.666667*AFULL(M)/GEOM3(M) | DPSI 65X |
| RETURN                                                           | DPSI 66  |
| 105 DPSI = (PSI(ALPHA+0.0005)-PSI(ALPHA-0.0005))/0.01            | DPSI 67  |
| RETURN                                                           | DPSI 68  |
| C FUNCTIONAL FORM FOR RECTANGULAR, TRIANGULAR BOTTOM.            | DPSI 69  |
| 110 IF (ALPHA.EQ.0.0) GO TO 45                                   | DPSI 70  |
| IF (AA.GT.AB) GO TO 115                                          | DPSI 71  |
| DPSI = P7(M)*1.333333*ALPHA**0.3333333                           | DPSI 72  |
| RETURN                                                           | DPSI 73  |
| 115 IF (ALPHA.GT.ALFXAX(NT)) GO TO 35                            | DPSI 74  |
| DPSI=CATH*(-0.666667*ALPHA/AAA*(2.0*GEOM1(M)-GEOM3(M))+1.666667) | DPSI 75  |
| RETURN                                                           | DPSI 76  |
| C FUNCTIONAL FORM FOR RECTANGULAR, ROUND BOTTOM.                 | DPSI 77  |
| 120 IF (ALPHA.EQ.0.0) GO TO 45                                   | DPSI 78  |
| IF (AA.GT.F6(M)) GO TO 125                                       | DPSI 79  |
| DPSI = (CNCRM(1,I+1)-CNORM(1,I))/0.02*P7(M)                      | DPSI 80  |
| RETURN                                                           | DPSI 81  |
| 125 IF (ALPHA.GT.ALFXAX(NT)) GO TO 35                            | DPSI 82  |
| DPSI = CATH*(1.666667-1.333333*AFULL(M)/GEOM2(M)/D2)             | DPSI 83  |
| RETURN                                                           | DPSI 84  |
| END                                                              | DPSI 85  |



|    |                                                                 |          |
|----|-----------------------------------------------------------------|----------|
|    | SUBROUTINE DWLOAD(DWDAYS)                                       | DWLO 1   |
| C  | THIS SUBROUTINE SETS UP INITIAL CONDITIONS OF SEDIMENTATION     | DWLO 2   |
| C  | BASED UPON 'DWDAYS' OF DRY WEATHER FLOW PRIOR TO THE STORM      | DWLO 3   |
|    | DIMENSION QI(160),QO(160),QOI(160),QO2(160)                     | DWLO 4   |
|    | COMMON/DWLF/ CVCHF(7),CVBOD(7),DVSS(7),HVDWF(24),HVBOD(24),     | DWLO 5   |
| 1  | HVSS(24),HVCOLI(24),KTNUM,KDAY,KHOUR,KMINS                      | DWLO 6   |
|    | COMMON/TABLES/KDEPTH(25),KLASS(25),PSIMAX(15),ALFMAX(15),       | DWLO 7   |
| 1  | NA(25),MM(25),ANORM(15,51),QNORM(15,51),                        | DWLO 8   |
| 2  | DNORM(15,51),AFAC(15),RFAC(15)                                  | DWLO 9   |
|    | COMMON A(160,2,2),Q(160,2,2),CPOLL(160,2,2,3),QMAX(160),        | DWLO 10  |
| 1  | QFULL(160),AFULL(160),DXDT(160),C1(160),SLOPE(160),             | DWLO 11  |
| 2  | CIST(160),GEOM1(160),ROUGH(160),NOE(160),NUE(160,3),            | DWLO 12  |
| 3  | INUE(160,3),NTYPE(160),JR(160),NKLASS,NE,NDT,EPSIL,             | DWLO 13  |
| 4  | TIME,DT,M,KFULL,N,NOS,NPULL,NPRINT,ITER,                        | DWLO 14  |
| 5  | CDWF(160),IOLC(160),P1(160),RNOFF(160),QINFIL(160),             | DWLO 15  |
| 6  | WCWF(160,3),PLUTO(160,3),IR(160),P2(160),NIN(1000),             | DWLO 16  |
| 7  | P5(160),P6(160),P7(160),SCF(160),BARREL(160),                   | DWLO 17  |
| 8  | TITLE(40),NPE(20),NYN(20),NORDER(70),GEOM2(160),                | DWLO 18  |
| 9  | GEOM3(160),P4(160),SCCUR(160),KSTORE(160)                       | DWLO 19  |
|    | COMMON BCDIN(2,150),SSIN(2,150),BODOUT,SSOUT,COLIN(2,150),      | DWLO 20X |
| *  | QINST,COUST,QINSTL(2),QOUSTL(2),STURL(2),QOUTO(2),STORO(2),     | DWLO 21  |
| *  | NSTOR,KSTOR,IPRINT(2),IPOL(2),IFLOOD(2),ICOST(2),DEPMAX(2),     | DWLO 22  |
| *  | ATERM(2,11),AO2DT2(2,11),BDEPTH(2,11),BSTOR(2,11),COLOUT,       | DWLO 23X |
| *  | DUMSTR(11),DUMDEP(11),                                          | DWLO 24  |
| *  | KTSTEP,VCLIN(2,150),VOLOUT(2,150),STOR,CUMIN(2),CUMOUT(2),      | DWLO 25  |
| *  | SBCO(2),SSS(2),SCOL(2),                                         | DWLO 26X |
| *  | ISTMOC(2),ISTTYP(2),ISTOUT(2),                                  | DWLO 27  |
| *  | QPUMP(2),DSTART(2),DSTOP(2),                                    | DWLO 28  |
| *  | DTON(2),STORMX(2),DTPUMP(2),DTMORE(2),STORF(2),APLAN(2),        | DWLO 29  |
| *  | CLAND(2),CSTOR(2),CPS(2),CTOTAL(2),CPCUYD(2),CPACRE(2),         | DWLO 30  |
| *  | LP,JP,LPREV(2),LABEL,DETENT(150),FRAC(150),OUT1(10,200)         | DWLO 31X |
|    | EQUIVALENCE (QO1(1),QMAX(1)),(QO2(1),QFULL(1))                  | DWLO 32  |
|    | EQUIVALENCE (QO(1),Q(1,2,2)),(QI(1),Q(1,1,2))                   | DWLO 33  |
|    | REAL KVAL                                                       | DWLO 34  |
|    | DO 10 IK=1,NE                                                   | DWLO 35  |
| 10 | SCOUR(IK)=C.C                                                   | DWLO 36  |
| C  | BEGIN DO LOOP ON HOUR OF DAY                                    | DWLO 37  |
|    | DO 200 KJ=1,24                                                  | DWLO 38  |
| C  | MODEL DWF FOR AN HOUR                                           | DWLO 39  |
| C  | START WITH EMPTY SEWER EACH HOUR(EXCEPT FOR ANY DEPOSITION FROM | DWLO 40  |
| C  | PREVIOUS HOUR                                                   | DWLO 41  |
|    | DO 5 IK=1,NE                                                    | DWLO 42  |
|    | Q(IK,1,1)=0.0                                                   | DWLO 43  |
|    | Q(IK,2,1)=C.C                                                   | DWLO 44  |
|    | CPOLL(IK,2,1,2)=0.0                                             | DWLO 45  |
|    | CPOLL(IK,1,1,2)=0.0                                             | DWLO 46  |
| 5  | CONTINUE                                                        | DWLO 47  |
|    | DO 100 I=1,NE                                                   | DWLO 48  |
|    | M=JR(I)                                                         | DWLO 49  |
|    | SUM1=0.0                                                        | DWLO 50  |
|    | SUM2=0.0                                                        | DWLO 51  |
| C  | ADD UP UPSTREAM FLOWS AND CONVERT LOADS TO LBS/SEC              | DWLO 52  |
|    | DO 20 J=1,3                                                     | DWLO 53  |
|    | L=INUE(M,J)                                                     | DWLO 54  |
|    | NTU=NTYPE(L)                                                    | DWLO 55  |
|    | IF(NTU.LE.17.OR.NTU.GE.23) GO TO 19                             | DWLO 56  |
|    | KK=GEOM3(L)                                                     | DWLO 57  |
|    | QQ=QO2(L)                                                       | DWLO 58  |
|    | IF(NUE(M).EQ.KK) CQ=QO1(L)                                      | DWLO 59  |
|    | GO TO 19                                                        | DWLO 60  |

|     |                                                                 |          |
|-----|-----------------------------------------------------------------|----------|
| 18  | QQ=Q(L,2,1)*BARREL(L)                                           | DWLO 61  |
| 19  | SUM1=SUM1+CQ                                                    | DWLO 62  |
| 20  | SUM2=SUM2+CPOLL(L,2,1,2)*QQ                                     | DWLO 63  |
| C   | ADD DWF AND INFILTRATION FOR THIS HOUR                          | DWLO 64  |
|     | SUM1=SUM1+CDWF(M)*HVDWF(KJ) + QINFIL(M)                         | DWLO 65  |
|     | SUM2=SUM2+WDWF(M,2)*HVSS(KJ)*HVDWF(KJ)                          | DWLO 66  |
|     | Q(M,1,1)=SUM1/BARREL(M)                                         | DWLO 67  |
|     | Q(M,2,1)=Q(M,1,1)                                               | DWLO 68  |
| C   | CALCULATE LOADS IN LBS/CF                                       | DWLO 69  |
|     | IF(SUM1.LE.0.0) GO TO 100                                       | DWLO 70  |
|     | CPOLL(M,1,1,2)= SUM2/SUM1                                       | DWLO 71  |
|     | CPOLL(M,2,1,2)=SUM2/SUM1                                        | DWLO 72  |
|     | NT=NTYPE(M)                                                     | DWLO 73  |
|     | IF (NT.EC.22) GO TO 28                                          | DWLO 74  |
|     | IF(NT.LT.18.OR.NT.GT.22) GO TO 30                               | DWLO 75  |
| C   | IN A FLOW DIVIDER TYPE ELEMENT, ASSUME ALL DWF IS NOT DIVERTED. | DWLO 76  |
|     | QO1(M)=SUM1                                                     | DWLO 77  |
|     | QO2(M)=0.0                                                      | DWLO 78  |
|     | GO TO 30                                                        | DWLO 79  |
| 28  | QO1(M) = 0.0                                                    | DWLO 80  |
|     | QO2(M) = SUM1                                                   | DWLO 81  |
| C   | IF A NON-CONDUIT DO NOT CALCULATE ANY DEPOSITION                | DWLO 82  |
| 30  | IF(KLASS(NT).EQ.3) GO TO 100                                    | DWLO 83  |
| C   | ASSUME UNIFORM AREA OF FLOW THROUGHOUT ELEMENT                  | DWLO 84  |
|     | PS=Q(M,1,1)/CFULL(M)                                            | DWLO 85  |
|     | CALL FINDA(PS,A(M,1,1))                                         | DWLO 86  |
| C   | FIND HYDRAULIC RADIUS GIVEN THE FLOW AREA.                      | DWLO 87  |
|     | AREAF=A(M,1,1)                                                  | DWLO 88  |
|     | RHYD=RADH(AREAF)                                                | DWLO 89  |
| C   | DETERMINE AMOUNT OF DEPOSITION AND SCOUR                        | DWLO 90  |
|     | SPG=2.7                                                         | DWLO 91  |
|     | KVAL=.056                                                       | DWLO 92  |
| C   | FIND THE CRITICAL DIAMETER FOR MOTION OF SOLIDS.                | DWLO 93  |
|     | CRITD=RHYD*SLOPE(M)/(KVAL*(SPG-1.0))                            | DWLO 94  |
|     | CRITD=CRITD*304.8                                               | DWLO 95  |
|     | IF(CRITD.LE..59) PCT1=-1.2471*CRITD+1.0                         | DWLO 96  |
|     | IF(CRITD.GT..59.AND.CRITD.LE.2.0) PCT1=-.1501*CRITD+.3527       | DWLO 97  |
|     | IF(CRITD.GT.2.0.AND.CRITD.LE.10.0) PCT1=-.00656*CRITD+.0656     | DWLO 98  |
|     | IF(CRITD.GT.10.0) PCT1=0.0                                      | DWLO 99  |
|     | IF(CRITD.LE.0.0) PCT1=1.0                                       | DWLO 100 |
|     | PCT2=PCT1                                                       | DWLO 101 |
|     | SCOUR(M)=SCOUR(M)+PCT2*SUM2*3600.                               | DWLO 102 |
|     | SUM3=(1.0-PCT2)*SUM2+(1.0-PCT1)*SCOUR(M)/3600.                  | DWLO 103 |
|     | CPOLL(M,1,1,2)= SUM3 /SUM1                                      | DWLO 104 |
|     | CPOLL(M,2,1,2)=CPOLL(M,1,1,2)                                   | DWLO 105 |
|     | SCOUR(M)=PCT1*SCOUR(M)                                          | DWLO 106 |
| 100 | CONTINUE                                                        | DWLO 107 |
| 200 | CONTINUE                                                        | DWLO 108 |
| C   | ASSUME SAME BED LOAD ACCUMULATION FOR EACH DWDAY                | DWLO 109 |
|     | DO 210 I=1,NE                                                   | DWLO 110 |
|     | M=JR(I)                                                         | DWLO 111 |
|     | SCOUR(M)=DWDAYS*SCOUR(M)                                        | DWLO 112 |
| C   | SET UPPER LIMIT OF BED LOAD AS HALF OF PIPE VOLUME              | DWLO 113 |
| C   | THE NUMBER 84.28=2.7*62.4/2.0                                   | DWLO 114 |
|     | ULIMIT=84.28*AFULL(M)*DIST(M)*BARREL(M)                         | DWLO 115 |
|     | IF(SCOUR(M).GT.ULIMIT) SCOUR(M)=ULIMIT                          | DWLO 116 |
| 210 | CONTINUE                                                        | DWLO 117 |
| C   | PRINT OUT INITIAL DWLOAD CONDITIONS                             | DWLO 118 |
|     | WRITE(6,900) DWDAYS                                             | DWLO 119 |
|     | WRITE(6,910)                                                    | DWLO 120 |

|                                                                               |         |
|-------------------------------------------------------------------------------|---------|
| DO 220 I=1,NE                                                                 | DWLO121 |
| M=JR(I)                                                                       | DWLO122 |
| NT=NTYPE(M)                                                                   | DWLO123 |
| IF(KLASS(NT).GT.2) GO TO 220                                                  | DWLO124 |
| WRITE(6,920) NOE(M),SCOUR(M)                                                  | DWLO125 |
| 220 CONTINUE                                                                  | DWLO126 |
| 900 FORMAT('1',28X,'INITIAL BED OF SOLIDS (LBS) IN SEWER DUE TO',//,          | DWLO127 |
| 130X,F5.1,' DAYS OF DRY WEATHER PRIOR TO STORM '///)                          | DWLO128 |
| 910 FORMAT(' ',31X,'ELEMENT',20X,'SOLIDS IN',//,32X,'NUMBER',23X,'BOTTDWLO129 |         |
| 1M'//,62X,' (LBS)'///)                                                        | DWLO130 |
| 920 FURMAT(' ',30X,I6,20X,F10.5)                                              | DWLO131 |
| RETURN                                                                        | DWLO132 |
| END                                                                           | DWLO133 |

```

SUBROUTINE FILTH
C SUBROUTINE TO DETERMINE AVERAGE DAILY DWF QUAL AND QUAN
C ***
COMMON/CRWF/ DVDWF(7),DV8OD(7),DVSS(7),HVDWF(24),HVBOD(24),
1 HVSS(24),HVCOLI(24),KTNUM,KDAY,KHOUR,KMINS
COMMON/TABLES/KDEPTH(25),KLASS(25),PSIMAX(15),ALFMAX(15),
1 NN(25),MM(25),ANORM(15,51),QNORM(15,51),
2 DNORM(15,51),AFAC(15),RFAC(15)
COMMON A(160,2,2),Q(160,2,2),CPOLL(160,2,2,3),QMAX(160),
1 QFULL(160),AFULL(160),DXDT(160),C1(160),SLOPE(160),
2 EIST(160),GEOM1(160),ROUGH(160),NNE(160),NUE(160,3),
3 INUE(160,3),NTYPE(160),JR(160),NKLASS,NE,NDT,EPSIL,
4 TIME,DI,M,KFULL,N,NQS,NPOLL,NPRINT,1IEK,
5 CDWF(160),IOLC(160),P1(160),RNOFF(160),QINFIL(160),
6 WDW(160,3),PLUTO(160,3),IR(160),P2(160),NIN(1000),
7 P5(160),P6(160),P7(160),SCF(160),BARREL(160),
8 TITLE(40),NPE(20),NYN(20),NORDER(70),GEOM2(160),
9 GEOM3(160),P4(160),SCOUR(160),KSTORE(160)
COMMON BCDIN( 2,150),SSIN( 2,150),BODOUT,SSOUT,
* QINST,CGLST,QINSTL( 2),QOLSTL( 2),STURL( 2),QOUTO( 2),STURO( 2),
* NSTOR,KSTOR,IPRINT( 2),IPOL( 2),IFLOOD( 2),ICOST( 2),DEPMAX( 2),
* ATERM( 2,11),AO2DT2( 2,11),BDEPTH( 2,11),BSTOR( 2,11),COLOUT,
* DUMSTR(11),DUMDEP(11),
* KSTEP,VCLIN( 2,150),VOLOUT( 2,150),STOR,CUMIN( 2),CUMOUT( 2),
* SBOD( 2),SSS( 2),SCOL( 2),
* ISTHCD( 2),ISTTYP( 2),ISTOUT( 2),
* QPUMP( 2),DSTART( 2),DSTOP( 2),
* DTON( 2),STORMX( 2),DTPUMP( 2),DTMORE( 2),STORF( 2),APLAN( 2),
* CLAND( 2),CSTOR( 2),CPS( 2),CTOTAL( 2),CPCUYD( 2),CPACRE( 2),
* LP,JP,LPREV( 2),LABEL,DETENT(150),FRAC(150),OUT1(10,20)
C ***
C *****
C READ DAILY AND HOURLY CORRECTION FACTORS FOR SEWAGE
READ(5,501) DVCWF,DVBCD,DVSS
501 FORMAT(7F10.0)
READ(5,502) HVDWF,HVBOD,HVSS,HVCOLI
502 FORMAT (8F10.0)
C *****
C *****
C READ TOTAL NUMBER OF SUBAREAS, TYPE OF FLOW AND QUALITY DATA
C AVAILABLE, NUMBER OF PROCESS FLOWS, TIME SIMULATION BEGINS( DAY,
C HOUR, AND MINUTE), CURRENT VALUE OF THE CONSUMER PRICE INDEX,CPI,:
C CURRENT VALUE OF THE COMPOSITE CONSTRUCTION COST INDEX,CCCI
READ(5,503) KTNUM,KASE,NPF,KDAY,KHOUR,KHINS,CPI,CCCI,POPJLA
C POPJLA IS THE TCTAL POP IN ALL AREAS IN THOUSANDS
503 FORMAT (6I5,2F5.1,F10.3)
WRITE(6,601)
601 FORMAT('1',27X,'QUANTITY AND QUALITY OF D W F FOR EACH SUBAREA'//)
C *****
C COMPUTE TCTAL INFILTRATION WITHIN STUDY AREA
AINFIL=0.0
DO 415 KK=1,NE
415 AINFIL= AINFIL+QINFIL(KK)
C AINFIL=4.4
C IF(KASE.NE.1) ADWF=0.0
C IF KASE = 1 THE AVERAGE FLOW AND CHARACTERISTICS ARE KNOWN FOR THE
C ENTIRE AREA BUT NOT FOR THE INDIVIDUAL SUBAREAS (EG SEWAGE PLANT

```

|                                                                            |         |
|----------------------------------------------------------------------------|---------|
| C DATA)                                                                    | FILT 60 |
| C IF KASE = 2 NO SUCH DATA IS AVAILABLE                                    | FILT 61 |
| IF (KASE - 1) 411,411,412                                                  | FILT 62 |
| C                                                                          | FILT 63 |
| C*****KASE 1 FOLLOWS                                                       | FILT 64 |
| C                                                                          | FILT 65 |
| C ACWF ETC. ARE THE AVE DAILY VALUES OF DRY WEATHER FLOW IN CFS, BOD INFIL | FILT 66 |
| C MG/L, SUSP SOLIDS IN MG/L, RESPECTIVELY                                  | FILT 67 |
| C (FOR EXAMPLE, FROM A TREATMENT PLANT SERVING THE ENTIRE STUDY AREA       | FILT 68 |
| 411 READ(5,504) ADWF,ABOD,ASUSO,ACOLI                                      | FILT 69 |
| 504 FORMAT (3F10.0,E10.2)                                                  | FILT 70 |
| C ACOLI IS THE VALUE OF TOTAL COLIFORMS IN MPN/100ML                       | FILT 71 |
| C READ TOTAL AREA TO PLANT, INDUSTRIAL AREA, COMMERCIAL, RESIDENTIAL       | FILT 72 |
| C HIGH INCOME, RESIDENTIAL AVE INCOME, RESIDENTIAL LOW INCOME, RESI-       | FILT 73 |
| C DENTIAL WITH GARBAGE GRINDERS, PARKS AND OPEN AREA - ALL IN ACRES        | FILT 74 |
| C AND TOTAL POPULATION                                                     | FILT 75 |
| READ (5,505) TOTA, TINA, TCA, TRHA, TRAA, TRLA, TRGGA, TPOA                | FILT 76 |
| 505 FORMAT (8F6.0)                                                         | FILT 77 |
| C CCMPUTE TOTAL BOD AND SUSPENDED SOLIDS IN POUNDS PER DAY                 | FILT 78 |
| TOTBOD= (ACWF/1.547)*ABOD*8.34                                             | FILT 79 |
| TOTSS=(ACWF/1.547)*ASUSO*8.34                                              | FILT 80 |
| C CCMPUTE TOTAL COLIFORMS IN MPN/DAY                                       | FILT 81 |
| TCOLI=ACOLI*ADWF*2.447E+7                                                  | FILT 82 |
| C MAKE DATA CORRECTION FOR INFILTRATION                                    | FILT 83 |
| C10WF= ACWF- AINFIL                                                        | FILT 84 |
| C MAKE DATA CORRECTION FOR PROCESS FLOWS                                   | FILT 85 |
| C FIRST INITIALIZE SUMATION QUANTITIES                                     | FILT 86 |
| SUMQPF=0.0                                                                 | FILT 87 |
| SUMBOD=0.0                                                                 | FILT 88 |
| SUMSS=0.0                                                                  | FILT 89 |
| C NPF IS NUMBER OF PROCESS FLOWS                                           | FILT 90 |
| IF(NPF.LE.0) GO TO 416                                                     | FILT 91 |
| DO 500 JJ=1,NPF                                                            | FILT 92 |
| C READ PROCESS FLOW CHARACTERISTICS AND LOCATIONS                          | FILT 93 |
| READ(5,506) INPUT, QPF, BODPF, SUSPF                                       | FILT 94 |
| 506 FORMAT(15,6F10.3)                                                      | FILT 95 |
| SUMQPF=SUMQPF + QPF                                                        | FILT 96 |
| SUMBOD=SUMBOD + (QPF/1.547)*BODPF*8.34                                     | FILT 97 |
| SUMSS=SUMSS + (QPF/1.547)*SUSPF*8.34                                       | FILT 98 |
| 500 CONTINUE                                                               | FILT 99 |
| 416 C2DWF=C10WF-SUMQPF                                                     | FILT100 |
| IF(C2DWF.LE.0.0) GO TO 412                                                 | FILT101 |
| C1BOD=TGTBOD-SUMBOD                                                        | FILT102 |
| C1SS=TOTSS-SUMSS                                                           | FILT103 |
| C ****                                                                     | FILT104 |
| C MAKE FINAL CORRECTIONS TO ALLOW FOR INCOME VARIATIONS, COMMERCIAL        | FILT105 |
| C USE, GARBAGE GRINDERS, AND POPULATION                                    | FILT106 |
| C COMPUTE RESIDENTIAL AND COMMERCIAL AREA CONTRIBUTING TO PLANT            | FILT107 |
| TDWFA=TOTA-TINA-TPOA                                                       | FILT108 |
| C CCMPUTE WEIGHTED DWFA BASED ON EXPECTED VARIATIONS IN SEWAGE             | FILT109 |
| C STRENGTH.                                                                | FILT110 |
| WTDWFA= 0.9*TCA+1.2*TRHA+1.0*TRAA+0.8*TRLA+1.3*TRGGA                       | FILT111 |
| C CCMPUTE CORRECTION FACTOR TO WEIGHT SEWAGE STRENGTH - NOTE BOD AND       | FILT112 |
| C SS ARE AFFECTED EQUALLY                                                  | FILT113 |
| TDTR=TDWFA/TRGGA                                                           | FILT114 |
| IF(TDTR.LE.0.0) CF=1.0                                                     | FILT115 |
| IF(TDTR.LE.0.0) GO TO 413                                                  | FILT116 |
| CF=WTDWFA/TDWFA                                                            | FILT117 |
| C MODIFY MEASURED STRENGTHS TO PERMIT WEIGHTING IN SUBAREA COMPS           | FILT118 |
| 413 C2BOD=C1BOD/CF                                                         | FILT119 |

```

C2SS=C1SS/CF
IF(C2BCD.LE.0.0) C2BOD=0.0
IF(C2SS.LE.0.0) C2SS= 0.0
C ***
C CCMPUTE AVERAGE CORRECTED AND WEIGHTED DWF CHARACTERISTICS IN
C LBS/DAY/CFS.
A1BOD=C2BCD/C2DWF
A1SS=C2SS/C2DWF
A1COLI=TCOLI/(POPULA*1000.)
C A1COLI IS THE AVERAGE NUMBER OF COLI PER CAPITA PER DAY
GO TO 414
C
C*****KASE 2 FOLLOWS
C
C IN THE EVENT MEASURED FLOW CHARACTERISTICS ARE NOT AVAILABLE ASSUME
C FOR A1BOD, A1SS, AND A1COLI
412 A1BOD=1300.
C THIS ASSUMES 85 GAL/CAP/DAY AND 0.2CLBS/CAP/DAY FOR AVE INCOME
A1SS=1420.
C THIS ASSUMES 85 GAL/CAP/DAY AND 0.22 LBS/CAP/DAY FOR AVE INCOME
A1COLI=2.00E+11
C A1COLI,MPN/DAY/CAP, ASSUMES 200 BILLION COLIFORMS PER CAPITA PER DAY
ADWF=PCFULA*1000.*1.547*100./1000000.
C THIS ASSUMES 100 GAL/CAP/DAY INCLUDING INFILTRATION
C
C*****
C
414 WRITE (6,602) A1BOD, A1SS, A1COLI, ADWF
602 FORMAT ('C',19X,'A1BOD =', F10.2,'LBSPERDAY/CFS'/21X,
. 'A1SS =', F10.2,' LBSPERDAY/CFS'/19X,'A1COLI =',1PE11.2,
. ' MPN/DAY PER CAPITA'/21X,'ADWF =',0PF10.2,' CFS'//)
C ***
SMDWF=0.00
SMBOD=C.CC
SMTSS=C.CC
TOTPOP=0.0
SMMQ=C.C
WRITE(6,603)
603 FORMAT ('O','KNUM',1X,'INPUT',7X,'DWF',2X,'+',2X,'INFIL',2X,'=',2X,FIL158
*, 'QQDWF',1X,'KLAND',5X,'DWBOD',7X,'DWSS',9X,'TOTPOP',3X,'BODCONC' FIL159
.,4X,'SSCNC',4X,' COLIFORMS')
WRITE (6,604)
604 FORMAT (' ',17X,'CFS',6X,'CFS',7X,'CFS',11X,'LBS/MIN',4X,'LBS/MIN' FIL162
. ,7X,'PERSCNS',5X,'MG/L',6X,'MG/L',5X,' MPN/100ML'//)
QQ=0.0
QQDWF=C.0
SMTDWF=C.0
C
C*****TIMESTEP COMPUTATIONS FOLLOW
C*****FLOW
C
DO 300 I=1,KTAUM
DWF=0.0
C COMPUTE DWF FOR EACH SUBAREA
C SUBAREAS IN A CITY ARE CHOSEN PRIMARILY ON LAND USE CRITERIA
READ(5,507)KNUM,INPUT,KLAND,METHOD,KUNIT,WATER,PRICE,SEWAGE,ASUB, FIL175
1POPDEN,DWLNGS,FAMILY,VALUE,PCGG,SAQPF,SABPF,SASPF,XINCUM,MSUBT FIL176
507 FORMAT(243,311,13F5.0,12) FIL177X
C DATA CHECK AND ASSUMPTIONS NECESSARY TO OVERCOME MISSING DATA FIL178
C IF HOUSE VALUATION(VALUE) IS UNDEFINED, ASSUME $20,000 HOMES FIL179

```

|     |                                                                   |         |
|-----|-------------------------------------------------------------------|---------|
| 110 | IF(VALUE)111,111,112                                              | FILT180 |
| 111 | VALUE=20.C                                                        | FILT181 |
| C   | CHECK ON WHETHER POPULATION DENSITY OR THE NUMBER OF DWELLINGS    | FILT182 |
| C   | FOR EACH SUBAREA ARE INCLUDED AS INPUT DATA                       | FILT183 |
| C   | CORRECT VALUE TO 1960 DOLLARS USING DEPARTMENT OF INTERIOR        | FILT184 |
| C   | COMPOSITE CONSTRUCTION COST INDEX(1960 VALUE OF CCCI=103.0)       | FILT185 |
| 112 | IF(CCCI.LE.0.0) CCCI= 103.0                                       | FILT186 |
|     | VALUE= VALUE*103.0/CCCI                                           | FILT187 |
|     | IF(DWLNGS)113,113,130                                             | FILT188 |
| 113 | IF(PCPDEN)114,114,115                                             | FILT189 |
| C   | IF BOTH POPULATION DENSITY AND NUMBER OF DWELLINGS PER SUBAREA    | FILT190 |
| C   | ARE NOT TABULATED, ASSUME 10 DWELLING UNITS PER ACRE              | FILT191 |
| 114 | DWLNGS=10.0*ASUE                                                  | FILT192 |
|     | GO TO 130                                                         | FILT193 |
| 115 | IF(FAMILY)116,116,117                                             | FILT194 |
| C   | IF POPULATION DENSITY BUT NOT NUMBER PER HOUSEHOLD(FAMILY) IS     | FILT195 |
| C   | TABULATED, ASSUME FAMILY=3.0                                      | FILT196 |
| 116 | FAMILY=3.0                                                        | FILT197 |
| 117 | DWLNGS=PCPDEN*ASUB/FAMILY                                         | FILT198 |
| C   | METHOD USED TO ESTIMATE D W F WITH MEASURED SEWAGE OR WATER FLOWS | FILT199 |
| 130 | IF(SEWAGE)131,131,136                                             | FILT200 |
| 131 | IF(WATER)100,100,133                                              | FILT201 |
| C   | CHECK UNITS OF WATER DATA AND USE APPROPRIATE CONVERSION FACTOR   | FILT202 |
| 133 | IF(KUNIT)135,135,134                                              | FILT203 |
| 134 | DWF=WATER/(30.4*24.*3600.)*1000.                                  | FILT204 |
|     | GO TO 200                                                         | FILT205 |
| 135 | DWF=WATER/(30.4*24.*3600.)*0.134*1000.                            | FILT206 |
|     | GO TO 200                                                         | FILT207 |
| 136 | DWF=SEWAGE                                                        | FILT208 |
|     | GO TO 200                                                         | FILT209 |
| C   | GO TO APPROPRIATE DWF ESTIMATE DEPENDING ON TYPE OF LAND USE      | FILT210 |
| C   | TYPE 1 SINGLE FAMILY RESIDENTIAL                                  | FILT211 |
| C   | TYPE 2 MULTI-FAMILY RESIDENTIAL                                   | FILT212 |
| C   | TYPE 3 COMMERCIAL                                                 | FILT213 |
| C   | TYPE 4 INDUSTRIAL                                                 | FILT214 |
| C   | TYPE 5 UNDEVELOPED OR PARKLANDS                                   | FILT215 |
| 100 | GO TO(10,20,30,40,50),KLAND                                       | FILT216 |
| C   | METHOD TO ESTIMATE DWF IN SINGLE FAMILY RESIDENTIAL SUBAREAS      | FILT217 |
| C   | CHECK ON WHETHER PRICE OF WATER IS INCLUDED AS INPUT DATA         | FILT218 |
| C   | THIS DETERMINES WHICH OF LINAWEAVER'S EQUATION TO USE             | FILT219 |
| C   | LINAWEAVER'S EQUATIONS ALSO DEPEND ON WHETHER WATER IS METERED    | FILT220 |
| C   | ASSUME FLAT RATE PRICING IF METERING IS NOT INDICATED IN DATA     | FILT221 |
| 10  | IF(METHOD=1)20,11,20                                              | FILT222 |
| 11  | IF(PRICE)12,12,13                                                 | FILT223 |
| C   | EQUATION 1 LINAWEAVER                                             | FILT224 |
| 12  | DWF=(178.+3.28*VALUE)*DWLNGS*0.134/(24.*3600.)                    | FILT225 |
|     | GO TO 200                                                         | FILT226 |
| C   | EQUATION 2 METERED WITH PUBLIC SEWER LINAWEAVER AND HOWE          | FILT227 |
| C   | CORRECT PRICE TO 1965 DOLLARS USING THE CONSUMER PRICE INDEX(1965 | FILT228 |
| C   | CPI=109.0)                                                        | FILT229 |
| 13  | IF (CPI.LE.0.0) CPI= 109.9                                        | FILT230 |
|     | PRICE= PRICE*109.9/CPI                                            | FILT231 |
|     | DWF=(206.+3.47*VALUE-1.30*PRICE)*DWLNGS*0.134/(24.*3600.)         | FILT232 |
|     | GO TO 200                                                         | FILT233 |
| C   | METHOD TO ESTIMATE DWF IN MULTI-FAMILY RESIDENTIAL SUBAREAS       | FILT234 |
| C   | CHECK ON WHETHER VALUE AND FAMILY ARE INCLUDED AS INPUT DATA      | FILT235 |
| 20  | IF(FAMILY)21,21,22                                                | FILT236 |
| 21  | FAMILY=3.0                                                        | FILT237 |
| C   | EQUATION 3 FLAT RATE AND APARTMENTS WITH PUBLIC SEWER L. AND H.   | FILT238 |
| 22  | DWF=(28.9+4.39*VALUE+33.6*FAMILY)*DWLNGS*0.134/(24.*3600.)        | FILT239 |

|                                                                    |          |
|--------------------------------------------------------------------|----------|
| GO TO 200                                                          | FILT240  |
| C DWF MODEL REQUIRES WATER OR SEWAGE INPUTS                        | FILT241  |
| C TO ESTIMATE DWF FOR COMMERCIAL OR INDUSTRIAL SUBAREAS            | FILT242  |
| 30 DWF=DWF+SAQPF                                                   | FILT243  |
| GO TO 200                                                          | FILT244  |
| 40 DWF=DWF+SAQPF                                                   | FILT245  |
| GO TO 200                                                          | FILT246  |
| 50 DWF=0.0                                                         | FILT247  |
| 200 CONTINUE                                                       | FILT248  |
| POP=ASUB*POPCEN                                                    | FILT249  |
| TOTPOP=TCTPOP+POP                                                  | FILT250  |
| IF(ADWF.LE.0.0) QQF=0.0                                            | FILT251  |
| IF(ADWF.GT.0.0) QQF=A1NFIL/ADWF                                    | FILT252  |
| CQ=QQF*DWF                                                         | FILT253  |
| IF(KLAND.NE.4) CQ=QQF*DWF                                          | FILT253A |
| IF(KLAND.EQ.4) CQ=0.0                                              | FILT253B |
| QQDWF=DWF+CQ                                                       | FILT254  |
| C                                                                  | FILT255  |
| C*****QUALITY                                                      | FILT256  |
| C                                                                  | FILT257  |
| C COMPUTE DWF QUALITY FOR EACH SUBAREA                             | FILT258  |
| C DAILY QUALITY AVERAGES ARE CONVERTED TO SUBAREA QUALITY RATES    | FILT261  |
| C (LBS/SEC)                                                        | FILT262  |
| 201 CONTINUE                                                       | FILT263  |
| C1DT=CT/(24.0*60.0*60.0)                                           | FILT264  |
| IF(XINCC.LE.0.0) XINCOM=VALUE/2.5                                  | FILT265  |
| IF (KLAND.LE.2) GO TO 421                                          | FILT266  |
| IF(KLAND.EQ.3) GO TO 422                                           | FILT267  |
| IF (KLAND.EQ.4) GO TO 423                                          | FILT268  |
| 424 DWBOD=DWF*A1BOD*C1DT                                           | FILT269  |
| DWSS=DWF*A1SS*C1DT                                                 | FILT270  |
| GO TO 25                                                           | FILT271  |
| 422 DWBOD=DWF*0.9*A1BOD*C1DT                                       | FILT272  |
| DWSS=DWF*0.9*A1SS*C1DT                                             | FILT273  |
| GO TO 25                                                           | FILT274  |
| 423 DWBOD=(SAQPF/1.547)*SABPF*8.34*C1DT                            | FILT275  |
| DWSS=(SAQPF/1.547)*SASPF*8.34*C1DT                                 | FILT276  |
| GO TO 25                                                           | FILT277  |
| C CCMPUTE RESIDENTIAL STRENGTHS ON THE BASIS OF INCOME AND GARBAGE | FILT278  |
| C GRINDERS.                                                        | FILT279  |
| 421 DWBOD=DWF*A1BOD*C1DT                                           | FILT280  |
| DWSS=DWF*A1SS*C1DT                                                 | FILT281  |
| IF(XINCOM.GT.15.) DWBOD=1.2*DWBOD                                  | FILT282  |
| IF (XINCOM.GT.15.)DWSS=1.2*DWSS                                    | FILT283  |
| IF(XINCOM.LT.7.)DWBOD =0.8*DWBOD                                   | FILT284  |
| IF(XINCOM.LT.7.) DWSS=0.8*DWSS                                     | FILT285  |
| DWBOD=DWBOD+(0.3*PCGG*DWBOD)/100.                                  | FILT286  |
| DWSS=DWSS+(0.3*PCGG*DWSS)/100.                                     | FILT287  |
| 25 DW1BOD=DWBOD*60./DT                                             | FILT288  |
| DW1SS=DWSS*60./DT                                                  | FILT289  |
| WRITE(6,6C5) KNUM,INPUT,DWF,QQ,QQDWF,KLAND,DW1BOD,DW1SS            | FILT290  |
| 605 FORMAT(' ',2I5,3F10.2,16,1F11.2,1F11.2)                        | FILT291  |
| C CCMPUTE TOTAL QUANTITIES IN SYSTEM                               | FILT292  |
| SMMDWF=SMMDWF+ DWF                                                 | FILT293  |
| SMMBOD=SMMBOD + DWBOD                                              | FILT294  |
| SMMSS=SMMSS + DWSS                                                 | FILT295  |
| SMMQQ=SMMQQ+CQ                                                     | FILT296  |
| SMTDWF=SMMQQ+SMMDWF                                                | FILT297  |
| IF(SMTDWF.LE.0.0) DWCCLI=0.0                                       | FILT298  |
| IF(SMTDWF.GT.0.0) DWCCLI=A1COLI*TOTPOP/(SMTDWF*2.447E+7)           | FILT299  |
|                                                                    | FILT300  |



```

      D2COLI=AICCLI*PCP/(60.*24.*60.)
C  D2COLI IS THE TOTAL DWF COLI FROM THE ASUB IN MPN/SEC
      INPUT=AIN(INPUT)
      WDWF(INPUT,1)=(CWBOD/DT)+WDWF(INPUT,1)
      WDWF(INPUT,2)=(DWSS/DT)+WDWF(INPUT,2)
      WDWF(INPUT,3)=D2COLI+WDWF(INPUT,3)
      IF(MSUBT)800,800,426
426  BODCGK=(SMBOD*1000000.)/(SMTDWF*DT*7.48*8.34)
      SSQNGC=(SMMSS*1000000.)/(SMTDWF*DT*7.48*8.34)
      WRITE(6,606) SMDWDF,SMMQ, SMTDWF,SMMBOD,SMMSS,TOTPOP,BODCON,SSQNGC
      .,DWCOLI
606  FORMAT('0',10X,'SUBTOTALS'//11X,3F10.2,6X,1F11.2,' LBS', 1F7.2,' L
      *BS',3F10.0,3X,1PE10.2//)
800  CONTINUE
      QDWF(INPLT)=QDWF(INPUT)+DWF
300  CONTINUE
C
C*****END OF TIMESTEP COMPUTATION
C
      BODCON=(SMBOD*1000000.)/(SMTDWF*DT*7.48*8.34)
      SSQNGC=(SMMSS*1000000.)/(SMTDWF*DT*7.48*8.34)
      WRITE(6,607) SMDWDF,SMMQ, SMTDWF,SMMBOD,SMMSS,TOTPOP,BODCON,
      .SSQNGC,DWCOLI
607  FORMAT('0',10X,'TOTALS'//11X,3F10.2,6X,1F11.2,' LBS', 1F7.2,' LBS
      *,3F10.0,3X,1PE10.2)
C*****
C
      IF(KASE.NE.1) GO TO 430
      IF(ADWF.EQ.0.) GO TO 430
      CF2= ADWF/SMTDWF
      WRITE(6,608) ADWF,SMTDWF,CF2
608  FORMAT(/' COMPARISON OF MEASURED AND CALCULATED TOTAL SEWAGE
      1 FLOW:  ADWF=',F6.2,' CFS  SMTDWF=',F6.2,' CFS'/' CORRECTION FACT
      *OR (CF2) OF',F5.2,' APPLIED TO THE DWF (QUANTITY AND QUALITY) AT F
      *ACH INLET')
      GO TO 431
430  CF2=1.
C  CORRECTION FACTOR (CF2) APPLIED TO THE DWF (QUANTITY AND QUALITY)
C  WHEN ACH IS MEASURED (KASE=1)
431  DO 432 I=1,NE
      WDWF(I,1)=WDWF(I,1)*CF2
      WDWF(I,2)=WDWF(I,2)*CF2
      WDWF(I,3)=WDWF(I,3)*CF2
432  QDWF(I)=QDWF(I)*CF2
C
C*****
C
C  DAILY AND HOURLY CORRECTION FACTORS APPLIED WITHIN TRANSPORT MODEL
      WRITE(6,609)
609  FORMAT('1',T40,'DAILY AND HOURLY CORRECTION FACTORS',/T51
      1,'FOR SEWAGE DATA'//,T35,'DAY',T50,'DVDWF',T60,'DVBOD',
      2T70,'DVSS',T80,'DVCOLI'//)
      WRITE(6,610) (1,DVDWF(I),DVBOD(I),DVSS(I),I=1,7)
610  FORMAT(' ',T30,I3,12X,3F10.3)
      WRITE(6,611)
611  FORMAT('C',T35,'HOUR'//)
      WRITE(6,612) (1,HVDWF(I),HVBOD(I),HVSS(I),HVCOLI(I),I=1,24)
612  FORMAT(' ',T30,I3,12X,4F10.3)
      RETURN
      END

```

```

SUBROUTINE FINDA(PS,AA)
C   CALCULATES THE FLOW AREA IN CONDUITS GIVEN THE FLOW RATE.
COMMON/TABLES/KDEPTH(25),KLASS(25),PSIMAX(15),ALFMAX(15),
1   NN(25),MM(25),ANORM(15,51),QNGRM(15,51),
2   DNORM(15,51),AFACT(15),RFACT(15)
COMMON A(160,2,2),Q(160,2,2),CPOLL(160,2,2,3),QMAX(150),
1   QFULL(160),AFULL(160),DXDT(160),C1(160),SLOPE(160),
2   DIST(160),GEOM1(160),ROUGH(160),NOE(160),NUE(160,3),
3   INUE(160,3),NTYPE(160),JR(160),NKLASS,NE,NDT,EPSIL,
4   TIME,DT,M,KFULL,N,NOS,NPOLL,NPRINT,ITER,
5   QCWF(160),IOLD(160),P1(160),RNOFF(160),QINFIL(160),
6   WDW(160,3),PLUTO(160,3),IR(160),P2(160),NIN(1000),
7   P5(160),P6(160),P7(160),SEP(160),DARKEL(160),
8   TITLE(40),NPE(20),NYN(20),NORDER(70),GEOM2(160),
9   GEOM3(160),P4(160),SCOUR(160),KSTORE(160)
COMMON BODIN(2,150),SSIN(2,150),BODOUT,SSOUT,COLIN(2,150),
* QINST,CCLST,QINSTL(2),QOUSTL(2),STORL(2),QOUTO(2),STORO(2),
* NSTOR,KSTOR,IPRINT(2),IPOL(2),IFLOOD(2),ICOST(2),DEPMAX(2),
* ATERM(2,11),AO2DT2(2,11),BDEPTH(2,11),BSTOR(2,11),COLJUT,
* DUMSTR(11),CUMDEP(11),
* K1STEP,VOLIN(2,150),VOLOUT(2,150),STOR,CUMIN(2),CUMOUT(2),
* SBOD(2),SSS(2),SCOL(2),
* ISTMGD(2),ISTTYP(2),ISTOUT(2),
* QPUMP(2),DSTART(2),DSTOP(2),
* DTON(2),STORMX(2),DTPUMP(2),DTMORE(2),STORF(2),APLAN(2),
* CLANC(2),CSTOR(2),CPS(2),CTOTAL(2),CPCUYD(2),CPACRE(2),
* LP,JP,LPREV(2),LABEL,DETENT(150),FRAC(150),OUT1(10,200)
AA = 0.0
IF (PS.EQ.C.C) RETURN
NT = NTYPE(M)
IF (KLASS(NT).EQ.1) GO TO 150
C   CONDUITS WITH TABULAR Q-A RELATIONSHIP.
MMM = MM(NT)
DALPHA = ANORM(NT,2)-ANORM(NT,1)
DO 110 I=1,MMM
IF (PS-QNGRM(NT,I+1)) 107,108,110
107 ALPHA = ANORM(NT,I)+(PS-QNGRM(NT,I))/(QNGRM(NT,I+1)-QNGRM(NT,I))*
1DALPHA
GO TO 111
108 ALPHA = ANORM(NT,I+1)
GO TO 111
110 CONTINUE
111 AA = ALPHA*AFULL(M)
RETURN
C   CONDUITS WITH FUNCTIONAL Q-A RELATIONSHIP.
150 C2 = -PS
ALPHA=C.2
CALL NEWTON(ALPHA,PS,0.0,C2,KFLAG)
IF (KFLAG.EQ.2) GO TO 155
AA = ALPHA*AFULL(M)
RETURN
155 WRITE (6,910) TIME,N,NOE(M),A(M,1,1)
AA = A(M,1,1)
RETURN
910 FORMAT (/ ' **WARNING: NEWTON UNABLE TO FIND AREA GIVEN FLOW. TIME=
1'F7.1,' , TIME STEP='13,' , EXT. ELE. NUM.='14,' , USE OLD UPSTREAM A
2REA='F6.2)
END

```

|    |                                                                   |         |
|----|-------------------------------------------------------------------|---------|
|    | SUBROUTINE FIRST                                                  | FIRS 1  |
| C  | ROUTINE PERFORMS INITIAL CALCULATIONS FOR EACH SEWER ELEMENT.     | FIRS 2  |
|    | DIMENSION Q1(160),Q0(160),WELL2(160),Q01(160),Q02(160),PUMP(160)  | FIRS 3  |
| 1  | , SURGE1(160),SURGE2(160)                                         | FIRS 4  |
|    | COMMON/NAMES/ NAME(4,25),GNO,YES,BLANK                            | FIRS 5  |
|    | COMMON/TABLES/KDEPTH(25),KLASS(25),PSIMAX(15),ALFMAX(15),         | FIRS 6  |
| 1  | AN(25),PM(25),ANORM(15,51),QNORM(15,51),                          | FIRS 7  |
| 2  | DNORM(15,51),AFACT(15),RFACT(15)                                  | FIRS 8  |
|    | COMMON A(160,2,2), Q(160,2,2), CPOLL(160,2,2,3), QMAX(160),       | FIRS 9  |
| 1  | QFULL(160), AFULL(160), DXDT(160),C1(160), SLOPE(160),            | FIRS 10 |
| 2  | DIST(160), GEOM1(160), ROUGH(160), NOE(160), NUF(160,3),          | FIRS 11 |
| 3  | INUE(160,3), NTYPE(160), JR(160), NKLASS, NE, NDT, EPSIL,         | FIRS 12 |
| 4  | TIME, DT, H, KEFULL, H, NOD, NPOLL, LPRINT, ITER,                 | FIRS 13 |
| 5  | QCHF(160), IOLD(160), P1(160), RNOFF(160), QINFIL(160),           | FIRS 14 |
| 6  | WCHF(160,3), PLUTO(160,3), IR(160), P2(160), NIN(1000),           | FIRS 15 |
| 7  | P5(160),P6(160),P7(160),SCF(160),BARREL(160),                     | FIRS 16 |
| 8  | TITLE(40), NPE(20), NYN(20), NORDER(70), GEOM2(160),              | FIRS 17 |
| 9  | GEOM3(160), P4(160),SCOUR(160), KSTURE(160)                       | FIRS 18 |
|    | EQUIVALENCE (Q0(1),Q(1,2,2)),(Q1(1),Q(1,1,2))                     | FIRS 19 |
|    | EQUIVALENCE (WELL2(1),ROUGH(1))                                   | FIRS 20 |
|    | EQUIVALENCE (Q01(1),QMAX(1)),(Q02(1),QFULL(1))                    | FIRS 21 |
|    | EQUIVALENCE (PUMP(1),DIST(1)),(SURGE1(1),P1(1)),(SURGE2(1),P2(1)) | FIRS 22 |
| C  | CALCULATE AREAS AND MAX. FLOWS USING MANNING EQN. AND GEOMETRY.   | FIRS 23 |
| C  | ALSO CALCULATE CERTAIN PARAMETERS USED IN ROUTING COMPUTATIONS.   | FIRS 24 |
| C  | NTYPE=1 DENOTES CIRCULAR CONDUIT.                                 | FIRS 25 |
| C  | NTYPE=2 DENOTES RECTANGULAR CONDUIT.                              | FIRS 26 |
| C  | NTYPE=3 DENOTES PHILLIPS STANDARD EGG-SHAPED CONDUIT.             | FIRS 27 |
| C  | NTYPE=4 DENOTES BOSTON HORSESHOE CONDUIT.                         | FIRS 28 |
| C  | NTYPE=5 DENOTES GOTHIC CONDUIT.                                   | FIRS 29 |
| C  | NTYPE=6 DENOTES CATENARY CONDUIT.                                 | FIRS 30 |
| C  | NTYPE=7 DENOTES LOUISVILLE SEMI-ELLIPTIC CONDUIT.                 | FIRS 31 |
| C  | NTYPE=8 DENOTES BASKET-HANDLE CONDUIT.                            | FIRS 32 |
| C  | NTYPE=9 DENOTES SEMI-CIRCULAR CONDUIT.                            | FIRS 33 |
| C  | NTYPE=10 DENOTES MODIFIED BASKET-HANDLE CONDUIT.                  | FIRS 34 |
| C  | NTYPE=11 DENOTES RECTANGULAR CONDUIT, TRIANGULAR BOTTOM.          | FIRS 35 |
| C  | NTYPE=12 DENOTES RECTANGULAR CONDUIT, ROUND BOTTOM.               | FIRS 36 |
| C  | NTYPE=13,14,15 ARE USER SUPPLIED CONDUIT SHAPES.                  | FIRS 37 |
| C  | NTYPE=16 DENOTES MANHOLE.                                         | FIRS 38 |
| C  | NTYPE=17 DENOTES LIFT STATION.                                    | FIRS 39 |
| C  | NTYPE=18 DENOTES TYPE 18 FLOW DIVIDER.                            | FIRS 40 |
| C  | NTYPE=19 DENOTES STORAGE UNIT.                                    | FIRS 41 |
| C  | NTYPE=20 DENOTES TYPE 20 FLOW DIVIDER.                            | FIRS 42 |
| C  | NTYPE=21 DENOTES TYPE 21 FLOW DIVIDER.                            | FIRS 43 |
| C  | NTYPE=22 DENOTES BACKWATER ELEMENT.                               | FIRS 44 |
| C  | NTYPE=23,24,25 ARE NOT YET PROGRAMMED.                            | FIRS 45 |
|    | K = 0                                                             | FIRS 46 |
|    | DO 200 I=1,NE                                                     | FIRS 47 |
|    | NT = NTYPE(I)                                                     | FIRS 48 |
|    | GO TO (10,20,10,10,10,10,10,10,10,100,110,120,10,10,10,190,160,   | FIRS 49 |
|    | 1 190,190,190,170,190,190,190,190), NT                            | FIRS 50 |
| C  | CALCULATIONS FOR SHAPES DESCRIBED BY TABULAR Q-A RELATIONSHIPS.   | FIRS 51 |
| C  | GEOM1(M)=MAXIMUM VERTICAL DIMENSION FOR THESE SHAPES.             | FIRS 52 |
| C  | UNITS OF FEET AND SECONDS ARE USED THROUGHOUT.                    | FIRS 53 |
| 10 | AFULL(M) = AFACT(NT)*GEOM1(M)*GEOM1(M)                            | FIRS 54 |
|    | P1(M) = 1.49/ROUGH(M)*AFULL(M)*(RFACT(NT)*GEOM1(M))**0.6666667    | FIRS 55 |
|    | P2(M) = GEOM1(M)/DIST(M)                                          | FIRS 56 |
|    | P4(M) = PSIMAX(NT)                                                | FIRS 57 |
|    | GO TO 150                                                         | FIRS 58 |
| C  | CALCULATIONS FOR RECTANGULAR CONDUIT.                             | FIRS 59 |
| 20 | AFULL(M) = GEOM1(M)*GEOM2(M)                                      | FIRS 60 |

```

      RH = AFULL(M)/(2.0*(GECM1(M)+GEOM2(M)))
      P1(M) = 1.49/ROUGH(M)*AFULL(M)*RH**0.6666667
      P5(M) = GECM1(M)/GECM2(M)
      P6(M) = -1.333333*P5(M)
      P7(M) = 2.0*P5(M)+2.0
      P2(M) = GECM1(M)/DIST(M)
      P4(M) = PSI(ALFMAX(NT))
      GO TO 150
C     CALCULATIONS FOR MODIFIED BASKET-HANDLE CONDUIT.
100  GEOM3(M) = GEOM1(M)*GECM2(M)
      P5(M) = 0.7853982*GEOM2(M)*GEOM2(M)
      AFULL(M) = GEOM3(M)+P5(M)/2.0
      RH = AFULL(M)/(GECM2(M)+2.0*GECM1(M))
      P1(M) = 1.49/ROUGH(M)*AFULL(M)*RH**0.6666667
      RH = GECM3(M)/(2.0*(GECM1(M)+GECM2(M)))
      P6(M) = 1.49/ROUGH(M)*GEOM3(M)*RH**0.6666667/P1(M)
      ALM = (GECM3(M)+P5(M)*(ALFMAX(1)-0.5))/AFULL(M)
      P7(M) = 2.0*GECM1(M)/GECM2(M)+2.0
      P2(M) = (GECM1(M)+GECM2(M)/2.0)/DIST(M)
      P4(M) = PSI(ALM)
      GO TO 150
C     CALCULATIONS FOR RECTANGULAR CONDUIT, TRIANGULAR BOTTOM.
110  AFULL(M) = GECM2(M)*(GEOM1(M)-GECM3(M)/2.0)
      P5(M) = 0.5*SIN(ATAN(2.0*GEOM3(M)/GEOM2(M)))
      P6(M) = GECM3(M)/P5(M)+2.0*(GEOM1(M)-GECM3(M))+GEOM2(M)
      RH = P5(M)/SQRT(2.0*GECM3(M)/GEOM2(M))*P6(M)
      P7(M) = RH**0.6666667/AFULL(M)**0.3333333
      P1(M) = 1.49/ROUGH(M)*AFULL(M)*(AFULL(M)/P6(M))**0.6666667
      P2(M) = GEOM1(M)/DIST(M)
      P4(M) = PSI(ALFMAX(NT))
      GO TO 150
C     CALCULATIONS FOR RECTANGULAR CONDUIT, ROUND BOTTOM.
120  P5(M) = 2.0*ARSIN(GEOM2(M)/2.0/GEOM3(M))
      P6(M) = GECM3(M)*GECM3(M)/2.0*(P5(M)-SIN(P5(M)))
      AFULL(M) = GEOM2(M)*GECM1(M)+P6(M)
      RH = AFULL(M)/(GECM3(M)*P5(M)+2.0*GEOM1(M)+GEOM2(M))
      P1(M) = 1.49/ROUGH(M)*AFULL(M)*RH**0.6666667
      AA = 3.1415965*GECM3(M)*GECM3(M)
      P7(M) = 1.49/ROUGH(M)*AA*(GECM3(M)/2.0)**0.6666667/P1(M)
      P2(M) = (GECM1(M)+GECM3(M)*(1.0-COS(P5(M)/2.0)))/DIST(M)
      P4(M) = PSI(ALFMAX(NT))
C     CALCULATIONS COMMON TO ALL CONDUITS.
C     CONVERT SLOPE FROM FT./100 FT. TO FT./FT.
150  SLOPE(M) = SLOPE(M)*0.01
      QFULL(M) = P1(M)*SQRT(SLOPE(M))
      QMAX(M) = P4(M)*QFULL(M)
      DXDT(M) = DIST(M)/DT
      C1(M) = DXDT(M)*AFULL(M)/QFULL(M)
C     DETERMINE IF FLOW IN CONDUIT IS SUPER-CRITICAL MOST OF TIME.
C     CRITERION IS DERIVED BY COMPARING NORMAL AND CRITICAL VELOCITIES.
C     BOTH VELOCITIES DEPEND UPON DEPTH OF FLOW.
C     FACTOR OF 0.3 CORRESPONDS TO CIRCULAR PIPE 95% FULL.
      AA=1.49/ROUGH(M)*SQRT(SLOPE(M)/32.2)*(P2(M)*DIST(M))**0.1666667*0.3
      SCF(M) = GND
      IF (AA.GE.1.0) SCF(M) = YES
      GO TO 200
C     CALCULATIONS FOR LIFT STATION.
C     INITIAL VOLUME IN WET WELL IS HALF THE CAPACITY.
160  WELL2(M) = 0.5*GEOM1(M)
      GO TO 150

```

|     |                                                                 |          |
|-----|-----------------------------------------------------------------|----------|
| C   | DETERMINE NUMBER AND ELEMENT NUMBERS OF FLOW DIVIDER TYPE 21'S. | FIRS121  |
| 170 | K = K+1                                                         | FIRS122  |
|     | IR(K) = M                                                       | FIRS123  |
|     | GO TO 190                                                       | FIRS124  |
| C   | INITIALIZE SURCHARGE VOLUME IN NGN-CONDUITS.                    | FIRS125  |
| 190 | SURGE1(M) = 0.0                                                 | FIRS126  |
|     | SURGE2(M) = 0.0                                                 | FIRS127  |
|     | SCF(M) = BLANK                                                  | FIRS128  |
| 200 | CONTINUE                                                        | FIRS129  |
|     | IF (K.EQ.C) RETURN                                              | FIRS130  |
| C   | INITIALIZE FLW SETTING FOR TYPE 21 FLOW DIVIDERS.               | FIRS131  |
|     | DO 220 I=1,K                                                    | FIRS132  |
|     | M= IR(I)                                                        | FIRS133  |
|     | L=GEOM3(M)                                                      | FIRS134  |
|     | L=NIN(L)                                                        | FIRS135  |
| 220 | GEOM1(M)=(QFULL(L)/2.0)*BARREL(L)                               | FIRS136A |
|     | RETURN                                                          | FIRS137  |
|     | END                                                             | FIRS138  |

|     |                                                           |           |
|-----|-----------------------------------------------------------|-----------|
|     | SUBROUTINE INFIL                                          | INFIL 1   |
|     | COMMON/TABLES/KDEPTH(25),KLASS(25),PSIMAX(15),ALFMAX(15), | INFIL 2   |
| 1   | NN(25),MM(25),ANORM(15,51),QNORM(15,51),                  | INFIL 3   |
| 2   | DNORM(15,51),AFAC(15),RFAC(15)                            | INFIL 4   |
|     | COMMON/NAMES/ NAME(4,25),GNU,YES,BLANK                    | INFIL 5   |
|     | COMMON A(160,2,2),Q(160,2,2),CPOLL(160,2,2,3),QMAX(160),  | INFIL 6   |
| 1   | QFULL(160),AFULL(160),DXDT(160),C1(160),SLOPE(160),       | INFIL 7   |
| 2   | DIST(160),GEOM1(160),ROUGH(160),NOE(160),NUE(160,3),      | INFIL 8   |
| 3   | INUE(160,3),NTYPE(160),JR(160),NKLASS,NE,NDT,EPSIL,       | INFIL 9   |
| 4   | TIME,DT,M,KFULL,N,NOS,NPOLL,NPRINT,ITER,                  | INFIL 10  |
| 5   | QCHF(160),IOLD(160),P1(160),RNOFF(160),QINFIL(160),       | INFIL 11  |
| 6   | WCWF(160,3),PIUTO(160,3),IR(160),P2(160),NIN(1000),       | INFIL 12  |
| 7   | P5(160),P6(160),P7(160),SCF(160),BARREL(160),             | INFIL 13  |
| 8   | TITLE(40),NPE(20),NYN(20),NORDER(70),GEOM2(160),          | INFIL 14  |
| 9   | GEOM3(160),P4(160),SCOUR(160),KSTORE(160)                 | INFIL 15  |
| C   | SUBROUTINE TO ESTIMATE AND ALLOCATE SEWER INFILTRATION    | INFIL 16  |
|     | DIMENSION NDD(12),NDXDAY(380)                             | INFIL 17  |
|     | READ(5,5C0) DINFIL,GINFIL,RINFIL                          | INFIL 18  |
|     | READ(5,5C1) NDYUD,RSMAX,ULEN                              | INFIL 23X |
| 15  | READ(5,51C) (NDD(I),I=1,12)                               | INFIL 25X |
|     | SINFIL=C.0                                                | INFIL 19  |
|     | IF(GINFIL)10,10,50                                        | INFIL 20  |
| 10  | SUMINF=DINFIL+RINFIL                                      | INFIL 21  |
|     | IF(SUMINF.EQ.0.0)WRITE(6,600)                             | INFIL 22  |
|     | IF(RSMAX.LE.C.0) GO TO 200                                | INFIL 24  |
|     | MFREZ=0                                                   | INFIL 26  |
|     | MLTBE=0                                                   | INFIL 27  |
|     | MLTEN=0                                                   | INFIL 28  |
| C   | PLACE MONTHLY VALUES AT MIDDLE OF MONTH                   | INFIL 29  |
|     | II=0                                                      | INFIL 30  |
|     | DO 100 I=1,12                                             | INFIL 31  |
|     | NDDAY=II+15                                               | INFIL 32  |
|     | NDXDAY(NDDAY)=NDD(I)                                      | INFIL 33  |
|     | II=II+30                                                  | INFIL 34  |
| 100 | CONTINUE                                                  | INFIL 35  |
| C   | INTERPOLATE FOR FIRST AND LAST 15 DAYS OF 'YEAR'          | INFIL 36  |
|     | NX=345                                                    | INFIL 37  |
|     | NY1=NDXDAY(345)                                           | INFIL 38  |
|     | NY2=NDXDAY(15)                                            | INFIL 39  |
|     | NX1=345                                                   | INFIL 40  |
|     | NX2=375                                                   | INFIL 41  |
|     | DO 110 I=1,30                                             | INFIL 42  |
|     | NX=NX+1                                                   | INFIL 43  |
|     | NY=((NY2-NY1)*(NX-NX1))/30 + NY1                          | INFIL 44  |
|     | NDXDAY(NX)=NY                                             | INFIL 45  |
| 110 | CONTINUE                                                  | INFIL 46  |
| C   | CONVERT LAST 15 DAYS TO FIRST 15                          | INFIL 47  |
|     | NX=360                                                    | INFIL 48  |
|     | DO 120 I=1,15                                             | INFIL 49  |
|     | NX=NX+1                                                   | INFIL 50  |
|     | NDXDAY(I)=NDXDAY(NX)                                      | INFIL 51  |
| 120 | CONTINUE                                                  | INFIL 52  |
| C   | INTERPOLATE FROM NDDAY=16 THRU NDDAY=344                  | INFIL 53  |
|     | K=1                                                       | INFIL 54  |
|     | NX=16                                                     | INFIL 55  |
|     | NX1=15                                                    | INFIL 56  |
|     | NX2=45                                                    | INFIL 57  |
| 130 | NY1=NDXDAY(NX1)                                           | INFIL 58  |
|     | NY2=NDXDAY(NX2)                                           | INFIL 59  |
|     | NY=((NY2-NY1)*(NX-NX1))/30 + NY1                          | INFIL 60  |

|                                                     |          |
|-----------------------------------------------------|----------|
| NDXDAY(IX)=NY                                       | INF1 61  |
| K=K+1                                               | INF1 62  |
| NX=NK+1                                             | INF1 63  |
| IF(K.LE.3C) GO TO 130                               | INF1 64  |
| NX1=NK1+30                                          | INF1 65  |
| NX2=NK2+30                                          | INF1 66  |
| K=1                                                 | INF1 67  |
| IF(NX2.LE.345) GO TO 130                            | INF1 68  |
| C DETERMINE BEGINNING OF FREEZING PERIOD(MFREZ)     | INF1 69  |
| DO 150 I=1,360                                      | INF1 70  |
| IF(NDXDAY(I).GT.750) GO TO 160                      | INF1 71  |
| 150 CONTINUE                                        | INF1 72  |
| 160 MFREZ=1                                         | INF1 73  |
| C IF STORM DAY IS PRIOR TO FREEZING, SET SINFIL=0.0 | INF1 74  |
| IF(NDYUD.LE.MFREZ) GO TO 200                        | INF1 75  |
| C STORM OCCURED AFTER FREEZING BEGAN                | INF1 76  |
| NTOT=0                                              | INF1 77  |
| DO 170 I=MFREZ,360                                  | INF1 78  |
| IF(NDXDAY(I).LE.750) GO TO 175                      | INF1 79  |
| 170 NTOT=NTOT+(NDXDAY(I)-750)                       | INF1 80  |
| 175 NAREAL=NTCT                                     | INF1 81  |
| MLTBE=I                                             | INF1 82  |
| NTOT=0                                              | INF1 83  |
| IF(NDYUD.LT.MLTBE) GO TO 200                        | INF1 84  |
| DO 180 I=MLTBE,360                                  | INF1 85  |
| NTOT=NTCT+(750-NDXDAY(I))                           | INF1 86  |
| IF(NTCT.GE.NAREAL) GO TO 185                        | INF1 87  |
| 180 CONTINUE                                        | INF1 88  |
| C IF EQUAL AREA NOT REACHED SET MLTEN=360           | INF1 89  |
| 185 MLTEN=I                                         | INF1 90  |
| IF(NDYUD.GE.MLTEN) GO TO 200                        | INF1 91  |
| C CALCULATE SINFIL                                  | INF1 92  |
| XMLTBE=FLOAT(MLTBE)                                 | INF1 93  |
| XNDYUD=FLOAT(NDYUD)                                 | INF1 94  |
| XMLTEN=FLOAT(MLTEN)                                 | INF1 95  |
| XXARG= ((XNDYUD-XMLTBE)/(XMLTEN-XMLTBE))*3.1416     | INF1 96  |
| SINFIL=RS*AX*SIN(XXARG)                             | INF1 97  |
| GO TO 40                                            | INF1 98  |
| 200 SINFIL=0.0                                      | INF1 99  |
| 40 QINF=RINFIL+SINFIL+DINFIL                        | INF1 100 |
| GO TO 60                                            | INF1 101 |
| 50 QINF=GINFIL                                      | INF1 102 |
| 60 CONTINUE                                         | INF1 103 |
| OPINF=0.0                                           | INF1 104 |
| IF(ULEN.LE.0.0)ULEN=6.0                             | INF1 105 |
| ATERM=SQRT(4.0*3.1416)/ULEN                         | INF1 106 |
| OPNFIL=C.0                                          | INF1 107 |
| DO 70 K=1,NE                                        | INF1 108 |
| M=JR(K)                                             | INF1 109 |
| NT=NTYPE(M)                                         | INF1 110 |
| IF(KLASS(NT).GT.2)GO TO 70                          | INF1 111 |
| OPINF=ATERM*SQRT(AFULL(M))*DIST(M)+OPINF            | INF1 112 |
| 70 CONTINUE                                         | INF1 113 |
| WRITE(6,615)                                        | INF1 114 |
| WRITE(6,620)DINFIL,GINFIL,SINFIL,RINFIL             | INF1 115 |
| WRITE(6,630)                                        | INF1 116 |
| DO 80 K=1,NE                                        | INF1 117 |
| M=JR(K)                                             | INF1 118 |
| NT=NTYPE(M)                                         | INF1 119 |
| IF(KLASS(NT).GT.2)GO TO 80                          | INF1 120 |

|                                                                        |         |
|------------------------------------------------------------------------|---------|
| OPNFIL=ATERN*SQFT(AFULL(M))*DIST(M)                                    | INF1121 |
| OP=OPNFIL/OPINF                                                        | INF1122 |
| QINFIL(M)=QINF*OP/448.8                                                | INF1123 |
| WRITE(6,640)NOE(M),QINFIL(M),OP                                        | INF1124 |
| 80 CONTINUE                                                            | INF1125 |
| 500 FORMAT(1CF8.1)                                                     | INF1126 |
| 501 FORMAT(15,6F8.1)                                                   | INF1127 |
| 510 FORMAT(16I5)                                                       | INF1128 |
| 600 FORMAT('1 NO FLOW ESTIMATES FOR DINFIL OR RINFIL WERE INCLUDED'//) | INF1129 |
| 615 FORMAT('C',20X,'TOTAL AREA INFILTRATION(IN GPM) DUE TO: '//,       | INF1130 |
| 110X,'BASE FLOW',5X,'GROUND WATER',11X,'MELT',5X,'RAIN'//)             | INF1131 |
| 620 FORMAT(10X,F10.4,5X,F10.4,9X,F10.4,9X,F10.4,//)                    | INF1132 |
| 630 FORMAT('C',20X,'APPORTIONED INFILTRATION'//,10X,'ELEMENT NO.',5X,  | INF1133 |
| 1*QINFIL(CFS)',15X,'OPNFIL'//)                                         | INF1134 |
| 640 FORMAT(' ',10X,15,12X,F10.3,11X,F8.4)                              | INF1135 |
| 660 FORMAT(' ',215)                                                    | INF1136 |
| RETURN                                                                 | INF1137 |
| END                                                                    | INF1138 |



```

SUBROUTINE INITAL INIT 1
C ROUTINE INITIALIZES FLOWS, AREAS, AND CONCENTRATIONS TO VALUES INIT 2
C CORRESPONDING TO DRY WEATHER FLOW PLUS INFILTRATION. INIT 3
C WHEN THERE ARE NO INITIAL VALUES OF DWF OR INFIL, VALUES REMAIN 0. INIT 4
COMMON/TABLES/KDEPTH(25),KCLASS(25),PSIMAX(15),ALFMAX(15), INIT 5
1 NN(25),MM(25),ANORM(15,51),QNORM(15,51), INIT 6
2 DNORM(15,51),AFAC(15),RFACT(15) INIT 7
COMMON/CRWF/ CVDWF(7),DVBCD(7),DVSS(7),HVDWF(24),HVBOD(24), INIT 8
1 HVSS(24),HVCOLI(24),KTNUM,KDAY,KHOUR,KMINS INIT 9
COMMON A(160,2,2),Q(160,2,2),CPOLL(160,2,2,3),QMAX(150), INIT 10
1 QFULL(160),AFULL(160),DXDT(160),C1(160),SLOPE(160), INIT 11
2 CIST(160),GEOM1(160),POUGH(160),NOF(160),NHF(160,3), INIT 12
3 INUE(160,3),NTYPE(160),JR(160),NKLASS,NE,NDT,EPSIL, INIT 13
4 TIME,DT,M,KFULL,K,NOS,NPOLL,NPRINT,ITER, INIT 14
5 QDWF(160),IOLD(160),P1(160),RNUFF(160),QINFIL(160), INIT 15
6 KDW(160,3),PLUTO(160,3),IR(160),P2(160),NIN(1000), INIT 16
7 P5(160),P6(160),P7(160),SCF(160),BARREL(160), INIT 17
8 TITLE(40),NPE(20),NYN(20),NORDER(70),GEOM2(160), INIT 18
9 GEOM3(160),P4(160),SCOUR(160),KSTORE(160) INIT 19
COMMON BGIN( 2,150),SSIN( 2,150),BQOUT,SSOUT,CULIN(2,150), INIT 20X
* QINST,COLST,CINSTL( 2),QOUSTL( 2),STORL( 2),QOUTO( 2),STORO( 2), INIT 21
* NSTOR,KSTOR,IPRINT( 2),IPOL( 2),IFLOOD( 2),ICUST( 2),DEPMAX( 2), INIT 22
* ATERM( 2,11),AOZDT2( 2,11),BDEPTH( 2,11),BSTOR( 2,11),COLOUT, INIT 23X
* DUMSTR(11),DUMDEP(11), INIT 24
* KISTEP,VOLIN( 2,150),VOLOUT( 2,150),STOR,CUMIN( 2),CUMOUT( 2), INIT 25
* SBOC( 2),SSS( 2),SCOL( 2), INIT 26X
* ISTMOC( 2),ISTTYP( 2),ISTOUT( 2), INIT 27
* QPUMP( 2),DSTART( 2),DSTOP( 2), INIT 28
* DTON( 2),STGRMX( 2),DTPUMP( 2),DTMORE( 2),STORF( 2),APLAN( 2), INIT 29
* CLAND( 2),CSTOR( 2),CPS( 2),CTOTAL( 2),CPCUYD( 2),CPACRE( 2), INIT 30
* LP,JP,LPREV( 2),LABEL,DETENT(150),FRAC(150),OUT1(10,200) INIT 31X
DIMENSION SUM2(6),QO(160),QI(160),QO1(160),QO2(160) INIT 32
EQUIVALENCE (QO1(1),QMAX(1)),(QO2(1),QFULL(1)) INIT 33
EQUIVALENCE (CO(1),Q(1,2,2)),(QI(1),Q(1,1,2)) INIT 34
WRITE (6,SC0) INIT 35
WRITE (6,SO2) INIT 36
DO 100 I=1,NE INIT 37
M = JR(I) INIT 38
C SUM UPSTREAM FLOWS AND POLLUTANT INFLOWS. INIT 39
SUM1 = C.C INIT 40
DO 10 K=1,NPOLL INIT 41
10 SUM2(K) = 0.0 INIT 42
DO 20 J=1,3 INIT 43
L=INUE(M,J) INIT 44
NTU=NTYPE(L) INIT 45
IF(NTU.LE.17.OR.NTU.GE.23) GO TO 18 INIT 46
KK=GEOM3(L) INIT 47
QQ=QO2(L) INIT 48
IF(INCE(M).EQ.KK)QQ=QO1(L) INIT 49
GO TO 19 INIT 50
18 CG = Q(L,2,1)*BARREL(L) INIT 51
19 SUM1=SUM1+QQ INIT 52
DO 20 K=1,NPOLL INIT 53
20 SUM2(K)=SUM2(K)+CPOLL(L,2,1,K)*QQ INIT 54
SUM1 = SUM1+QINFIL(M)+QDWF(M)*DVDWF(KDAY)*HVDWF(KHOUR) INIT 55
C(M,1,1) = SUM1/BARREL(M) INIT 56
Q(M,2,1) = Q(M,1,1) INIT 57
IF(Q(M,1,1).LT.0.0) Q(M,1,1)=0.0 INIT 58
IF(Q(M,2,1).LT.C.0) Q(M,2,1)=0.0 INIT 59
IF (SUM1.EQ.0.0) GO TO 40 INIT 60

```

```

DO 30 K=1,NPOLL
  IF(K.EQ.1) SUM2(K)=SUM2(K)+WDWF(M,K)*DVBUD(KDAY)*HVBUD(KHOUR)*
  *DVDWF(KDAY)*HVDWF(KHOUR)
  IF(K.EQ.2) SUM2(K)=SUM2(K)+WDWF(M,K)*DVSS(KDAY)*HVSS(KHOUR)*
  *DVDWF(KDAY)*HVDWF(KHOUR)
  IF (K.EQ.3) SUM2(K)=SUM2(K)+WDWF(M,K)*1.0*HVCOLI(KHOUR)*DVDWF(KDAY
  *)*HVDWF(KHOUR)
  CPOLL(M,1,1,K) = SUM2(K)/SUM1
30 CPOLL(M,2,1,K) = CPOLL(M,1,1,K)
40 NT = NTYPE(M)
  IF (NT.EQ.22) GO TO 44
  IF(NT.LT.18.OR.NT.GT.22) GO TO 45
C  ASSUME ALL DWF IS NON-DIVERTED IN A FLOW DIVIDER TYPE ELEMENT.
  Q01(M)=SUM1
  Q02(M)=C.C
  GO TO 45
44 Q01(M) = 0.0
  Q02(M) = SUM1
45 IF (KLASS(NT).EQ.3) GO TO 100
C  DETERMINE INITIAL FLOW AREA FOR CONDUITS.
  PS = Q(M,1,1)/QFULL(M)
  CALL FINCA(PS,A(M,1,1))
  A(M,2,1) = A(M,1,1)
100 WRITE (6,901) NGE(M),NT,SUM1,A(M,1,1),(CPOLL(M,1,1,K),K=1,NPOLL)
900 FORMAT ('ELEMENT FLOWS, AREAS, AND CONCENTRATIONS ARE INITIALIZED
1 TO DRY WEATHER FLOW AND INFILTRATION VALUES.')
```

|  |         |
|--|---------|
|  | INIT 61 |
|  | INIT 62 |
|  | INIT 63 |
|  | INIT 64 |
|  | INIT 65 |
|  | INIT 66 |
|  | INIT 67 |
|  | INIT 68 |
|  | INIT 69 |
|  | INIT 70 |
|  | INIT 71 |
|  | INIT 72 |
|  | INIT 73 |
|  | INIT 74 |
|  | INIT 75 |
|  | INIT 76 |
|  | INIT 77 |
|  | INIT 78 |
|  | INIT 79 |
|  | INIT 80 |
|  | INIT 81 |
|  | INIT 82 |
|  | INIT 83 |
|  | INIT 84 |
|  | INIT 85 |
|  | INIT 86 |
|  | INIT 87 |
|  | INIT 88 |
|  | INIT 89 |
|  | INIT 90 |
|  | INIT 91 |

```

  1  CCAC4      CONC5      CONC6')
  RETURN
END
```

```

SUBROUTINE NEWTON(ALPHA,PS,C11,C2,KFLAG) NEWT 1
C NEWTON-RAPHSON ITERATION ASSUMING FUNCTIONAL FORM FOR Q-A CURVE. NEWT 2
  DIMENSION QI(160),QO(160) NEWT 3
  COMMON/TABLES/KDEPTH(25),KLASS(25),PSIMAX(15),ALFMAX(15), NEWT 4
1      NN(25),MM(25),ANORM(15,51),QNORM(15,51), NEWT 5
2      DNORM(15,51),AFAC(15),RFACT(15) NEWT 6
  COMMON A(160,2,2),Q(160,2,2),CPOLL(160,2,2,3),QMAX(160), NEWT 7
1      CFULL(160),AFULL(160),DXDT(160),C1(160),SLOPE(160), NEWT 8
2      DIST(160),GEOM1(160),ROUGH(160),NOE(160),NUE(160,3), NEWT 9
3      INUE(160,3),NTYPE(160),JR(160),NKLASS,NE,NDT, EPSIL, NEWT 10
4      TIME,DT,M,KFULL,N,NOS,NPOLL,NPRINT,ITER, NEWT 11
5      QDWF(160),IOLD(160),P1(160),RNOFF(160),QINFIL(160), NEWT 12
6      WDW(160,3),PLUTU(160,3),IK(160),P2(160),NIN(1000), NEWT 13
7      P5(160),P6(160),P7(160),SCF(160),BARREL(160), NEWT 14
8      TITLE(40),NPE(20),NYN(20),NURDER(70),GEUM2(160), NEWT 15
9      GECH3(160),P4(160),SCOUR(160),KSTORE(160) NEWT 16
  COMMON BODIN( 2,150),SSIN( 2,150),BODOUT,SSOUT,COLIN(2,150), NEWT 17X
* QINST,COLST,QINSTL( 2),QUUSTL( 2),STORL( 2),QUUTO( 2),STORO( 2), NEWT 18
* NSTOR,KSTCR,IPRINT( 2),IPOL( 2),IFLOOD( 2),ICOST( 2),DEPMAX( 2), NEWT 19
* ATERM( 2,11),AO2DT2( 2,11),BDEPTH( 2,11),BSTOR( 2,11),CULOUT, NEWT 20X
* DUMSTR(11),DUMDEP(11), NEWT 21
* KSTEP,VOLIN( 2,150),VOLOUT( 2,150),STOR,CUMIN( 2),CUMOUT( 2), NEWT 22
* SBOD( 2),SSS( 2),SCOL( 2), NEWT 23X
* ISTMOD( 2),ISTTYP( 2),ISTOUT( 2), NEWT 24
* QPUMP( 2),DSTART( 2),DSTUP( 2), NEWT 25
* DTON( 2),STORMX( 2),DTPUMP( 2),DTMORE( 2),STORF( 2),APLAN( 2), NEWT 26
* CLAND( 2),CSTOR( 2),CPS( 2),CTOTAL( 2),CPCJYD( 2),CPACRE( 2), NEWT 27
* LP,JP,LPREV( 2),LABEL,DETENT(150),FRAC(150),OUT1(10,200) NEWT 28X
  EQUIVALENCE (CC(1),Q(1,2,2)),(QI(1),Q(1,1,2)) NEWT 29
C FIRST GUESS FOR ALPHA ASSIGNED IN CALLING PROGRAM. NEWT 30
  KFLAG = 1 NEWT 31
  HELP = 0.0 NEWT 32
  ICHK = 0 NEWT 33
  I = 0 NEWT 34
10 I=I+1 NEWT 35
  PS = PSI(ALPHA) NEWT 36
  D=(PS+C11*ALPHA+C2)/(DPSI(ALPHA)+C11) NEWT 37
  IF (ABS(D).LE.EPSIL) GO TO 20 NEWT 38
  ALPHA = ALPHA-D NEWT 39
  IF (ALPHA.LT.0.0.OR.ALPHA.GT.1.0.OR.ICHK.EQ.1) GO TO 15 NEWT 40
  IF (I.LE.20) GO TO 10 NEWT 41
  ALPHA = ALPHA+D/2.0 NEWT 42
  ICHK = 1 NEWT 43
  GO TO 10 NEWT 44
C IF NEEC BE, ASSIGN NEW VALUES TO ALPHA BY MARCHING ALONG ABCISSA. NEWT 45
15 ALPHA = HELP NEWT 46
  HELP = HELP+0.05 NEWT 47
  I = 0 NEWT 48
  ICHK = 0 NEWT 49
  IF (HELP.LE.1.05) GO TO 10 NEWT 50
C ITERATION DOES NOT CONVERGE. RETURN TO MAIN FOR ERROR MESSAGE. NEWT 51
  KFLAG = 2 NEWT 52
  RETURN NEWT 53
20 IF (ALPHA.GE.0.0.AND.ALPHA.LE.1.0) RETURN NEWT 54
  GO TO 15 NEWT 55
  END NEWT 56

```

|                  |                                                                    |          |
|------------------|--------------------------------------------------------------------|----------|
| SUBROUTINE PRINT |                                                                    | PRIN 1   |
| C                |                                                                    | PRIN 2   |
| C                | *****THIS IS A NEW SUBROUTINE WHICH WILL REPLACE THE OLD PRINT     | PRIN 3   |
| C                | WHICH EXISTED AT THE TIME OF THE MARCH MEETING                     | PRIN 4   |
| C                |                                                                    | PRIN 5   |
|                  | COMMON /TAPES/ INCNT,ICUTCT,JIN(10),JOUT(10),NSCRAT(5)             | PRIN 6   |
|                  | COMMON A(160,2,2),Q(160,2,2),CPOLL(160,2,2,3),QMAX(160),           | PRIN 6A  |
| 1                | CFULL(160),AFULL(160),DXDT(160),C1(160),SLOPE(160),                | PRIN 6B  |
| 2                | DIST(160),GEOM1(160),ROUGH(160),NOE(160),NUE(160,3),               | PRIN 6C  |
| 3                | INUE(160,3),NTYPE(160),JR(160),NKLASS,NE,NDT,EPSIL,                | PRIN 6D  |
| 4                | TIME,CT,M,KFULL,N,NOS,NPOLL,NPRINT,ITER,                           | PRIN 6E  |
| 5                | QCHF(160),IOLD(160),P1(160),RNOFF(160),QINFIL(160),                | PRIN 6F  |
| 6                | WLWF(160,3),PLUTU(160,3),IR(160),P2(160),NIN(1000),                | PRIN 6G  |
| 7                | P5(160),P6(160),P7(160),SCF(160),BARREL(160),                      | PRIN 6H  |
| 8                | TITLE(40),NPE(20),NYN(20),NORDER(70),GEOM2(160),                   | PRIN 6I  |
| 9                | GEOM3(160),P4(160),SCOUR(160),KSTORE(160)                          | PRIN 6J  |
|                  | COMMON BODIN( 2,150),SSIN( 2,150),BODOUT,SSOUT,COLIN(2,150),       | PRIN 6K  |
|                  | * QINST,CCLST,QINSTL( 2),QOUSTL( 2),STORL( 2),QOUTO( 2),STORO( 2), | PRIN 6L  |
|                  | * NSTOR,KSTCR,IPRINT( 2),IPOL( 2),IFLUOD( 2),ICUST( 2),DEPMAX( 2), | PRIN 6M  |
|                  | * ATERM( 2,11),AO2DT2( 2,11),BDEPTH( 2,11),BSTOR( 2,11),COLOUT,    | PRIN 6N  |
|                  | * DUMSTR(11),CUMDEP(11),                                           | PRIN 6O  |
|                  | * KTSTEP,VOLIN( 2,150),VOLOUT( 2,150),STOR,CUMIN( 2),CUMOUT( 2),   | PRIN 6P  |
|                  | * SBOD( 2),SSS( 2),SCOL( 2),                                       | PRIN 6Q  |
|                  | * ISTMCD( 2),ISTTYP( 2),ISTOUT( 2),                                | PRIN 6R  |
|                  | * QPUMP( 2),DSTART( 2),DSTOP( 2),                                  | PRIN 6S  |
|                  | * DTION( 2),STORMX( 2),DTPUMP( 2),DTMORE( 2),STORF( 2),APLAN( 2),  | PRIN 6T  |
|                  | * CLAND( 2),CSTOR( 2),CPS( 2),CTOTAL( 2),CPCUYD( 2),CPACRE( 2),    | PRIN 6U  |
|                  | * LP,JP,LPREV( 2),LABEL,DETENT(150),FRAC(150),OUT1(10,200)         | PRIN 6V  |
|                  | COMMON /XX/ OUT2(10,200,4),PP(4)                                   | PRIN 7X  |
|                  | II=0                                                               | PRIN 8   |
| 5                | II=II+1                                                            | PRIN 9   |
|                  | NTX=NSCRAT(II)                                                     | PRIN 10  |
|                  | REWIND NTX                                                         | PRIN 11  |
|                  | DO 10 J=1, 10                                                      | PRIN 12  |
|                  | NYN(J)=0                                                           | PRIN 13  |
|                  | DO 10 N=1,200                                                      | PRIN 14  |
|                  | OUT1(J,N)=0.0                                                      | PRIN 15  |
|                  | DO 10 I=1,4                                                        | PRIN 16  |
|                  | OUT2(J,N,I)=0.0                                                    | PRIN 17  |
|                  | PP(I)=0.0                                                          | PRIN 18  |
| 10               | CONTINUE                                                           | PRIN 19  |
|                  | READ(NTX) NDT,NAYN,NPOLL,NNPE,DT,JPRINT,JPLOT                      | PRIN 20  |
|                  | READ(NTX) (AYN(I),I=1,NAYN)                                        | PRIN 21  |
| 900              | FORMAT(10I5)                                                       | PRIN 22  |
|                  | DO 20 N=1,NDT                                                      | PRIN 23  |
|                  | DO 20 J=1,NAYN                                                     | PRIN 24  |
|                  | READ(NTX) CQ,(PP(I),I=1,NPOLL)                                     | PRIN 25X |
|                  | OUT1(J,N)=CQ                                                       | PRIN 26X |
|                  | DO 15 JJ=1,NPOLL                                                   | PRIN 27  |
| 15               | OUT2(J,N,JJ)=PP(JJ)                                                | PRIN 28  |
| 20               | CONTINUE                                                           | PRIN 29  |
|                  | IF(II.GT.1) GO TO 29                                               | PRIN 30  |
| C                | PRINT INFLOWS                                                      | PRIN 30A |
|                  | WRITE(6,921)                                                       | PRIN 31  |
|                  | WRITE(6,911)(I,I=1,10)                                             | PRIN 32  |
|                  | DO 216 I= 1,NAYN                                                   | PRIN 33  |
| 216              | WRITE(6,912) AYN(I),(OUT1(I,J),J=1,NDT)                            | PRIN 34  |
|                  | WRITE(6,922)                                                       | PRIN 35  |
|                  | WRITE(6,911)(I,I=1,10)                                             | PRIN 36  |
|                  | DO 217 I=1,NAYN                                                    | PRIN 37  |

|                                                           |          |
|-----------------------------------------------------------|----------|
| DO 217 IP=1,NPOLL                                         | PRIN 38  |
| GO TO (30,32,36,34), IP                                   | PRIN 39  |
| 30 WRITE(6,960)                                           | PRIN 40  |
| GO TO 160                                                 | PRIN 41  |
| 32 WRITE(6,961)                                           | PRIN 42  |
| GO TO 160                                                 | PRIN 43  |
| 34 WRITE(6,962)                                           | PRIN 44  |
| GO TO 160                                                 | PRIN 45  |
| 36 WRITE(6,963)                                           | PRIN 46  |
| 160 CONTINUE                                              | PRIN 47  |
| IF(IP.NE.3)WRITE(6,912) NYN(1),(OUT2(I,J,IP),J=1,NDT)     | PRIN 48  |
| IF(IP.EQ.2)WRITE(6,967) NYN(1),(OUT2(I,J,IP),J=1,NDT)     | PRIN 49  |
| 217 CONTINUE                                              | PRIN 50  |
| IF(II.LE.1) GO TO 5                                       | PRIN 51  |
| C PRINT OUTFLOWS                                          | PRIN 51A |
| 29 WRITE(6,923)                                           | PRIN 52  |
| WRITE(6,911)(I,I=1,10)                                    | PRIN 53  |
| DO 218 I=1,NNPE                                           | PRIN 54  |
| 218 WRITE(6,912) NYN(I),(OUT1(I,J),J=1,NDT)               | PRIN 55C |
| 30C WRITE(6,924)                                          | PRIN 67C |
| WRITE(6,911)(I,I=1,10)                                    | PRIN 63  |
| DO 219 I=1,NNPE                                           | PRIN 69  |
| DO 219 IP=1,NPOLL                                         | PRIN 70  |
| GO TO (40,42,46,44),IP                                    | PRIN 71  |
| 40 WRITE(6,960)                                           | PRIN 72  |
| GO TO 161                                                 | PRIN 73  |
| 42 WRITE(6,961)                                           | PRIN 74  |
| GO TO 161                                                 | PRIN 75  |
| 44 WRITE(6,962)                                           | PRIN 76  |
| GO TO 161                                                 | PRIN 77  |
| 46 WRITE(6,963)                                           | PRIN 78  |
| 161 CONTINUE                                              | PRIN 79  |
| IF(IP.NE.3) WRITE(6,912) NYN(1),(OUT2(I,J,IP),J=1,NDT)    | PRIN 80  |
| IF(IP.EQ.3) WRITE(6,967) NYN(1),(OUT2(I,J,IP),J=1,NDT)    | PRIN 81  |
| 219 CONTINUE                                              | PRIN 82  |
| C FIND TCTAL OUTPUT OF SS AND BOD IN POUNDS               | PRIN 83  |
| TSSOUT=0.0                                                | PRIN 84  |
| TBODOT=0.0                                                | PRIN 85  |
| DO 1000 J=1,NDT                                           | PRIN 86  |
| TSSOUT=TSSOUT+OUT2(NNPE,J,2)*DT/60.                       | PRIN 87  |
| TBODOT=TBCDOT+OUT2(NNPE,J,1)*DT/60.                       | PRIN 88  |
| 1000 CONTINUE                                             | PRIN 89C |
| WRITE(6,964)                                              | PRIN104C |
| WRITE(6,911)(I,I=1,10)                                    | PRIN105  |
| DO 236 I=1,NNPE                                           | PRIN106  |
| DO 236 IP=1,NPOLL                                         | PRIN107  |
| GO TO (50,52,56,54),IP                                    | PRIN108  |
| 50 WRITE(6,968)                                           | PRIN109  |
| GO TO 234                                                 | PRIN110  |
| 52 WRITE(6,969)                                           | PRIN111  |
| GO TO 234                                                 | PRIN112  |
| 54 WRITE(6,962)                                           | PRIN113  |
| GO TO 234                                                 | PRIN114  |
| 56 CONTINUE                                               | PRIN115  |
| IF(IP.EQ.3) WRITE(6,966)                                  | PRIN116  |
| 234 CONTINUE                                              | PRIN117  |
| DO 235 J=1,NDT                                            | PRIN118  |
| IF(OUT1(I,J).EQ.0.0) GO TO 237                            | PRIN119  |
| IF(IP.NE.3) OUT2(I,J,IP)=OUT2(I,J,IP)*267.5/OUT1(I,J)     | PRIN120  |
| IF(IP.EQ.3) OUT2(I,J,IP)=OUT2(I,J,IP)/(OUT1(I,J)*17040.0) | PRIN121  |

|                                                                        |         |
|------------------------------------------------------------------------|---------|
| GO TO 238                                                              | PRIN122 |
| 237 OUT2(I,J,IP)=0.0                                                   | PRIN123 |
| 238 CONTINUE                                                           | PRIN124 |
| 235 CONTINUE                                                           | PRIN125 |
| IF(IP.NE.3) WRITE(6,912)NYN(I),(OUT2(I,J,IP),J=1,NDT)                  | PRIN126 |
| IF(IP.EQ.3) WRITE (6,967)NYN(I),(OUT2(I,J,IP),J=1,NDT)                 | PRIN127 |
| 236 CONTINUE                                                           | PRIN128 |
| WRITE(06,965) NYN(NNPE),TSSOUT,NYN(NNPE),TBODOT                        | PRIN129 |
| 965 FORMAT('1','0',5X,'TOTAL POUNDS OF SUSPENDED SOLIDS OUTPUT FROM EL | PRIN130 |
| *EMENT ',I4,' =',F10.2//',5X,'TOTAL POUNDS OF FIVE-DAY BOD OUTPUT      | PRIN131 |
| * FROM ELEMENT ',I4,' =',F10.2//')                                     | PRIN132 |
| 311 RETURN                                                             | PRIN133 |
| 964 FORMAT(/42X,'SELECTED OUTFLOW POLLUTOGRAPHS')                      | PRIN134 |
| 911 FORMAT(' EXTERNAL'/' ELEMENT TIME STEP'/' NUMBER',3X,10(16,4X))    | PRIN135 |
| 912 FORMAT (/16,4X,10F10.3,/, (10X,10F10.3))                           | PRIN136 |
| 921 FORMAT(/43X,'SELECTED INLET HYDROGRAPHS - CFS')                    | PRIN137 |
| 922 FORMAT(/43X,'SELECTED INLET POLLUTOGRAPHS')                        | PRIN138 |
| 923 FORMAT(/42X,'SELECTED OUTFLOW HYDROGRAPHS - CFS')                  | PRIN139 |
| 924 FORMAT(/42X,'SELECTED OUTFLOW POLLUTOGRAPHS')                      | PRIN140 |
| 955 FORMAT('C',40X,'POLLUTANT',15)                                     | PRIN141 |
| 960 FORMAT('C',40X,'*** BOD IN LBS/MIN ***')                           | PRIN142 |
| 961 FORMAT('C',40X,'*** SUSPENDED SOLIDS IN LBS/MIN ***')              | PRIN143 |
| 962 FORMAT('O',40X,'*** DC ***')                                       | PRIN144 |
| 963 FORMAT('C',40X,'*** COLIFORM IN MPN/MIN ***')                      | PRIN145 |
| 966 FORMAT('C',40X,'***COLIFORM IN MPN/100ML ***')                     | PRIN146 |
| 967 FORMAT(/16,4X,10F10.2,/, (10X,10F10.2))                            | PRIN147 |
| 968 FORMAT('C',40X,'*** BOD IN MG/L ***')                              | PRIN148 |
| 969 FORMAT('O',40X,'*** SUSPENDED SOLIDS IN MG/L ***')                 | PRIN149 |
| END                                                                    | PRIN150 |

```

FUNCTION PSI(ALPHA)
C FINDS Q/CFULL (PSI) GIVEN A/AFULL (ALPHA) FOR FUNCTIONAL Q-A CURVE
COMMON /PSICPS/ AA,AB,D1,D2,AAA,CATH,ALF,I,R
COMMON/TABLES/KDEPTH(25),KCLASS(25),PSIMAX(15),ALFMAX(15),
1 NN(25),MM(25),ANORM(15,51),QNORM(15,51),
2 DNORM(15,51),AFACT(15),RFACT(15)
COMMON A(160,2,2),Q(160,2,2),CPULL(160,2,2,3),QMAX(160),
1 QFULL(160),AFULL(160),DXDT(160),C1(160),SLOPE(160),
2 DIST(160),GECM1(160),ROUGH(160),NOE(160),NUE(160,3),
3 INUE(160,3),NTYPE(160),JR(160),NKLASS,NE,NDT,EPSIL,
4 TIME,DT,M,KFULL,N,NDS,NPOLL,NPRINT,ITER,
5 QCHF(160),IOLD(160),P1(160),RNOFF(160),QINFIL(160),
6 WCHF(160,3),PLUJ(160,3),IK(160),P2(160),NIN(1000),
7 P5(160),P6(160),P7(160),SCF(160),BARREL(160),
8 TITLE(40),NPE(20),NYN(20),NORDER(70),GEOM2(160),
9 GECM3(160),P4(160),SCOUR(160),KSTORE(160)
COMMON BODIN( 2,150),SSIN( 2,150),BUDJUT,SSJUT,COLIN(2,150),
* QINST,COLST,QINSTL( 2),QOUSTL( 2),STORL( 2),QOUTO( 2),STORO( 2),
* NSTOR,KSTOR,IPRINT( 2),IPOL( 2),IFLOOD( 2),ICOST( 2),DEPMAX( 2),
* ATERF( 2,11),AO2DT2( 2,11),BDEPTH( 2,11),BSTOR( 2,11),COLOUT,
* DUMSTR(11),DUMDEP(11),
* KTSTEP,VOLIN( 2,150),VGLOUT( 2,150),STUR,CUMIN( 2),CUMOUT( 2),
* SBOC( 2),SSS( 2),SCCL( 2),
* ISTMOG( 2),ISTTYP( 2),ISTOUT( 2),
* QPUMP( 2),DSTART( 2),DSTUP( 2),
* DTUN( 2),STORMX( 2),CTPUMP( 2),DTMORE( 2),STORF( 2),APLAN( 2),
* CLANC( 2),CSTCR( 2),CPS( 2),CTOTAL( 2),CPCUYD( 2),CPACRE( 2),
* LP,IP,LPREV( 2),LABEL,DETENT(150),FRAC(150),OUTI(10,200)
PSI = 0.0
IF(ALPHA.LE.0.0) RETURN
NT = NTYPE(M)
IF (NT.EQ.2) GO TO 20
IF (NT.EQ.10) GO TO 100
IF (NT.EQ.11) GO TO 110
IF (NT.EQ.12) GO TO 120
C INCLUDE TABULAR PSI CALC. IN CASE PSI IS CALLED BY KCLASS=2 CONDUIT
DALPHA = ANORM(NT,2)-ANORM(NT,1)
I = ALPHA/DALPHA+1.0
PSI = QNORM(NT,I)+(QNORM(NT,I+1)-QNORM(NT,I))/DALPHA*(ALPHA-
1 ANORM(NT,1))
RETURN
C SPECIAL FUNCTIONAL FORM FOR RECTANGULAR CONDUITS.
20 R = P5(M)
IF (ALPHA.GT.ALFMAX(NT)) GO TO 25
AAA = 2.C*R*ALPHA+1.0
CATH = (ALPHA*P7(M)/AAA)**0.6666667
PSI = ALPHA*CATH
RETURN
25 PSI = P4(M)+(ALPHA-ALFMAX(NT))*(1.0-P4(M))/(1.0-ALFMAX(NT))
RETURN
C FUNCTIONAL FORM FOR MODIFIED BASKET-HANDLE.
100 AA = ALPHA*AFULL(M)
IF (AA.GT.GECM3(M)) GO TO 105
ALF = AA/GECM3(M)
R = GECM1(M)/GECM2(M)
AAA = 2.C*ALF*R+1.0
CATH = (ALF*P7(M)/AAA)**0.6666667*P6(M)
PSI = ALF*CATH
RETURN
105 RH = RACH(AA)

```

|                                                                  |        |
|------------------------------------------------------------------|--------|
| PSI = 1.49/ROUGH(M)*AA*RH**0.6666667/P1(M)                       | PSI 61 |
| RETURN                                                           | PSI 62 |
| C FUNCTIONAL FORM FOR RECTANGULAR, TRIANGULAR BOTTOM.            | PSI 63 |
| 110 AB = GECM3(M)*GECM2(M)/2.0                                   | PSI 64 |
| AA = ALPHA*AFULL(M)                                              | PSI 65 |
| IF (AA.GT.AB) GO TO 115                                          | PSI 65 |
| PSI = P7(M)*ALPHA**1.333333                                      | PSI 67 |
| RETURN                                                           | PSI 68 |
| 115 IF (ALPHA.GT.ALFXAX(NT)) GO TO 25                            | PSI 69 |
| AAA = GECM3(M)/P5(M)-GECM3(M)+(2.0*GEOM1(M)-GECM3(M))*ALPHA      | PSI 70 |
| CATH = (ALPHA*P6(M)/AAA)**0.6666667                              | PSI 71 |
| PSI = ALPHA*CATH                                                 | PSI 72 |
| RETURN                                                           | PSI 73 |
| C FUNCTIONAL FORM FOR RECTANGULAR, ROUND BOTTOM.                 | PSI 74 |
| 120 AA = ALPHA*AFULL(M)                                          | PSI 75 |
| IF (AA.GT.P6(M)) GO TO 125                                       | PSI 76 |
| ALF = ALPHA*AFULL(M)/(3.1415965*GEOM3(M)*GEOM3(M))               | PSI 77 |
| I = ALF/0.02+1.0                                                 | PSI 78 |
| PSI = QACRM(1,I)+(QNORM(1,I+1)-QNORM(1,I))/0.02*(ALF-ANORM(1,I)) | PSI 79 |
| PSI = PSI*P7(M)                                                  | PSI 80 |
| RETURN                                                           | PSI 81 |
| 125 IF (ALPHA.GT.ALFXAX(NT)) GO TO 25                            | PSI 82 |
| D1 = GECM3(M)*P5(M)+2.0*GEOM1(M)+GEOM2(M)                        | PSI 83 |
| D2 = GECM3(M)*P5(M)+2.0/GEOM2(M)*(AFULL(M)*ALPHA-P6(M))          | PSI 84 |
| CATH = (ALPHA*D1/D2)**0.6666667                                  | PSI 85 |
| PSI = ALPHA*CATH                                                 | PSI 86 |
| RETURN                                                           | PSI 87 |
| END                                                              | PSI 88 |



|     |                                                                    |      |     |
|-----|--------------------------------------------------------------------|------|-----|
|     | SUBROUTINE CUAL(DUMY2,DUMY3,DUMY4,DUMY5)                           | QUAL | 1   |
| C   | ROUTINE SUBPROGRAM FOR POLLUTANTS MOVING THROUGH SEWER ELEMENTS    | QUAL | 2   |
| C   | COMPLETE MIXING ASSUMED IN ALL ELEMENTS SURFACE AND SUB-           | QUAL | 3   |
| C   | SURFACE INPUTS(PIPEFLOW,GROUNDWATER,SEWAGE, AND CATCHBASIN OUTFLOW | QUAL | 4   |
| C   | MIXING VOLUME IS THE DIFFERENTIATING FACTOR THAT AFFECTS QUALITY   | QUAL | 5   |
| C   | CHANGE FROM ONE ELEMENT TO THE NEXT                                | QUAL | 6   |
|     | DIMENSION SURGE1(160),SURGE2(160),WELL1(160),WELL2(160)            | QUAL | 7   |
|     | 1,QO(160),QI(160),QO1(160),QO2(160),PUMP(160)                      | QUAL | 8   |
|     | COMMON/TABLES/KDEPTH(25),KCLASS(25),PSIMAX(15),ALFMAX(15),         | QUAL | 9   |
| 1   | NN(25),MM(25),ANORM(15,51),QNORM(15,51),                           | QUAL | 10  |
| 2   | DNORM(15,51),AFAC(15),RFACT(15)                                    | QUAL | 11  |
|     | COMMON A(160,2,2),Q(160,2,2),CPOLL(160,2,2,3),QMAX(160),           | QUAL | 12  |
| 1   | CFULL(160),AFULL(160),DXDT(160),C1(160),SLOPE(160),                | QUAL | 13  |
| 2   | DIST(160),GEOM1(160),ROUGH(160),NOE(160),NUE(160,3),               | QUAL | 14  |
| 3   | INUE(160,3),NTYPE(160),JR(160),NKLASS,NE,NDT,EPSIL,                | QUAL | 15  |
| 4   | TIME,DT,M,KFULL,N,NOS,NPOLL,NPRINT,ITER,                           | QUAL | 16  |
| 5   | CDWF(160),IOLC(160),P1(160),RNOFF(160),QINFIL(160),                | QUAL | 17  |
| 6   | KCDWF(160,3),PLUTO(160,3),IR(160),P2(160),NIN(1000),               | QUAL | 18  |
| 7   | P5(160),P6(160),P7(160),SCF(160),BARREL(160),                      | QUAL | 19  |
| 8   | TITLE(40),NPE(20),NYN(20),NORDER(70),GEOM2(160),                   | QUAL | 20  |
| 9   | GECM3(160),P4(160),SCOUR(160),KSTORE(160)                          | QUAL | 21  |
|     | COMMON BCDIN( 2,150),SSIN( 2,150),BODOUT,SSOUT,COLIN(2,150),       | QUAL | 22X |
| *   | QINST,COUST,QINSTL( 2),COUSTL( 2),STURL( 2),QOUTO( 2),STURO( 2),   | QUAL | 23  |
| *   | NSTOR,KSTOR,IPRINT( 2),IPOL( 2),IFLOOD( 2),ICOST( 2),DEPMAX( 2),   | QUAL | 24  |
| *   | ATERM( 2,11),AO2DT2( 2,11),BDEPTH( 2,11),BSTOR( 2,11),COLOUT,      | QUAL | 25X |
| *   | DUMSTR(11),DUMDEP(11),                                             | QUAL | 26  |
| *   | KTSTEP,VOLIN( 2,150),VOLOUT( 2,150),STOR,CUMIN( 2),CUMOUT( 2),     | QUAL | 27  |
| *   | SBOD( 2),SSS( 2),SCCL( 2),                                         | QUAL | 28X |
| *   | ISTMOC( 2),ISTTYP( 2),ISTOUT( 2),                                  | QUAL | 29  |
| *   | QPUMP( 2),DSTART( 2),DSTOP( 2),                                    | QUAL | 30  |
| *   | DTGN( 2),STORMX( 2),DTPUMP( 2),DTMORE( 2),STORF( 2),APLAN( 2),     | QUAL | 31  |
| *   | CLAND( 2),CSTOR( 2),CPS( 2),CTOTAL( 2),CPCUYD( 2),CPACRE( 2),      | QUAL | 32  |
| *   | LP,JP,LPREV( 2),LABEL,DETENT(150),FRAC(150),OUT1(10,200)           | QUAL | 33X |
|     | EQUIVALENCE (PUMP(1),DIST(1)),(SURGE1(1),P1(1)),(SURGE2(1),P2(1))  | QUAL | 34  |
|     | EQUIVALENCE (QO1(1),QMAX(1)),(QO2(1),QFULL(1))                     | QUAL | 35  |
|     | EQUIVALENCE (QO(1),Q(1,2,2)),(QI(1),Q(1,1,2))                      | QUAL | 36  |
|     | EQUIVALENCE (WELL1(1),SLOPE(1)),(WELL2(1),ROUGH(1))                | QUAL | 37  |
|     | REAL KVAL                                                          | QUAL | 38  |
|     | P3=2./DT                                                           | QUAL | 39  |
|     | NT=NTYPE(M)                                                        | QUAL | 40  |
|     | GO TO(110,110,110,110,110,110,110,110,110,110,110,110,             | QUAL | 41  |
|     | 1 110,110,110,120,130,120,120,120,120,120,150), NT                 | QUAL | 42  |
| C   | PIPE***MIXING VOLUME FOUND BY AVERAGING UPSTREAM AND DOWNSTREAM    | QUAL | 43  |
| C   | FLOW AREAS AND BY MULTIPLYING BY LENGTH OF PIPE                    | QUAL | 44  |
| 110 | VOL1=DIST(M)*(A(M,1,1)+A(M,2,1))/2.0*BARREL(M)                     | QUAL | 45  |
|     | VOL2=DIST(M)*(A(M,1,2)+A(M,2,2))/2.0*BARREL(M)                     | QUAL | 46  |
|     | AREAF=(A(M,1,2)+A(M,2,2))/2.                                       | QUAL | 47  |
|     | RHYD=RADH(AREAF)                                                   | QUAL | 48  |
|     | GO TO 160                                                          | QUAL | 49  |
| C   | MANHOLE***MIXING VOLUME IS VOLUME CURRENTLY STORED AS SURCHARGING  | QUAL | 50  |
| 120 | VOL1=SURGE1(M)                                                     | QUAL | 51  |
|     | VOL2=SURGE2(M)                                                     | QUAL | 52  |
|     | SURGE1(M)=SURGE2(M)                                                | QUAL | 53  |
|     | GO TO 160                                                          | QUAL | 54  |
| C   | LIFT STATION***MIXING VOLUME IS VOLUME CURRENTLY IN WET WELL       | QUAL | 55  |
| 130 | VOL1=WELL1(M)                                                      | QUAL | 56  |
|     | VOL2=WELL2(M)                                                      | QUAL | 57  |
|     | WELL1(M)=WELL2(M)                                                  | QUAL | 58  |
|     | GO TO 160                                                          | QUAL | 59  |
| C   | ME FLOW CONTRL STRUCTURES***MIXING VOLUMES INITIALLY ZERO          | QUAL | 60  |

|     |                                                               |          |
|-----|---------------------------------------------------------------|----------|
| 150 | VOL1=0.0                                                      | QUAL 61  |
|     | VOL2=0.0                                                      | QUAL 62  |
| C   | QUALITY CONCENTRATIONS(LBS PER CU. FT.)***CPOLL(M,1,2)        | QUAL 63  |
| C   | POUNDS POLLUTANT IN SEWAGE                                    | QUAL 64  |
| 160 | DO 172 IP=1, APCLL                                            | QUAL 65  |
|     | TOTAL1=0.0                                                    | QUAL 66  |
|     | DO 170 JP=1,3                                                 | QUAL 67  |
|     | NNEED=INUE(M,JP)                                              | QUAL 68  |
|     | NTU=NTYPE(NNEED)                                              | QUAL 69  |
|     | IF(NTU.LE.17.OR.NTU.GE.23)GO TO 168                           | QUAL 70  |
|     | KK=GECF3(NNEED)                                               | QUAL 71  |
|     | QQ=QQ2(NNEED)                                                 | QUAL 72  |
|     | IF(NOE(X).EQ.KK)QQ=QO1(NNEED)                                 | QUAL 73  |
|     | GO TO 170                                                     | QUAL 74  |
| 168 | QQ=Q(NNEED,2,2)                                               | QUAL 75  |
| 170 | TOTAL1=TOTAL1+CPOLL(NNEED,2,2,IP)*QQ                          | QUAL 76  |
|     | NNEED=IR(M)                                                   | QUAL 77  |
|     | TOTAL3=PLUTO(NNEED,IP)                                        | QUAL 78  |
|     | GO TO(210,220,240,230),IP                                     | QUAL 79  |
| C   | PARAMETERS FOR B O O                                          | QUAL 80  |
| 210 | TOTAL2= CUMY2                                                 | QUAL 81  |
| C   | THE NUMBER $0.2314815E-5 = 0.2/(24.0*60.0*60.0)$              | QUAL 82  |
|     | D1 = $0.2314815E-5$                                           | QUAL 83  |
|     | D2=0.0                                                        | QUAL 84  |
|     | S=0.0                                                         | QUAL 85  |
|     | GO TO 250                                                     | QUAL 86  |
| C   | PARAMETERS FOR S S                                            | QUAL 87  |
| 220 | TOTAL2=CUMY3                                                  | QUAL 88  |
|     | D1=0.0                                                        | QUAL 89  |
|     | D2=0.0                                                        | QUAL 90  |
|     | S=0.0                                                         | QUAL 91  |
|     | IF(Q1(M).LE.0.0) GO TO 250                                    | QUAL 92  |
|     | IF(KLASS(NT).GT.2) GO TO 245                                  | QUAL 93  |
|     | SPG=2.7                                                       | QUAL 94  |
|     | KVAL=.056                                                     | QUAL 95  |
| C   | DETERMINE CRITICAL DIAMETER OF SOLIDS FOR MOTION.             | QUAL 96  |
|     | CRITD=RHYD*SLOPE(M)/(KVAL*(SPG-1.0))                          | QUAL 97  |
|     | CRITD=CRITD*304.8                                             | QUAL 98  |
|     | IF(CRITD.LE..59) PCT1=-1.2471*CRITD+1.0                       | QUAL 99  |
|     | IF(CRITD.GT..59.AND.CRITD.LE.2.0) PCT1=-.1501*CRITD+.3527     | QUAL 100 |
|     | IF(CRITD.GT.2.0.AND.CRITD.LE.10.0) PCT1=-.00656*CRITD+.0655   | QUAL 101 |
|     | IF(CRITD.GT.10.0) PCT1=0.0                                    | QUAL 102 |
|     | IF(CRITD.LE.0.0) PCT1=1.0                                     | QUAL 103 |
|     | PCT2=PCT1                                                     | QUAL 104 |
|     | SCOUR(M)=SCOUR(M)+PCT2*(TOTAL1+TOTAL2+TOTAL3)*DT              | QUAL 105 |
|     | TOTAL4=(1.-PCT2)*(TOTAL1+TOTAL2+TOTAL3)+(1.-PCT1)*SCOUR(M)/DT | QUAL 106 |
|     | CPOLL(M,1,2,2)=TOTAL4/Q1(M)                                   | QUAL 107 |
|     | SCOUR(M)=PCT1*SCOUR(M)                                        | QUAL 108 |
| C   | THE NUMBER $84.28 = 2.7*62.4/2.0$                             | QUAL 109 |
|     | ULIMIT=84.28*AFULL(M)*DIST(M)*BARREL(M)                       | QUAL 110 |
|     | IF(SCOUR(M).GT.ULIMIT) GO TO 300                              | QUAL 111 |
|     | GO TO 250                                                     | QUAL 112 |
| 300 | CPOLL(M,1,2,2)=((SCOUR(M)-ULIMIT)/Q1(M))/DT + CPOLL(M,1,2,2)  | QUAL 113 |
|     | SCOUR(M)=ULIMIT                                               | QUAL 114 |
|     | GO TO 250                                                     | QUAL 115 |
| C   | PARAMETERS FOR O O                                            | QUAL 116 |
| 230 | TOTAL2=DUMY4                                                  | QUAL 117 |
| C   | THE NUMBER $0.2314815E-5 = 0.2/(24.0*60.0*60.0)$              | QUAL 118 |
| C   | THE NUMBER $0.3472222E-5 = 0.3/(24.0*60.0*60.0)$              | QUAL 119 |
| C   | THE NUMBER $0.4356716E-3 = 7.0*8.34/134000.0$                 | QUAL 120 |

|     |                                                                 |         |
|-----|-----------------------------------------------------------------|---------|
|     | D1 = 0.2314815E-5                                               | QUAL121 |
|     | D2 = 0.3472222E-5                                               | QUAL122 |
|     | S = 0.4356716E-3                                                | QUAL123 |
|     | GO TO 250                                                       | QUAL124 |
| C   | PARAMETERS FOR NONSETTLABLE, CONSERVATIVE POLLUTANT, (COLIFORM) | QUAL125 |
| 240 | TOTAL2=CLMYS                                                    | QUAL126 |
|     | D1=0.0                                                          | QUAL127 |
|     | D2=0.0                                                          | QUAL128 |
|     | S=0.0                                                           | QUAL129 |
|     | GO TO 250                                                       | QUAL130 |
| 245 | IF(Q(M,1,2).LE.0.0001) GO TO 171                                | QUAL131 |
|     | CPOLL(M,1,2,IP)=(TOTAL1+TOTAL2+TOTAL3)/Q(M,1,2)                 | QUAL132 |
| 250 | CONTINUE                                                        | QUAL133 |
| C   | CONCENTRATION OF INFLCW TO ELEMENT                              | QUAL134 |
|     | IF(Q(M,1,2).LE.0.0001) GO TO 171                                | QUAL135 |
|     | IF(IP.NE.2) CPOLL(M,1,2,IP)=(TOTAL1+TOTAL2+TOTAL3)/Q(M,1,2)     | QUAL136 |
| C   | CONCENTRATION OF OUTFLOW FROM ELEMENT                           | QUAL137 |
| 171 | IF(Q(M,2,2).LE.0.0001) GO TO 172                                | QUAL138 |
|     | CPOLL(M,2,2,IP)=(CPOLL(M,2,1,IP)*(VOL1*(P3-D1-D2)-Q(M,2,1))+    | QUAL139 |
|     | 1CPOLL(M,1,1,IP)*Q(M,1,1)+CPOLL(M,1,2,IP)*Q(M,1,2)+D2*S*(VOL1   | QUAL140 |
|     | 2+VOL2))/(VOL2*(P3+D1+D2)+Q(M,2,2))                             | QUAL141 |
|     | IF(CPOLL(M,2,2,IP).LE.0.0) CPOLL(M,2,2,IP) =0.0                 | QUAL142 |
| 172 | CONTINUE                                                        | QUAL143 |
|     | RETURN                                                          | QUAL144 |
|     | END                                                             | QUAL145 |

|     |                                                                   |         |
|-----|-------------------------------------------------------------------|---------|
|     | FUNCTION RADH(AA)                                                 | RADH 1  |
| C   | FUNCTION TO COMPUTE THE HYDRAULIC RADIUS FOR A GIVEN FLOW AREA.   | RADH 2  |
|     | COMMON/TABLES/KDEPTH(25),KCLASS(25),PSIMAX(15),ALFMAX(15),        | RADH 3  |
| 1   | NN(25),MM(25),ANORM(15,51),QNORM(15,51),                          | RADH 4  |
| 2   | DNORM(15,51),AFAC(15),RFAC(15)                                    | RADH 5  |
|     | COMMON A(160,2,2),Q(160,2,2),CPOLL(160,2,2,3),QMAX(160),          | RADH 6  |
| 1   | QFULL(160),AFULL(160),DXDT(160),C1(160),SLOPE(160),               | RADH 7  |
| 2   | CIST(160),GEOM1(160),ROUGH(160),NOE(160),NUE(160,3),              | RADH 8  |
| 3   | INUE(160,3),NTYPE(160),JR(160),NKLASS,NE,NDT,EPSIL,               | RADH 9  |
| 4   | TIME,DT,M,KFULL,N,NUS,NPULL,NPRINT,ITER,                          | RADH 10 |
| 5   | QDWF(160),IOLE(160),P1(160),RNOFF(160),QINFIL(160),               | RADH 11 |
| 6   | KCWF(160,3),PLUTO(160,3),IR(160),P2(160),NIN(1000),               | RADH 12 |
| 7   | P5(160),P6(160),P7(160),SCF(160),BARKEL(160),                     | RADH 13 |
| 8   | TITLE(40),NPE(20),NYN(20),NORDER(70),GEOM2(160),                  | RADH 14 |
| 9   | GECM3(160),P4(160),SCCUR(160),KSTORE(160)                         | RADH 15 |
| C   | PARAMETERS NEEDED ARE:                                            | RADH 16 |
| C   |                                                                   | RADH 17 |
| C   | AA= AREA OF FLOW FOR WHICH RADH IS TO BE CALCULATED               | RADH 18 |
| C   |                                                                   | RADH 19 |
|     | REAL L1                                                           | RADH 20 |
|     | NT=NTYPE(M)                                                       | RADH 21 |
|     | AF=AFULL(M)                                                       | RADH 22 |
|     | IF(AA.LE.0.0) GO TO 330                                           | RADH 23 |
|     | GO TO (100,300,400,400,400,400,400,400,400,500,520,540,400,       | RADH 24 |
|     | 1 400,400,200,200,200,200,200,200,200,200,200,200), NT            | RADH 25 |
| C   | CIRCULAR PIPE                                                     | RADH 26 |
| 100 | DIAM=GECM1(M)                                                     | RADH 27 |
| 105 | ALPHA=AA/AF                                                       | RADH 28 |
|     | DALPHA=ANORM(1,2)-ANORM(1,1)                                      | RADH 29 |
|     | I=ALPHA/DALPHA +1.0                                               | RADH 30 |
|     | D1=DNORM(1,1)+(ALPHA-ANORM(1,1))/DALPHA*(DNORM(1,I+1)-DNORM(1,I)) | RADH 31 |
|     | D1=D1*DIAM                                                        | RADH 32 |
|     | RR=DIAM/2.0                                                       | RADH 33 |
|     | IF(D1-RR) 110,120,130                                             | RADH 34 |
| C   | HERE IF FLOW IS BELOW HALF WAY MARK                               | RADH 35 |
| 110 | D2=RR-D1                                                          | RADH 36 |
|     | L1= SQRT(RR**2-D2**2)                                             | RADH 37 |
|     | ARG=L1/D2                                                         | RADH 38 |
|     | THETA=2.0*ATAN(ARG)                                               | RADH 39 |
|     | S=RR*THETA                                                        | RADH 40 |
|     | RADH=AA/S                                                         | RADH 41 |
|     | RETURN                                                            | RADH 42 |
| C   | HERE IF FLOW IS AT HALF WAY MARK                                  | RADH 43 |
| 120 | RADH=AA/(3.14159*RR)                                              | RADH 44 |
|     | RETURN                                                            | RADH 45 |
| C   | HERE IF FLOW IS OVER HALF WAY MARK                                | RADH 46 |
| 130 | D2=D1-RR                                                          | RADH 47 |
|     | L1= SQRT(RR**2-D2**2)                                             | RADH 48 |
|     | THETA=2.0*ATAN2(L1,D2)                                            | RADH 49 |
|     | S=RR*THETA                                                        | RADH 50 |
|     | S= 2.0*3.14159*RR-S                                               | RADH 51 |
|     | RADH=AA/S                                                         | RADH 52 |
|     | RETURN                                                            | RADH 53 |
| C   | NO CALCULATIONS FOR NON CONDUITS                                  | RADH 54 |
| 200 | RADH=C.0                                                          | RADH 55 |
|     | RETURN                                                            | RADH 56 |
| C   | RECTANGULAR PIPE                                                  | RADH 57 |
| 300 | XL=GECM2(M)                                                       | RADH 58 |
|     | DIAM=GECM1(M)                                                     | RADH 59 |
|     | RADH= AA/(XL+2.0*(AA/AF)*DIAM)                                    | RADH 60 |

|                                                        |         |
|--------------------------------------------------------|---------|
| RETURN                                                 | RADH 61 |
| C HERE IF NO FLOW IN PIPE                              | RADH 62 |
| 330 RADH=0.0                                           | RADH 63 |
| RETURN                                                 | RADH 64 |
| C ASSUME EQUIVALENT CIRCULAR PIPE FOR UDD SHAPES       | RADH 65 |
| 400 DIAM=SQRT(4.0*AF/3.1417)                           | RADH 66 |
| GO TO 105                                              | RADH 67 |
| C HYDRAULIC RADIUS FOR MODIFIED BASKET HANDLE CONDUIT. | RADH 68 |
| 500 DD = (GECM1(M)+GECM2(M)/2.0)*DEPTH(AA/AFULL(M))    | RADH 69 |
| IF (DD.GT.GECM1(M)) GO TO 505                          | RADH 70 |
| RADH = AA/(GECM2(M)+2.0*DD)                            | RADH 71 |
| RETURN                                                 | RADH 72 |
| 505 CATHY = DD-GECM1(M)                                | RADH 73 |
| THETA = ARSIN(2.0*CATHY/GEOM2(M))                      | RADH 74 |
| PER = 2.0*GECM1(M)+GECM2(M)*(1.0+THETA)                | RADH 75 |
| RADH = AA/PER                                          | RADH 76 |
| RETURN                                                 | RADH 77 |
| C HYDRAULIC RADIUS FOR RECTANGULAR, TRIANGULAR BOTTOM. | RADH 78 |
| 520 DD = GECM1(M)*DEPTH(AA/AFULL(M))                   | RADH 79 |
| IF (DD.GT.GEOM3(M)) GO TO 525                          | RADH 80 |
| PER = CC/P5(M)                                         | RADH 81 |
| RADH = AA/PER                                          | RADH 82 |
| RETURN                                                 | RADH 83 |
| 525 CATHY = DD-GECM3(M)                                | RADH 84 |
| PER = 2.0*CATHY+GEOM3(M)/P5(M)                         | RADH 85 |
| RADH = AA/PER                                          | RADH 86 |
| RETURN                                                 | RADH 87 |
| C HYDRAULIC RADIUS FOR RECTANGULAR, ROUND BOTTOM.      | RADH 88 |
| 540 IF (AA.GT.P6(M)) GO TO 545                         | RADH 89 |
| DIAM = 2.0*GEOM3(M)                                    | RADH 90 |
| AF = 3.141597*GECM3(M)*GECM3(M)                        | RADH 91 |
| GO TO 105                                              | RADH 92 |
| 545 CATHY = AA-P6(M)                                   | RADH 93 |
| PER = 2.0*CATHY/GEOM2(M)+GEOM3(M)*P5(M)                | RADH 94 |
| RADH = AA/PER                                          | RADH 95 |
| RETURN                                                 | RADH 96 |
| END                                                    | RADH 97 |

```

SUBROUTINE RCUTE(NITER)                                ROUT 1
ROUTING SUBPROGRAM FOR FLOW THROUGH SEWER ELEMENTS.    ROUT 2
IN PROGRAM, ALPHA REPRESENTS NORMALIZED AREA, A/AFULL. ROUT 3
IN PROGRAM, PSI AND PS REPRESENT NORMALIZED FLOW, Q/QFULL. ROUT 4
DIMENSION QI(160),QU(160),SURGE1(160),SURGE2(160),WELL1(160), ROUT 5
1 WELL2(160),PUMPI(160),G(400),QO1(160),QO2(160)      ROUT 6
COMMON/TABLES/KDEPTH(25),KLASS(25),PSIMAX(15),ALFMAX(15), ROUT 7
1 NN(25),MM(25),ANORM(15,51),QNORM(15,51),            ROUT 8
2 DNORM(15,51),AFAC(15),RFACT(15)                    ROUT 9
COMMON/NAMES/ NAME(4,25),GNO,YES,BLANK                ROUT 10
COMMON A(160,2,2),Q(160,2,2),CPOLL(160,2,2,3),QMAX(160), ROUT 11
1 QFULL(160),AFULL(160),DXDT(160),C1(160),SLOPE(160),  ROUT 12
2 DIST(160),GEOM1(160),ROUGH(160),NOE(160),NUE(160,3), ROUT 13
3 INUE(160,3),NTYPE(160),JR(160),NKLASS,NE,NOT,EPSIL,  ROUT 14
4 TIME,DT,M,KFULL,N,NJS,NPULL,NPRINT,ITER,           ROUT 15
5 GCWF(160),IOLD(160),P1(160),RNOFF(160),QINFIL(160),  ROUT 16
6 hCWF(160,3),PLUTO(160,3),IR(160),P2(160),NIN(1000),  ROUT 17
7 P5(160),P6(160),P7(160),SCF(160),BARREL(160),       ROUT 18
8 TITLE(40),NPE(20),NYN(20),NORDER(70),GEOM2(160),    ROUT 19
9 GEOM3(160),P4(160),SCGUR(160),KSTORE(160)           ROUT 20
COMMON BCLIN( 2,150),SSIN( 2,150),BDDOUT,SSOUT,COLIN(2,150), ROUT 21X
* QINST,COLST,QINSTL( 2),QOUSTL( 2),STORL( 2),QOUTO( 2),STORO( 2),ROUT 22
* NSTOR,KSTOR,IPRINT( 2),IPOL( 2),IFLOOD( 2),ICOST( 2),DEPMAX( 2),ROUT 23
* ATERM( 2,11),A02DT2( 2,11),BDEPTH( 2,11),BSTOR( 2,11),COLOUT,  ROUT 24X
* DUMSTR(11),DUMDEP(11),                                ROUT 25
* KSTEP,VCLIN( 2,150),VOLOUT( 2,150),STOR,CUMIN( 2),CUMOUT( 2),  ROUT 26
* SBOD( 2),SSS( 2),SCCL( 2),                            ROUT 27X
* ISTMOD( 2),ISTTYP( 2),ISTOUT( 2),                      ROUT 28
* QPUMP( 2),DSTART( 2),DSTOP( 2),                        ROUT 29
* DTGN( 2),STORFX( 2),DTPUMP( 2),DTMORE( 2),STORF( 2),APLAN( 2),  ROUT 30
* CLAND( 2),CSTOR( 2),CPS( 2),CTOTAL( 2),CPCUYD( 2),CPACRE( 2),  ROUT 31
* LP,JP,LPREV( 2),LABEL,DETENT(150),FRAC(150),OUT1(10,200)      ROUT 32X
EQUIVALENCE (QO(1),Q(1,2,2)),(QI(1),Q(1,1,2))        ROUT 33
EQUIVALENCE (PUMP(1),DIST(1)),(SURGE1(1),P1(1)),(SURGE2(1),P2(1)) ROUT 34
EQUIVALENCE (CQ1(1),QMAX(1)),(QO2(1),QFULL(1))        ROUT 35
EQUIVALENCE (WELL1(1),SLOPE(1)),(WELL2(1),ROUGH(1))    ROUT 36
DATA WT/C.55/,WD/O.55/                                ROUT 37
C USE GENERALIZED WEIGHTS OF TIME AND SPACE DERIVATIVES OF  ROUT 38
C CONTINUITY EQUATION.                                  ROUT 39
C WT=WEIGHT ON TIME DERIVATIVE.                         ROUT 40
C WD=WEIGHT ON SPATIAL DERIVATIVE.                      ROUT 41
C THESE CONSTANTS APPEAR IN DEFINITIONS OF C1(M) AND C2.  ROUT 42
C NOTE, WT=WD=0.5 CORRESPOND TO ORIGINAL VERSION OF TRANSPORT MODEL. ROUT 43
C WT=WD=0.55 APPEAR TO GIVE BEST ATTENUATION OF HYDROGRAPHS.  ROUT 44
C NT = NTYPE(M)                                          ROUT 45
C M = ELEMENT NUMBER                                    ROUT 46
C KLASS(NT)=1 FOR CONDUIT WITH FUNCTIONAL Q-A RELATIONSHIP.  ROUT 47
C KLASS(NT)=2 FOR CONDUIT WITH TABULAR Q-A RELATIONSHIP.    ROUT 48
C KLASS(NT)=3 FOR ELEMENT OTHER THAN CONDUIT.             ROUT 49
C K = KLASS(NT)                                          ROUT 50
GO TO (5,5,200), K                                     ROUT 51
C COMPUTE TOTAL HEAD SLOPE AND ASSOCIATED PARAMETERS.      ROUT 52
C KFULL=1 IF CONDUIT IS FULL AT UPSTREAM END.             ROUT 53
C KFULL=2 IF CONDUIT IS NOT FULL AT UPSTREAM END.         ROUT 54
5 IF (KFULL.EQ.1) GO TO 10                              ROUT 55
IF (SCF(M).EQ.GNO) GO TO 6                              ROUT 56
C NO ITERATIONS REQUIRED FOR CONDUITS WITH SUPER-CRITICAL FLOW.  ROUT 57
ITER = 0                                                  ROUT 58
GO TO (20,100), K                                       ROUT 59
6 IF(ITER.GT.NITER) GO TO 7                              ROUT 60

```

|                                                                   |          |
|-------------------------------------------------------------------|----------|
| QOLD=QFULL(M)                                                     | ROUT 61  |
| A1 = A(M,1,1)/AFULL(M)                                            | ROUT 62  |
| A2 = A(M,2,1)/AFULL(M)                                            | ROUT 63  |
| DV = (VEL(Q(M,1,1),A(M,1,1))**2- VEL(Q(M,2,1),A(M,2,1))**2)/      | ROUT 64  |
| 1DIST(M)/64.4                                                     | ROUT 65  |
| WSLOPE = SLOPE(M)+P2(M)*(DEPTH(A1)-DEPTH(A2))+DV                  | ROUT 66  |
| GO TO 9                                                           | ROUT 67  |
| 7 A1 = A(M,1,2)/AFULL(M)                                          | ROUT 68  |
| A2 = A(M,2,2)/AFULL(M)                                            | ROUT 69  |
| DV = (VEL(Q(M,1,2),A(M,1,2))**2- VEL(Q(M,2,2),A(M,2,2))**2)/      | ROUT 70  |
| 1DIST(M)/64.4                                                     | ROUT 71  |
| WSLOPE = SLOPE(M)+P2(M)*(DEPTH(A1)-DEPTH(A2))+DV                  | ROUT 72  |
| 9 IF (WSLOPE.LT.SLOPE(M)) WSLOPE = SLOPE(M)                       | ROUT 73  |
| 8 QFULL(M) = P1(M)*SQRT(WSLOPE)                                   | ROUT 74  |
| DELQ=ABS(QFULL(M)-QOLD)                                           | ROUT 75  |
| IF(DELQ.LT..00002*QFULL(M)) ITER=0                                | ROUT 76  |
| QFULL(M)=QOLD+(CFULL(M)-QOLD)/2.                                  | ROUT 77  |
| QOLD=QFULL(M)                                                     | ROUT 78  |
| IF (ITER-NITER.GE.NITER-1) ITER=0                                 | ROUT 79  |
| 10 C1(M) = AFULL(M)/QFULL(M)*DXDT(M)*WT/WD                        | ROUT 80  |
| QMAX(M) = P4(M)*QFULL(M)                                          | ROUT 81  |
| GO TO (20,100), K                                                 | ROUT 82  |
| C ROUTINE FOR CONDUIT WITH FUNCTIONAL Q-A RELATIONSHIP.           | ROUT 83  |
| C FIND NEW UPSTREAM AREA.                                         | ROUT 84  |
| 20 GO TO (25,30), KFULL                                           | ROUT 85  |
| 25 A(M,1,2)=AFULL(M)                                              | ROUT 86  |
| GO TO 40                                                          | ROUT 87  |
| 30 C2 = -Q(M,1,2)/QFULL(M)                                        | ROUT 88  |
| ALPHA = A(M,1,1)/AFULL(M)                                         | ROUT 89  |
| CALL NEWTON(ALPHA,PS,0.0,C2,KFLAG)                                | ROUT 90  |
| IF (KFLAG.NE.2) GO TO 35                                          | ROUT 91  |
| WRITE (6,910) TIME,N,NOE(M),A(M,1,1)                              | ROUT 92  |
| ALPHA = A(M,1,1)/AFULL(M)                                         | ROUT 93  |
| 35 A(M,1,2) = ALPHA*AFULL(M)                                      | ROUT 94  |
| C ASSIGN VALUES TO CONSTANTS AND SOLVE FOR DOWNSTREAM Q AND A.    | ROUT 95  |
| 40 C2 = (1.0-WC)*Q(M,2,1)-(1.0-WD)*Q(M,1,1)-WD*Q(M,1,2)           | ROUT 96  |
| C2 = C2+CXDT(M)*((1.0-WT)*A(M,1,2)-(1.0-WT)*A(M,1,1)-WT*A(M,2,1)) | ROUT 97  |
| C2 = C2/QFULL(M)/WD                                               | ROUT 98  |
| ALPHA = A(M,2,1)/AFULL(M)                                         | ROUT 99  |
| CALL NEWTON(ALPHA,PS,C1(M),C2,KFLAG)                              | ROUT 100 |
| GO TO (45,50),KFLAG                                               | ROUT 101 |
| C DOWNSTREAM Q AND A MAY NOW BE CALCULATED.                       | ROUT 102 |
| 45 A(M,2,2) = ALPHA*AFULL(M)                                      | ROUT 103 |
| Q(M,2,2) = PS*QFULL(M)                                            | ROUT 104 |
| RETURN                                                            | ROUT 105 |
| C DETERMINE REASON FOR NON-CONVERGENCE.                           | ROUT 106 |
| C THEN USE DEFAULT OPTIONS TO DETERMINE Q AND A.                  | ROUT 107 |
| 50 IF(NPRINT.LT.1) GO TO 52                                       | ROUT 108 |
| WRITE (6,900)                                                     | ROUT 109 |
| WRITE(6,901) TIME,M,QFULL(M),AFULL(M),DXDT(M),Q(M,1,1),Q(M,1,2),  | ROUT 110 |
| 1Q(M,2,1),A(M,1,1),A(M,1,2),A(M,2,1),ALPHA                        | ROUT 111 |
| C SEE IF LINE-C1*ALPHA-C2 INTERSECTS Q-A CURVE.                   | ROUT 112 |
| 52 IF ((-C1(M)*ALFMAX(NT)-C2).GT.P4(M)) GO TO 60                  | ROUT 113 |
| IF(C2.GT.0) GO TO 65                                              | ROUT 114 |
| C REASON FOR NON-CONVERGENCE UNDETERMINED.                        | ROUT 115 |
| C LET NEW C AND A EQUAL VALUES AT PREVIOUS TIME STEP.             | ROUT 116 |
| IF (NPRINT.GE.1) WRITE (6,902)                                    | ROUT 117 |
| A(M,2,2) = A(M,2,1)                                               | ROUT 118 |
| Q(M,2,2) = Q(M,2,1)                                               | ROUT 119 |
| RETURN                                                            | ROUT 120 |

|     |                                                                   |         |
|-----|-------------------------------------------------------------------|---------|
| C   | CONTINUITY EQUATION TRIES TO FORCE Q.GT.QMAX.                     | ROUT121 |
| C   | LET DOWNSTREAM FLOW BE QFULL UNLESS UPSTREAM Q IS GT QFULL.       | ROUT122 |
| 60  | IF (NPRINT.GE.1) WRITE (6,903)                                    | ROUT123 |
|     | IF(Q(M,1,2).GT.QFULL(M)) GO TO 75                                 | ROUT124 |
|     | Q(M,2,2) = QFULL(M)                                               | ROUT125 |
|     | A(M,2,2) = AFULL(M)                                               | ROUT126 |
|     | RETURN                                                            | ROUT127 |
| C   | CONTINUITY EQUATION TRIES TO FORCE Q. LT. ZERO.                   | ROUT128 |
| C   | LET DOWNSTREAM FLOW BE ZERO.                                      | ROUT129 |
| 65  | IF (NPRINT.GE.1) WRITE (6,904)                                    | ROUT130 |
|     | Q(M,2,2) = 0.0                                                    | ROUT131 |
|     | A(M,2,2)=0.0                                                      | ROUT132 |
|     | RETURN                                                            | ROUT133 |
| 75  | Q(M,2,2) = Q(M,1,2)                                               | ROUT134 |
|     | A(M,2,2) = A(M,1,2)                                               | ROUT135 |
|     | RETURN                                                            | ROUT136 |
| C   | ROUTINE FOR CONDUIT WITH TABULAR Q-A RELATIONSHIP.                | ROUT137 |
| 100 | DALPHA = ANORM(NT,2)-ANORM(NT,1)                                  | ROUT138 |
|     | MMM = MM(NT)                                                      | ROUT139 |
| C   | CALCULATE UPSTREAM AREA.                                          | ROUT140 |
|     | GO TO (102,105),KFULL                                             | ROUT141 |
| 102 | A(M,1,2) = AFULL(M)                                               | ROUT142 |
|     | GO TO 115                                                         | ROUT143 |
| 105 | PS = Q(M,1,2)/QFULL(M)                                            | ROUT144 |
|     | CALL FINCA(PS,A(M,1,2))                                           | ROUT145 |
| C   | CALCULATE DOWNSTREAM Q AND A.                                     | ROUT146 |
| 115 | ISIGN = 1                                                         | ROUT147 |
|     | IF (A(M,1,2).LT.A(M,1,1)) ISIGN = -1                              | ROUT148 |
|     | ICLK = 1                                                          | ROUT149 |
|     | I = IOLD(M)                                                       | ROUT150 |
|     | C2 = (1.0-WD)*Q(M,2,1)-(1.0-WC)*Q(M,1,1)-WD*Q(M,1,2)              | ROUT151 |
|     | C2 = C2+DXDT(M)*((1.0-WT)*A(M,1,2)-(1.0-WT)*A(M,1,1)-WT*A(M,2,1)) | ROUT152 |
|     | C2 = C2/QFULL(M)/WD                                               | ROUT153 |
| C   | CALCULATE SLOPE OF LINE SEGMENT I OF Q-A CURVE.                   | ROUT154 |
| 120 | SLUPE=(QNORM(NT,I+1)-QNORM(NT,I))/DALPHA                          | ROUT155 |
|     | IF(SLUPE+C1(M).EQ.0.0) GO TO 130                                  | ROUT156 |
| C   | COMPUTE ALPHA CORRESPONDING TO INTERSECTION OF LINE SEGMENT I OF  | ROUT157 |
| C   | Q-A CURVE WITH LINE -C1*ALPHA-C2.                                 | ROUT158 |
|     | ALPHA=(-QNORM(NT,I)-C2+SLUPE*ANORM(NT,I))/(SLUPE+C1(M))           | ROUT159 |
| C   | CHECK TO SEE IF ALPHA IS IN PROPER RANGE.                         | ROUT160 |
|     | IF (ALPHA.GT.1.0.OR.ALPHA.LT.0) GO TO 125                         | ROUT161 |
|     | L= (ALPHA/DALPHA) + 1.0                                           | ROUT162 |
|     | PS = QNORM(NT,L)+(ALPHA-ANORM(NT,L))/DALPHA *(QNORM(NT,L+1)       | ROUT163 |
|     | 1-QNORM(NT,L))                                                    | ROUT164 |
|     | IF(ABS(PS+C1(M)*ALPHA+C2).LE.EPSIL) GO TO 150                     | ROUT165 |
| C   | TRY NEXT LINE SEGMENT.                                            | ROUT166 |
| 125 | I = I+ISIGN                                                       | ROUT167 |
| 126 | IF (I.GT.0.AND.I.LT.MMM) GO TO 120                                | ROUT168 |
|     | GO TO (127,50),ICLK                                               | ROUT169 |
| C   | IF I HAS REACHED ZERO OR MMM START AT IOLD AND GO OTHER WAY       | ROUT170 |
| 127 | ISIGN = -ISIGN                                                    | ROUT171 |
|     | I = IOLD(M)+ISIGN                                                 | ROUT172 |
|     | ICLK = 2                                                          | ROUT173 |
|     | GO TO 126                                                         | ROUT174 |
| C   | LINE-C1*ALPHA-C2 AND LINE SEGMENT ARE PARALLEL.                   | ROUT175 |
| C   | CHECK TO SEE IF THEY ARE CO-LINEAR.                               | ROUT176 |
| 130 | IF(ABS(C2+QNORM(NT,I)-SLUPE*ANORM(NT,I)).GT.EPSIL) GO TO 125      | ROUT177 |
|     | ALPHA = ANORM(NT,I)+DALPHA/2.0                                    | ROUT178 |
|     | L= (ALPHA/DALPHA) + 1.0                                           | ROUT179 |
|     | PS = QNORM(NT,L)+(ALPHA-ANORM(NT,L))/DALPHA *(QNORM(NT,L+1)       | ROUT180 |



|                                                                  |         |
|------------------------------------------------------------------|---------|
| I=CNORM(AT,L1)                                                   | ROUT181 |
| 150 A(M,2,2) = ALPHA*AFULL(M)                                    | ROUT182 |
| Q(M,2,2) = PS*QFULL(M)                                           | ROUT183 |
| RETURN                                                           | ROUT184 |
| C ELEMENT IS NOT A CONDUIT.                                      | ROUT185 |
| 200 NGOTO = AT-15                                                | ROUT186 |
| IF (NGOTO.LT.1) GO TO 210                                        | ROUT187 |
| GO TO (220,230,250,400,500,250,520,220,220,220), NGOTO           | ROUT188 |
| C ERROR***CONDUITS SHOULD BE CLASS 1 OR 2                        | ROUT189 |
| 210 QO(M)=QI(M)                                                  | ROUT190 |
| WRITE (6,905) M                                                  | ROUT191 |
| RETURN                                                           | ROUT192 |
| C HANHOLE *** SIMPLY TRANSLATE FLOW WITH NO TIME DELAY           | ROUT193 |
| 220 QO(M)=QI(M)                                                  | ROUT194 |
| RETURN                                                           | ROUT195 |
| C LIFT STATICA***PUMPS ASSUMED TO BE PUMPING AT CONSTANT RATE    | ROUT196 |
| C (=PUMP(M)).                                                    | ROUT197 |
| C VOLUME IN WET WELL INITIALLY IS HALF THE CAPACITY(GEOM1).      | ROUT198 |
| C FORCE MAIN ASSUMED TO REMAIN FULL AT ALL TIMES RESULTING IN NO | ROUT199 |
| C TIME DELAY.                                                    | ROUT200 |
| 230 WELL2(M) = WELL2(M)+QI(M)*DT                                 | ROUT201 |
| IF (WELL2(M)-GECM1(M))233,234,234                                | ROUT202 |
| 233 QO(M) = 0.0                                                  | ROUT203 |
| RETURN                                                           | ROUT204 |
| 234 IF (WELL2(M).GE.PUMP(M)*DT) GO TO 236                        | ROUT205 |
| QO(M) = WELL2(M)/DT                                              | ROUT206 |
| WELL2(M) = 0.0                                                   | ROUT207 |
| RETURN                                                           | ROUT208 |
| 236 QO(M) = PUMP(M)                                              | ROUT209 |
| WELL2(M) = WELL2(M)-PUMP(M)*DT                                   | ROUT210 |
| RETURN                                                           | ROUT211 |
| C ROUTINE FOR TYPE 18 AND TYPE 21 FLOW DIVIDERS.                 | ROUT212 |
| C TYPE 18 MAY BE USED FOR A SIMPLE OVERFLOW STRUCTURE.           | ROUT213 |
| C TYPE 21 MAY BE USED WITH A CUNNETTE SECTION DOWNSTREAM.        | ROUT214 |
| 250 QO(M) = QI(M)                                                | ROUT215 |
| IF (QI(M).GT.GECM1(M)) GO TO 255                                 | ROUT216 |
| QO1(M) = QI(M)                                                   | ROUT217 |
| QO2(M) = 0.0                                                     | ROUT218 |
| RETURN                                                           | ROUT219 |
| 255 QO1(M) = GECM1(M)                                            | ROUT220 |
| QO2(M) = QI(M)-GECM1(M)                                          | ROUT221 |
| RETURN                                                           | ROUT222 |
| C STORAGE UNIT. ADAPTED FROM M&E ROUTINE.                        | ROUT223 |
| 400 KSTOR = KSTORE(M)                                            | ROUT224 |
| IF (IFLUCD(KSTOR) .EQ. 1) GO TO 450                              | ROUT225 |
| KISTEP= N                                                        | ROUT226 |
| QINST = QI(M)                                                    | ROUT227 |
| DO 420 IP=1,NPOLL                                                | ROUT228 |
| CPOLL(M,1,2,IP)=0.0                                              | ROUT229 |
| IF(QI(M).LE.0.0) GO TO 420                                       | ROUT230 |
| DO 415 J=1,3                                                     | ROUT231 |
| NNEED = INUE(M,J)                                                | ROUT232 |
| NTU = NTYPE(NNEED)                                               | ROUT233 |
| IF (NTU.LE.17.OR.NTU.GE.23) GO TO 413                            | ROUT234 |
| L = GECM3(NNEED)                                                 | ROUT235 |
| QC = QO2(NNEED)                                                  | ROUT236 |
| IF (NOE(M).EQ.L) QC=QO1(NNEED)                                   | ROUT237 |
| GO TO 415                                                        | ROUT238 |
| 413 QC = QC(NNEED)                                               | ROUT239 |
| 415 CPOLL(M,1,2,IP) = CPOLL(M,1,2,IP)+CPOLL(NNEED,2,2,IP)*QC     | ROUT240 |

|                                                                               |          |
|-------------------------------------------------------------------------------|----------|
| CPOLL(M,1,2,IP) = CPOLL(M,1,2,IP)/QI(M)                                       | ROUT241  |
| 420 CONTINUE                                                                  | ROUT242  |
| BUDIN(KSTOR,KTSTEP) = CPOLL(M,1,2,1)*QI(M)*DT                                 | ROUT243  |
| SSIN(KSTOR,KTSTEP) = CPOLL(M,1,2,2)*QI(M)*DT                                  | ROUT244  |
| CULIN(KSTOR,KTSTEP) = CPOLL(M,1,2,3)*QI(M)*DT                                 | ROUT244A |
| CALL TSTORG                                                                   | ROUT245  |
| IF (ISTOUT(KSTOR).NE.9) CO2(M)=QOUST                                          | ROUT246  |
| QO(M) = QOUST                                                                 | ROUT247  |
| IF(QC(M).GT.0.0) GO TO 440                                                    | ROUT248  |
| CPOLL(M,2,2,1)=C.0                                                            | ROUT249  |
| CPOLL(M,2,2,2)=C.0                                                            | ROUT250  |
| CPOLL(M,2,2,3)=0.0                                                            | ROUT250A |
| GO TO 450                                                                     | ROUT251  |
| 440 CPOLL(M,2,2,1) = BDDCUT/(QO(M)*DT)                                        | ROUT252  |
| CPOLL(M,2,2,2) = SSCUT/(QO(M)*DT)                                             | ROUT253  |
| CPOLL(M,2,2,3) = COLCUT/(QO(M)*DT)                                            | ROUT253A |
| 450 RETURN                                                                    | ROUT254  |
| C ROUTINE FOR TYPE 20 FLOW DIVIDER.                                           | ROUT255  |
| C USE WITH SIDE WEIR OR OTHER WEIR TYPE DIVERSION.                            | ROUT256  |
| C DIST=MAX FLOW WITHOUT FLOW OVER WEIR.                                       | ROUT257  |
| C SLOPE=MAXIMUM FLOW THROUGH WHOLE STRUCTURE.                                 | ROUT258  |
| C ROUGH=WEIR CONSTANT TIMES WEIR LENGTH.                                      | ROUT259  |
| C GEOM1 = WEIR HEIGHT.                                                        | ROUT260  |
| C GEOM2 = DEPTH IN STRUCTURE AT TIME OF MAXIMUM FLOW.                         | ROUT261  |
| C GEOM3 = DOWNSTREAM EXT. ELE. NUM. INTO WHICH GOES UNDIVERTED FLOW.          | ROUT262  |
| C FLOW OVER WEIR IS THE DIVERTED FLOW.                                        | ROUT263  |
| 500 QO(M) = QI(M)                                                             | ROUT264  |
| IF (QI(M).GT.DIST(M)) GO TO 505                                               | ROUT265  |
| CO2(M) = 0.0                                                                  | ROUT266  |
| QO1(M) = QI(M)                                                                | ROUT267  |
| RETURN                                                                        | ROUT268  |
| 505 DH = (QI(M)-DIST(M))/(SLOPE(M)-DIST(M))*(GEOM2(M)-GEOM1(M))               | ROUT269  |
| QO2(M) = ROUGH(M)*DH*SQRT(DH)                                                 | ROUT270  |
| QO1(M) = QI(M)-QO2(M)                                                         | ROUT271  |
| RETURN                                                                        | ROUT272  |
| C ROUTINE FOR BACKWATER ELEMENT.                                              | ROUT273  |
| C ELEMENT ACTS AS FLOW DIVIDER.                                               | ROUT274  |
| C GEOM3=ELEMENT NUMBER CF DOWNSTREAM STORAGE UNIT.                            | ROUT275  |
| 520 QO(M) = QI(M)                                                             | ROUT276  |
| L = GEOM3(M)                                                                  | ROUT277  |
| L = MIN(L)                                                                    | ROUT273  |
| KSTOR = KSTORE(L)                                                             | ROUT279  |
| C IF BACKWATER EXTENDS ALL THE WAY UP TO ELEMENT,                             | ROUT280  |
| C THEN WHOLE FLOW IS DIVERTED TO STORAGE ELEMENT.                             | ROUT281  |
| C ASSUME LENGTH OF BACKWATER IS PROPORTIONAL TO SQRT OF STORAGE               | ROUT282  |
| C VOLUME.                                                                     | ROUT283  |
| QO1(M) = QI(M)*SQRT(ISTORL(KSTOR)/BSTOR(KSTOR,11))                            | ROUT284  |
| QO2(M) = QI(M)-QO1(M)                                                         | ROUT285  |
| RETURN                                                                        | ROUT286  |
| 900 FORMAT (' ITERATION FOR DOWNSTREAM FLOW HAS NOT CONVERGED.')              | ROUT287  |
| 13X'TIME ELEMENT QFULL AFULL DXDT Q(M,1,1) Q(M,1,2) Q(M,2,1) AROUT288         |          |
| 2(M,1,1) A(M,1,2) A(M,2,1) ALPHA')                                            | ROUT289  |
| 901 FORMAT (F8.1,16,F9.1,F7.2,F7.1,3F9.2,3F9.3,F7.3)                          | ROUT290  |
| 902 FORMAT (' REASON FOR NON-CONVERGENCE UNDETERMINED. USE Q,A VALUES         | ROUT291  |
| 1AT PREVIOUS TIME STEP.'//)                                                   | ROUT292  |
| 903 FORMAT (' CONTINUITY EQN TRIES TO FORCE Q.GT.QMAX. USE QFULL UNLESROUT293 |          |
| IS UPSTREAM Q.GT.QFULL - THEN USE UPSTREAM VALUE.')                           | ROUT294  |
| 904 FORMAT (' CONTINUITY EQN TRIES TO FORCE Q.LT.ZERO. USE ZERO.')            | ROUT295  |
| 905 FORMAT (' ERROR: CONDUITS SHOULD BE CLASS 1 OR 2. M='15)                  | ROUT296  |
| 910 FORMAT (' **WARNING: NEWTON UNABLE TO FIND AREA GIVEN FLOW. TIME=ROUT297  |          |

1'F7.1,', TIME STEP='I3,', EXT. ELE. NUM.='I4,', USE OLD UPSTREAM AROUT298  
2REA='F6.2) ROUT299  
END ROUT300

```

SUBROUTINE SLOP                                SLOP 2
C ROUTINE TO SEQUENCE ELEMENTS FOR COMPUTATION. SLOP 3
C THIS VERSION OF SLOP USES VECTOR 'NIN' TO POINT TO INTERNAL NUMBERS SLOP 4
C EXTERNAL ELEMENT NUMBERS MUST BE NUMERIC AND LE 1000. SLOP 5
COMMON/NAMES/ NAME(4,25),GNO,YES,BLANK SLOP 6
COMMON/TABLES/KDEPTH(25),KLASS(25),PSIMAX(15),ALFMAX(15), SLOP 7
1 NN(25),MM(25),ANORM(15,51),QNORM(15,51), SLOP 8
2 DNORM(15,51),AFACT(15),RFAC(15) SLOP 9
COMMON A(160,2,2),Q(160,2,2),CPOLL(160,2,2,3),QMAX(160), SLOP 10
1 QFULL(160),AFULL(160),DXDT(160),C1(160),SLOPE(160), SLOP 11
2 DIST(160),GEOM1(160),ROUGH(160),NOE(160),NUE(160,3), SLOP 12
3 INUE(160,3),NTYPE(160),JR(160),NKLASS,NE,NDT,EPSIL, SLOP 13
4 TIME,D1,M,KFULL,N,NUS,NPULL,NPRINT,ITER, SLOP 14
5 QDWF(160),IOLD(160),PI(160),RNOFF(160),QINFIL(160), SLOP 15
6 WCHF(160,3),PLUTO(160,3),IR(160),P2(160),NIN(1000), SLOP 16
7 P5(160),F6(160),P7(160),SCF(160),BARREL(160), SLOP 17
8 TITLE(40),NPE(20),NYN(20),NURDER(70),GEOM2(160), SLOP 18
9 GECM3(160),P4(160),SCOUR(160),KSTORE(160) SLOP 19
COMMON BODIN( 2,150),SSIN( 2,150),BODOUT,SSOUT,CULIN(2,150), SLOP 20X
* QINST,COUST,QINSTL( 2),QOUSTL( 2),STORL( 2),QOUTO( 2),STORO( 2), SLOP 21
* NSTOR,KSTOR,IFRINT( 2),IPOL( 2),IFLOOD( 2),ICOST( 2),DEPMAX( 2), SLOP 22
* ATERM( 2,11),A02DT2( 2,11),BDEPTH( 2,11),BSTOR( 2,11),COLOJT, SLOP 23X
* DJKSTR(11),CUMDEP(11), SLOP 24
* KSTEP,VOLIN( 2,150),VOLOUT( 2,150),STOR,CUMIN( 2),CJMOUT( 2), SLOP 25
* SEOB( 2),SSS( 2),SCCL( 2), SLOP 26X
* ISTMG( 2),ISTTYP( 2),ISTOUT( 2), SLOP 27
* QPUMP( 2),DSTART( 2),DSTOP( 2), SLOP 28
* DTCK( 2),STORMX( 2),DTPUMP( 2),DTMORE( 2),STORF( 2),APLAN( 2), SLOP 29
* CLAND( 2),CSTOR( 2),CPS( 2),CTOTAL( 2),CPCUYD( 2),CPACRE( 2), SLOP 30
* LP,JP,LPREV( 2),LABEL,DETENT(150),FRAC(150),OUT1(10,200) SLOP 31X
NEP1 = NE+1 SLOP 32
C ZERO OUT ARRAYS SLOP 33
DO 10 I=1,NE SLOP 34
IR(I) = 0 SLOP 35
10 JR(I) = C SLOP 36
C COMPUTE INTERNAL UPSTREAM ELEMENT NUMBERS. SLOP 37
DO 15 N=1,NE SLOP 38
DO 15 J=1,3 SLOP 39
IF (NUE(N,J)) 12,12,13 SLOP 40
C IF THERE IS NO UPSTREAM ELEMENT, ASSIGN ARTIFICIAL VALUE. SLOP 41
12 INUE(N,J) = NEP1 SLOP 42
GO TO 15 SLOP 43
13 L = NUE(N,J) SLOP 44
C UPSTREAM ELEMENT NUMBERS NOW FOUND FROM POINTING VECTOR, 'NIN'. SLOP 45
INUE(N,J) = NIN(L) SLOP 46
15 CONTINUE SLOP 47
C SEQUENCE ELEMENTS FOR COMPUTATION. SLOP 48
C FLOW MAY BE ROUTED IN ELEMENT I IF IT HAS BEEN ROUTED IN ALL SLOP 49
C UPSTREAM ELEMENTS. SLOP 50
DO 30 N=1,NE SLOP 51
I = 1 SLOP 52
17 IF(IR(I)) 20,20,18 SLOP 53
18 I = I+1 SLOP 54
IF (I-NE) 17,17,30 SLOP 55
20 DO 25 J=1,3 SLOP 56
IF (NUE(I,J)) 25,25,22 SLOP 57
22 L = INUE(I,J) SLOP 58
IF (IR(L)) 18,18,25 SLOP 59
25 CONTINUE SLOP 60
IR(I) = 1 SLOP 61

```

|                                                                               |         |
|-------------------------------------------------------------------------------|---------|
| JR(N) = I                                                                     | SLOP 62 |
| 30 CONTINUE                                                                   | SLOP 63 |
| WRITE (6,905)                                                                 | SLOP 64 |
| DO 50 I=1,NE                                                                  | SLOP 65 |
| L = JR(I)                                                                     | SLOP 66 |
| NT = NTYPE(I)                                                                 | SLOP 67 |
| 50 WRITE (6,906) NOE(I),NT,(NAME(J,NT),J=1,4),(NUE(I,J),J=1,3),I,             | SLOP 68 |
| 1NOE(L),L,(INUE(L,J),J=1,3)                                                   | SLOP 69 |
| RETURN                                                                        | SLOP 70 |
| 905 FORMAT ('OEXTERNAL TYPE DESCRIPTION      UPSTREAM ELEMENTS   INTERSLOP 71 |         |
| INAL'10X,'ELEMENT COMPTATION SEQUENCE'/' ELEMENT'29X,'1      2      3 SLOP 72 |         |
| 2 ELEMENT'9X,'EXTERNAL INTERNAL INTERNAL UPSTRFAM'/' NUMRER' SLOP 73          |         |
| 3 45X,'NUMBER'10X,'NUMBER      NUMBER      ELEMENT NUMBERS') SLOP 74          |         |
| 906 FORMAT (16,18,3X,4A4,3I5,19,116,110,19,2I5) SLOP 75                       |         |
| END                                                                           | SLOP 76 |

```

      FUNCTION VEL(QQ,AA)
C      ROUTINE TO COMPUTE VELOCITY GIVEN FLOW AND AREA.
      IF (AA.LE.0.0001) GO TO 10
      VEL = QQ/AA
      RETURN
10  VEL = 0.0
      RETURN
      END

```

```

VEL  1
VEL  2
VEL  3
VEL  4
VEL  5
VEL  6
VEL  7
VEL  8

```

```

C=====METCALF & EDDY, ENGINEERS' STORAGE MODEL=====TSTR 1
  SUBROUTINE TSTRCT TSTR 2
  COMMON/TABLES/KDEPTH(25),KCLASS(25),PSIMAX(15),ALFMAX(15), TSTR 3
1      NN(25),MM(25),ANORM(15,51),QNORM(15,51), TSTR 4
2      DNORM(15,51),AFACT(15),RFACT(15) TSTR 5
  COMMON A(160,2,2),Q(160,2,2),CPOLL(160,2,2,3),QMAX(160), TSTR 6X
1      CFULL(160),AFULL(160),DXDT(160),C1(160),SLOPE(160), TSTR 7
2      DIST(160),GEOM1(160),ROUGH(160),NOE(160),NUE(160,3), TSTR 8
3      INUE(160,3),NTYPE(160),JR(160),NKLASS,NE,NDT,EPSIL, TSTR 9
4      TIME,DT,M,KFULL,N,NOS,NPOLL,NPRINT,ITER, TSTR 10
5      CDWF(160),IOLC(160),PI(160),RNOFF(160),QINFIL(160), TSTR 11
6      WDW(160,3),PLUTO(160,3),IR(160),P2(160),NIN(1000), TSTR 12X
7      P5(160),P6(160),PT(160),SCF(160),BANKREL(160), TSTR 13
8      TITLE(40),NPE(20),NYN(20),NORDER(70),GEOM2(160), TSTR 14X
9      GEOM3(160),P4(160),SCOUR(160),KSTORE(160) TSTR 15
  COMMON BGDIN( 2,150),SSIN( 2,150),BODOUT,SSOUT,COLIN(2,150), TSTR 16X
*   QINST,CCUST,QINSTL( 2),QOUSTL( 2),STORL( 2),QOUTO( 2),STORO( 2),TSTR 17
*   NSTOR,KSTOR,IPRINT( 2),IPOL( 2),IFLOUO( 2),ICUST( 2),DEPMAX( 2),TSTR 18
*   ATERM( 2,11),AOZDT2( 2,11),BDEPTH( 2,11),BSTOR( 2,11),COLOUT, TSTR 19X
*   DUMSTR(11),DUMDEP(11), TSTR 20
*   KTSTEP,VOLIN( 2,150),VOLOUT( 2,150),STOR,CUMIN( 2),CUMOUT( 2), TSTR 21
*   SBOD( 2),SSS( 2),SCOL( 2), TSTR 22X
*   ISTMOD( 2),ISTTYP( 2),ISTOUT( 2), TSTR 23
*   QPUMP( 2),DSTART( 2),DSTOP( 2), TSTR 24
*   DTON( 2),STORMX( 2),DTPUMP( 2),DTMORE( 2),STORF( 2),APLAN( 2), TSTR 25
*   CLAND( 2),CSTOR( 2),CPS( 2),CTOTAL( 2),CPCUYD( 2),CPACRE( 2), TSTR 26
*   LP,JP,LPREV( 2),LABEL,DETENT(150),FRAC(150),OUT1(10,20) TSTR 27X
C TSTR 28X
  DIMENSION ADEPTH(11),AASURF(11),NUED(3) TSTR 29
  DIMENSION CQ1(150),CQ2(150) TSTR 30
  EQUIVALENCE (QO1(1),QMAX(1)),(QO2(1),QFULL(1)) TSTR 31
10000 IF (ASTOR.EQ. 0) STOP 10000 TSTR 32
C TSTR 33
11111 DO 8888 KSTOR=1, NSTOR TSTR 34
  WRITE(6,601) KSTOR TSTR 35
  601 FORMAT('1',//, 'OINPUT DATA FOR STORAGE UNIT NO.', I3, ' FOLLOWS' TSTR 36
*) TSTR 37
  READ(5,501) ISTMOD(KSTOR), ISTTYP(KSTOR), ISTOUT(KSTOR) TSTR 39X
  501 FORMAT(10I5) TSTR 40
C TSTR 41
C NCTE.. OPERATIONAL OPTIONS INDICATED BY '*' TSTR 42
C TSTR 43
C KSTOR = STORAGE UNIT NO. TSTR 44
C NOED = NO. OF ELEMENT TSTR 45
C NUED = NO. OF UPSTREAM ELEMENT TSTR 46
C TSTR 47
C ISTMOD = STORAGE MODE (ROUTING, HOLDING, ETC) TSTR 48
C * = 1 = IN-LINE TSTR 49
C = 2 = OFF-LINE TSTR 50
C = 3 = INTRASYSTEM TSTR 51
C = 4 = REROUTING TSTR 52
C ISTTYP = STORAGE STRUCTURE (NATURAL, TANK, BAG) TSTR 53
C * = 1 = IRREGULAR (NATURAL) RESERVOIR TSTR 54
C * = 2 = GEOMETRIC (REGULAR) RESERVOIR - COVERED TSTR 55
C * = 3 = GEOMETRIC (REGULAR) RESERVOIR - UNCOVERED TSTR 56
C = 4 = INTRASYSTEM TSTR 57
C = 5 = RUBBER BAG TSTR 58
C-EXCLUDED ISTEXS = EXCESS FLOW HANDLING (BYPASS, BACK-UP, FLOUTSTR 59
C-EXCLUDED = 1 = BYPASS FRACTION CONTINUOUSLY, UNDER GRATSTR 60
C-EXCLUDED = 2 = BYPASS FRACTION CONTINUOUSLY, CONTROLLETSTR 61

```

```

C-EXCLUDED          = 3 = BYPASS ALL, AFTER SURCHARGE BEGINS (CONTSTR 62
C-EXCLUDED          = 4 = BACK UP                                TSTR 63
C-EXCLUDED          = 5 = PASS THROUGH (UNDER HEAD)            TSTR 64
C-EXCLUDED          *   = 6 = FLOODS (AND SIMULATION TERMINATES) TSTR 65
C                    ISTOUT = OUTLET TYPE (GRAVITY, PUMP)        TSTR 66
C                    *   = 1 = GRAVITY WITH FIXED ORIFICE        TSTR 67
C                    *   = 2 = GRAVITY WITH FIXED WEIR           TSTR 68
C                    = 3 = GRAVITY WITH FIXED SIDE-WEIR         TSTR 69
C                    = 4 = GRAVITY WITH FIXED SIPHON             TSTR 70
C                    = 5 = NEW PUMPS                             TSTR 71
C                    *   = 6 = EXISTING PUMPS                    TSTR 72
C                    = 7 = ADJUSTABLE VALVE WITH GRAVITY        TSTR 73
C                    = 8 = SLUICE GATE WITH GRAVITY              TSTR 74
C                    *   = 9 = GRAVITY WITH FIXED WEIR AND ORIFICE TSTR 75
C-EXCLUDED          ISTBUP = BACK-UP EFFECTS (YES/NO)          TSTR 76
C-EXCLUDED          = 1 = NO                                     TSTR 77
C-EXCLUDED          = 2 = YES, UPSTREAM OF INLET POINT           TSTR 78
C-EXCLUDED          = 3 = YES, UPSTREAM OF OUTLET POINT          TSTR 79
C                                                            TSTR 80
      DO 950 KK=1,NE                                           TSTR 80A
950 IF(KSTORE(KK).EQ.KSTOR) GC TO 970                          TSTR 80B
      WRITE(6,604)                                             TSTR 80C
604 FORMAT('ELEMENT NO. TRACEBACK ERROR')                      TSTR 80D
      STOP                                                    TSTR 80E
970 WRITE(6,602) KSTOR, NCE(KK), (NUE(KK,II), II=1,3),        TSTR 81X
      *   ISTOUT(KSTOR), ISTMOD(KSTOR), ISTTYP(KSTOR)         TSTR 82
602 FORMAT('C', /, ' ', 5X,                                   TSTR 83
      *   'CHARACTERISTICS OF STORAGE UNIT NUMBER', I4, ' ARE' TSTR 84
      *   /, '0', 10X, 'ELEMENT NUMBER =', I4, 24X,          TSTR 85
      *   'U/S ARE', I4, ' ', I4, ' ', I4, ' ', I4, 23X,      TSTR 86
      *   'OUTLET TYPE =', I3, 12X,                            TSTR 87
      *   /, ' ', 10X, 'STORAGE MODE =', I3, 27X,             TSTR 88
      *   'STORAGE TYPE =', I3)                                TSTR 89
C                    IPOL = 0 = HYDRAULICS ONLY (NO SOLIDS)    TSTR 90
C                    = 1 = SOLIDS PRESENT                       TSTR 91
C                    IPRINT = 0 = NO PRINTOUT EACH TIME-STEP (SUMMARY POSTSTR 92
C                    = 1 = PRINTOUT ALL TIME-STEP SOLUTIONS (EACH TSTR 93
C                    READ(5,502) IPOL(KSTOR), IPRINT(KSTOR), ICOST(KSTOR) TSTR 94
502 FORMAT(3I10)                                               TSTR 95
      WRITE(6,603) IPOL(KSTOR), IPRINT(KSTOR), ICOST(KSTOR)   TSTR 96
603 FORMAT('0', 99X, 'IPOL =', I2, ' ', IPRINT =', I2, ' ', ICOST =', I2) TSTR 97
      NOED=NCE(KK)                                             TSTR 97A
C                                                            TSTR 98
C----- READ RESERVOIR PARAMETERS                             TSTR 99
C                                                            TSTR100
C                                                            TSTR101
C                    BRANCH TO STORAGE TYPE (ISTTYP)          TSTR102
C                                                            TSTR103
1000 K = ISTTYP(KSTOR)                                         TSTR104
      GO TO (1100,1200,1200, 904, 904), K                     TSTR105
C                                                            TSTR106
      GO TO (1100,1200,1200, 904, 904), K                     TSTR107
C                                                            TSTR108
C                    A) FOR IRREGULAR (NATURAL) RESERVOIR     TSTR109
C                                                            TSTR110
1100 READ(5,511) DEPMAX(KSTOR)                                 TSTR111
511 FORMAT(F10.2)                                              TSTR112
C                    DEPMAX = MAX. ALLOWABLE DEPTH IN RESERVOIR (FT.) TSTR113
      WRITE(6,611) DEPMAX(KSTOR)                               TSTR114
611 FORMAT('C', 15X, 'NATURAL RESERVOIR, WITH MAX. DEPTH =', F7.2, TSTR115

```



```

      *' FT.',10X, ' 11 DEPTH/AREA PARAMETERS ARE', //, TSTR116
      * ' ', 20X, 4('DEPTH(FT) AREA(SQ.FT)', 4X)) TSTR117
      READ(5,512) (ADEPTH(II), AASURF(II), II=1,11) TSTR118
512 FORMAT(4(F10.2, F10.0)) TSTR119
C      ADEPTH = DATA DEPTH (FT), 0 - DEPMAX. TSTR120
C      AASURF = DATA SURF. AREA OF NAT. RESERVOIR (SQ. FT.) TSTR121
      WRITE(6,612) (ADEPTH(II), AASURF(II), II=1,11) TSTR122
612 FORMAT(' ', 18X, F10.2, F10.0, 5X, F10.2, F10.0, 5X, TSTR123
      * F10.2, F10.0, 5X, F10.2, F10.0) TSTR124
      APLAN(KSTOR) = AASURF(11) TSTR125
      IF (ADEPTH(11) .LT. DEPMAX(KSTOR)) GO TO 903 TSTR126
      GO TO 2000 TSTR127
C TSTR128
C      B) FOR REGULAR (MAN-MADE) RESERVOIR TSTR129
C TSTR130
1200 READ(5,511) DEPMAX(KSTOR) TSTR131
      REAC(5,513) BASEA, BASEC, COTSLO TSTR132
513 FORMAT(2F10.0, F10.5) TSTR133
C      BASEA = BASE AREA OF RESERVOIR (SQ.FT.) TSTR134
C      BASEC = BASE CIRCUMF. OF RESERVOIR (FT.) TSTR135
C      COTSLO = COTAN OF SIDESLOPE ANGLE (HORIZ/VERT) TSTR136
      APLAN(KSTOR) = BASEA + BASEC*COTSLO*DEPMAX(KSTOR)*1.1 TSTR137
      WRITE(6,613) DEPMAX(KSTOR), BASEA, BASEC, COTSLO TSTR138
613 FORMAT('0', 15X, 'MAN-MADE RESERVOIR, WITH MAX. DEPTH =', F7.2, TSTR139
      * ' FT., AND CHARACTERISTICS', /, '0', 20X, TSTR140
      * 'BASE AREA =', F10.0, ' SQ.FT.', 7X, TSTR141
      * 'BASE CIRCUMF. =', F10.0, ' FT.', 6X, TSTR142
      * 'COT(SIDESLOPE) =', F10.5) TSTR143
C TSTR144
C----- READ OUTLET CONTROL DATA TSTR145
C TSTR146
C      BRANCH TO OUTLET TYPE (ISTOUT) TSTR147
C TSTR148
2000 K = ISTCUT(KSTOR) TSTR149
      GO TO (2100,2200, 906, 906, 906,2600, 906, 906,2900), K TSTR150
C TSTR151
C      A) OUTLET BY GRAVITY WITH FIXED ORIFICE TSTR152
C TSTR153
2100 READ(5,521) CDAOUT TSTR154
521 FORMAT(F10.3) TSTR155
C      CDAOUT = OUTLET ORIFICE AREA * DISCHARGE COEFFICIENT TSTR156
      WRITE(6,621) CDAOUT, DT TSTR157
621 FORMAT('0', 15X, 'RESERVOIR OUTLET CONTROL BY GRAVITY WITH FIXED OTSTR158
      *RIFICE', /, ' ', 15X, 'ORIFICE AREA*CD =', F10.3, ' SQ.FT, DT =', TSTR159
      *F10.2, ' SEC. (FROM INPUT HYDROGRAF )', /, ' ', 15X, 'NOTE..', /, TSTR160
      *' ', 18X, 'ORIFICE CENTERLINE ASSUMED AT TANK DEPTH = 0, WHEN STURTSTR161
      *AGE = 0') TSTR162
C TSTR163
      GO TO 3000 TSTR164
C TSTR165
C      B) OUTLET BY GRAVITY WITH FIXED WEIR TSTR166
C TSTR167
2200 READ(5,522) WEIRHT, WEIRL TSTR168
522 FORMAT(2F10.3) TSTR169
C      WEIRHT = RES. DEPTH (FT) WHEN SURF. AT WEIR ELEV. TSTR170
C      WEIRL = WEIR LENGTH (FT) TSTR171
C      WEIRQ = 3.33*(DEPTH-WEIRHT)**1.5 (CFS/FT) TSTR172
      WRITE(6,622) WEIRHT, WEIRL, DT TSTR173
622 FORMAT('0', 15X, 'RESERVOIR OUTLET CONTROL BY GRAVITY WITH FIXED WTSTR174
      *EIR', /, ' ', 15X, 'WEIR HEIGHT =', F6.2, ' FT, WEIR LENGTH =', TSTR175

```

```

* F7.2, ' FT, DT =', F10.2, ' SEC. (FROM INPUT HYDROGRAPH)')
C
GO TO 3000
C
C
C
C
F) OUTLET BY EXISTING PUMPS
2600 READ(5,526) QPUMP(KSTOR),DSTART(KSTOR),DSTOP(KSTOR)
526 FORMAT(3F10.3)
C
QPUMP = CONSTANT PUMPED OUTFLOW RATE (CFS)
C
DSTART = RESERVOIR DEPTH AT START OF PUMPING (FT)
C
DSTOP = RESERVOIR DEPTH AT END OF PUMPING (FT)
C
WRITE(6,626) QPUMP(KSTOR),DSTART(KSTOR),DSTOP(KSTOR)
626 FORMAT('C', 15X, 'RESERVOIR OUTFLOW BY FIXED-RATE PUMPING', /,
* ' ', 15X, 'PUMPING RATE =', F7.2, ' CFS, PUMPING START',
* ' DEPTH =', F6.2, ' FT, PUMPING STOP DEPTH =', F6.2, ' FT')
IF (DSTART(KSTOR) .GT. DSTOP(KSTOR)) GO TO 3000
C
DSTOP(KSTOR) = DSTART(KSTOR) - 0.5
WRITE(6,627)
627 FORMAT('O***** WARNING. STOP DEPTH SHOULD BE LESS THAN START',
* ' DEPTH', /, ' ', 10X, 'RESET STOP DEPTH TO', F6.2, ' FT')
GO TO 3000
C
C
C
I) OUTLET BY GRAVITY WITH FIXED WEIR AND ORIFICE
2900 READ (5,529) WEIRHT,WEIRL,CDAOUT,ORIFHT
529 FORMAT (8F10.5)
C
WEIRHT = RES.DEPTH (FT) WHEN SURF. AT WEIR ELEV.
C
WEIRL = WEIR LENGTH (FT)
C
CDAOUT = OUTLET ORIFICE AREA * DISCHARGE COEFFICIENT
C
ORIFHT = ORIF. CENTERLINE ELEV. ABOVE ZERO DEPTH(FT)
C
WRITE (6,629) WEIRHT,WEIRL,DT,CDAOUT,ORIFHT
629 FORMAT (/16X,'RESERVOIR OUTLET CONTROL BY GRAVITY WITH FIXED ORIFIT
ICE AND FIXED WEIR'/16X,'WEIR HEIGHT=',F6.2,' FT, WEIR LENGTH='
2 F7.2,' FT, DT=',F10.2,' SEC. (FROM INPUT HYDROGRAPH)'/16X,
3'ORIFICE AREA*CD='F10.3,' SQ.FT, HEIGHT OF ORIFICE CENTERLINE ABOVE
4E ZERO DEPTH='F6.2,' FT')
J = NIN(NDED)
J = GEOM3(J)
WRITE (6,630) J
630 FORMAT (16X,'OUTFLOW FROM ORIFICE IS TO ELEMENT NUMBER'15)
GO TO 3000
C
C
C
C
END OF INPUT PARAMETERS FOR STORAGE UNITS
C
C
C
C
----- COMPUTE DEPTH/STORAGE RELATION (ARRAY) -----
C
C
C
C
NOTE
C
RUBBER BAG AND INTRASYSTEM DO NOT HAVE 'DEPTH'
C
C
C
C
BRANCH TO STORAGE TYPE (ISTTYP)
C
3000 K = ISTTYP(KSTOR)
GO TO (3100,3200,3200, 904, 904), K
C
C
C
C
A) FOR 'NATURAL' RESERVOIR
C
3100 DDEPTH = DEPMAX(KSTOR)/10.0
J = ISTCLT(KSTOR)

```

|                                                                    |         |
|--------------------------------------------------------------------|---------|
| IF (J.EQ.2.OR.J.EQ.3.OR.J.EQ.9) DDEPTH = WEIRHT/3.0                | TSTR236 |
| DUMDEP(1) = 0.0                                                    | TSTR237 |
| DEPTH = 0.0                                                        | TSTR238 |
| C                                                                  | TSTR239 |
| CALL TINTRP(ADEPTH, AASURF, 11, DEPTH, AREA, KFLAG)                | TSTR240 |
| C                                                                  | TSTR241 |
| IF (KFLAG .NE. 0) GO TO 901                                        | TSTR242 |
| AREA2 = AREA                                                       | TSTR243 |
| DUMSTR(1) = 0.0                                                    | TSTR244 |
| DO 3150 II=2,11                                                    | TSTR245 |
| AREA1 = AREA2                                                      | TSTR246 |
| IF ((J.EQ.2.OR.J.EQ.3.OR.J.EQ.9).AND.II.EQ.5) DDEPTH =             | TSTR247 |
| * (DEPMAX(KSTOR) - WEIRHT)/7.0                                     | TSTR248 |
| DUMDEP(II) = DUMDEP(II-1) + DDEPTH                                 | TSTR249 |
| DEPTH = DUMDEP(II)                                                 | TSTR250 |
| C                                                                  | TSTR251 |
| CALL TINTFP(ADEPTH, AASURF, 11, DEPTH, AREA, KFLAG)                | TSTR252 |
| C                                                                  | TSTR253 |
| IF (KFLAG .NE. 0) GO TO 901                                        | TSTR254 |
| AREA2 = AREA                                                       | TSTR255 |
| 3150 DUMSTR(II) = DUMSTR(II-1) + 0.5*DDEPTH*(AREA1+AREA2)          | TSTR256 |
| WRITE(6,631)                                                       | TSTR257 |
| 631 FORMAT('C', 20X, 4('DEPTH(FT) STOR(CU.FT)', 4X))               | TSTR258 |
| WRITE(6,632) (DUMDEP(II), DUMSTR(II), II=1,11)                     | TSTR259 |
| 632 FORMAT(' ', 18X, F9.2, F11.0, 5X, F9.2, F11.0, 5X,             | TSTR260 |
| * F9.2, F11.0, 5X, F9.2, F11.0)                                    | TSTR261 |
| GO TO 3300                                                         | TSTR262 |
| C                                                                  | TSTR263 |
| C----- B) FOR MAN-MADE RESERVOIR                                   | TSTR264 |
| C                                                                  | TSTR265 |
| 3200 DDEPTH = DEPMAX(KSTOR)/10.0                                   | TSTR266 |
| J = ISTOLT(KSTOR)                                                  | TSTR267 |
| IF (J.EQ.2.OR.J.EQ.3.OR.J.EQ.9) DDEPTH = WEIRHT/3.0                | TSTR268 |
| DUMDEP(1) = 0.0                                                    | TSTR269 |
| DUMSTR(1) = 0.0                                                    | TSTR270 |
| DO 3250 II=2,11                                                    | TSTR271 |
| IF ((J.EQ.2.OR.J.EQ.3.OR.J.EQ.9).AND.II.EQ.5) DDEPTH =             | TSTR272 |
| * (DEPMAX(KSTOR) - WEIRHT)/7.0                                     | TSTR273 |
| DUMDEP(II) = DUMDEP(II-1) + DDEPTH                                 | TSTR274 |
| 3250 DUMSTR(II) = (BASEA + 0.5*BASEC*DUMDEP(II)*CJTSLO)*DUMDEP(II) | TSTR275 |
| WRITE(6,631)                                                       | TSTR276 |
| WRITE(6,632) (DUMDEP(II), DUMSTR(II), II=1,11)                     | TSTR277 |
| C                                                                  | TSTR278 |
| 3300 DO 3350 II=1,11                                               | TSTR279 |
| BSTOR(KSTOR,II) = DUMSTR(II)                                       | TSTR280 |
| 3350 BDEPTH(KSTOR,II) = DUMDEP(II)                                 | TSTR281 |
| C                                                                  | TSTR282 |
| C                                                                  | TSTR283 |
| C                                                                  | TSTR284 |
| C----- BRANCH TO STORAGE MODE (ISTMOD) -----                       | TSTR285 |
| C                                                                  | TSTR286 |
| K = ISTMOD(KSTOR)                                                  | TSTR287 |
| GO TO (4000, 905, 905, 905), K                                     | TSTR288 |
| C                                                                  | TSTR289 |
| C----- COMPUTE AND PRINT ROUTING PARAMETERS -----                  | TSTR290 |
| C                                                                  | TSTR291 |
| C                                                                  | TSTR292 |
| C                                                                  | TSTR293 |
| 4000 K = ISTOUT(KSTOR)                                             | TSTR294 |
| GO TO (4100,4200, 906, 906, 8000,4600, 906, 906,4900), K           | TSTR295 |

|         |                                                                    |         |
|---------|--------------------------------------------------------------------|---------|
| C       |                                                                    | TSTR296 |
| C-----  | COMPUTE AND PRINT ROUTING PARAMETERS                               | TSTR297 |
| C       |                                                                    | TSTR298 |
| C-----  | A) FOR ORIFICE OUTLET                                              | TSTR299 |
| C       |                                                                    | TSTR300 |
| 4100    | DT2 = 0.5*CT                                                       | TSTR301 |
| DO 4150 | II=1,11                                                            | TSTR302 |
|         | DEPTH = CUMDEP(II)                                                 | TSTR303 |
|         | STOR = DUMSTR(II)                                                  | TSTR304 |
|         | QCUT = CDAGUT*SQRT(64.4*DEPTH)                                     | TSTR305 |
| C       |                                                                    | TSTR306 |
|         | AO2DT2(KSTOR,II) = QCUT*DT2                                        | TSTR307 |
| 4150    | ATERM(KSTOR,II) = QCUT*DT2 + STOR                                  | TSTR308 |
|         | WRITE(6,641) DT, (ATERM(KSTOR,II), II=1,11),                       | TSTR309 |
|         | * (AO2DT2(KSTOR,II), II=1,11)                                      | TSTR310 |
| 641     | FORMAT('O', 20X, 'THE TWO SETS OF 11 STORAGE PARAMETERS, FOR DT =' | TSTR311 |
| *       | , F10.3, ' SEC., ARE..', /, 'O', 11X, 'ATERM..',                   | TSTR312 |
| *       | 11F10.0, /, ' ', 11X, 'AO2DT2..', 11F10.0)                         | TSTR313 |
|         | GO TO 8CCC                                                         | TSTR314 |
| C       |                                                                    | TSTR315 |
| C-----  | B) FOR WEIR OUTLET                                                 | TSTR316 |
| C       |                                                                    | TSTR317 |
| 4200    | DT2 = 0.5*DT                                                       | TSTR318 |
| DO 4250 | II=1,11                                                            | TSTR319 |
|         | DEPTH = DUMDEP(II)                                                 | TSTR320 |
|         | STOR = DUMSTR(II)                                                  | TSTR321 |
|         | H = DEPTH - WEIRHT                                                 | TSTR322 |
|         | QCUT = 0.0                                                         | TSTR323 |
|         | IF (H .GT. 0.0) QCUT = WEIRL*3.33*H*SQRT(H)                        | TSTR324 |
| C       |                                                                    | TSTR325 |
|         | AO2DT2(KSTOR,II) = QCUT*DT2                                        | TSTR326 |
| 4250    | ATERM(KSTOR,II) = QCUT*DT2 + STOR                                  | TSTR327 |
|         | WRITE(6,641) DT, (ATERM(KSTOR,II), II=1,11),                       | TSTR328 |
| *       | (AO2DT2(KSTOR,II), II=1,11)                                        | TSTR329 |
|         | GO TO 80CC                                                         | TSTR330 |
| C       |                                                                    | TSTR331 |
| C-----  | CHECK BUFFER VOLUME FOR PUMPED OUTFLOW                             | TSTR332 |
| C       |                                                                    | TSTR333 |
| 4600    | DSTRT = DSTART(KSTOR)                                              | TSTR334 |
|         | CALL TINTRP(DUMDEP, DUMSTR, 11, DSTRT, STORHI, KFLAG)              | TSTR335 |
| C       |                                                                    | TSTR336 |
|         | IF (KFLAG .NE. 0) GO TO 901                                        | TSTR337 |
| C       |                                                                    | TSTR338 |
|         | DSTP = DSTOP(KSTOR)                                                | TSTR339 |
|         | CALL TINTRP(CUMDEP, DUMSTR, 11, DSTP, STORLO, KFLAG)               | TSTR340 |
| C       |                                                                    | TSTR341 |
|         | IF (KFLAG .NE. 0) GO TO 901                                        | TSTR342 |
|         | STORDV = STORHI - STORLO                                           | TSTR343 |
|         | PUMPDV = QPUMP(KSTOR)*DT                                           | TSTR344 |
|         | DVR = STORCV/PUMPDV                                                | TSTR345 |
|         | IF (DVR .LT. 1.0) GO TO 4650                                       | TSTR346 |
| C       |                                                                    | TSTR347 |
|         | WRITE(6,646) DVR                                                   | TSTR348 |
| 646     | FORMAT('O', 15X, 'STORAGE BETWEEN LEVELS DSTART AND DSTOP =',      | TSTR349 |
| *       | F7.3, ' TIMES (QPUMP*DT)')                                         | TSTR350 |
|         | GO TO 8000                                                         | TSTR351 |
| C       |                                                                    | TSTR352 |
| 4650    | WRITE(6,647) STORHI, STORLO, STORDV, PUMPDV                        | TSTR353 |
| 647     | FORMAT('O', 20X, 'AT LEVEL DSTART, STORAGE =', F10.0, ' CU.FT', /, | TSTR354 |
| *       | ' ', 20X, 'AT LEVEL DSTOP, STORAGE =', F10.0, ' CU.FT', /,         | TSTR355 |

```

*      ' ',20X, 'DIFFERENCE = BUFFER STORAGE =', F10.0, ' CU.FT',/,TSTR356
*      '0',16X, 'CF. VOLUME PUMPED / TIME-STEP  =', F10.0, ' CU.FT',/,TSTR357
*      '0',10X, 'A RELIABLE MDEL REQUIRES THE VOLUME PUMPED / TIME-',TSTR358
*      'STEP TO BE LESS THAN THE BUFFER STORAGE', /, TSTR359
*      ' ',10X, 'THEREFORE ONE OF THE FOLLOWING AMENDMENTS SHOULD ', TSTR360
*      'PROBABLY BE MADE TO THE INPUT DATA -', /, TSTR361
*      ' ',15X, 'A) REDUCE QPUMP RATE', /, TSTR362
*      ' ',15X, 'B) REDUCE DSTOP LEVEL', /, TSTR363
*      ' ',15X, 'C) INCREASE DSTART LEVEL', /, TSTR364
*      ' ',15X, 'D) INCREASE RESERVOIR PLAN AREAS', /, TSTR365
*      '0', 2X, '**** FOR THE ABOVE REASONS, THE FOLLOWING OUTPUT',TSTR366
*      'IS NOT NECESSARILY RELIABLE') TSTR367
GO TO 8000 TSTR368

C TSTR369
C----- 1) FOR WEIR AND ORIFICE OUTLETS COMBINED TSTR370
C TSTR371
4900 DT2 = 0.5*DT TSTR372
DO 4950 II=1,11 TSTR373
  DEPTH = CUMCEP(II) TSTR374
  STOR = CLMSTR(II) TSTR375
  QOUT = 0.0 TSTR376
  DD = DEPTH-GRIFHT TSTR377
  IF (DD.GT.0.0) QOUT=QOUT+CDAOUT*SQRT(64.4*DD) TSTR378
  OUT1(KSTCR,II) = QOUT*DT2 TSTR379
  H = DEPTH-WEIRHT TSTR380
  IF (H.GT.0.0) QOUT=QOUT+WEIRL*3.33*H*SQRT(H) TSTR381
  AO2DT2(KSTCR,II) = QOUT*DT2 TSTR382
4950 ATERM(KSTCR,II) = QOUT*DT2+STOR TSTR383
  WRITE (6,641) DT,(ATERM(KSTCR,II),II=1,11), TSTR384
  1 (AO2DT2(KSTCR,II),II=1,11) TSTR385
GO TO 8000 TSTR386

C TSTR387
C----- READ RESERVOIR INITIAL CONDITIONS ----- TSTR388
C TSTR389
C      STOR0 = STORAGE (CU.FT.) AT T = 0 TSTR390
C      QCUTO = OUTFLOW RATE (CFS) AT T = 0 TSTR391
8000 READ(5,551) STCR0(KSTOR), QCUTO(KSTOR) TSTR392
551 FORMAT(2F10.2) TSTR393
STORL(KSTOR)=STOR0(KSTOR) TSTR394
C TSTR395
C----- READ STORAGE UNIT UNIT COSTS ----- TSTR396
C TSTR397
C      READ(5,561) CPCUYD(KSTOR), CPACRE(KSTOR), CPS(KSTOR) TSTR398
561 FORMAT(F10.2, 2F10.0) TSTR399
C TSTR400
C      IFLOC(KSTOR) = 0 TSTR401
C TSTR402
8888 CONTINUE TSTR403
C TSTR404
GO TO 5555 TSTR405
C TSTR406
C----- ERROR MESSAGES ----- TSTR407
C TSTR408
901 IF (KFLAG .EQ. 10) GO TO 902 TSTR409
  WRITE(6,691) TSTR410
691 FORMAT('0 *** TERMINATE - INPUT TO TINTRP PROCEDURE IS LESS THAN LTSTR411
  *LOWEST VALUE ON CURVE (IN SUBRT. TSTRDT)') TSTR412
  STOP TSTR413
C TSTR414
902 WRITE(6,692) TSTR415

```

|        |                                                                     |         |
|--------|---------------------------------------------------------------------|---------|
| 692    | FORMAT('O *** TERMINATE - INPUT TO TINTRP PROCEDURE IS GREATER THAN | TSTR416 |
|        | *N LARGEST VALUE ON CURVE (IN SUBRT. TSTROT)')                      | TSTR417 |
|        | STOP                                                                | TSTR418 |
| C      |                                                                     | TSTR419 |
| 903    | WRITE(6,693) KSTCR, DEPMAX(KSTOR), ADEPTH(11)                       | TSTR420 |
| 693    | FORMAT('C *** FOR RESERVOIR NO.', I4,                               | TSTR421 |
|        | * , THE MAX. DEPTH, DEPMAX =',                                      | TSTR422 |
|        | * F7.2, ' FT, IS OUTSIDE DEPTH PARAMETER RANGE. LARGEST DEPTH PARAT | TSTR423 |
|        | *METER, ADEPTH(11) =', F7.2)                                        | TSTR424 |
|        | STOP                                                                | TSTR425 |
| C      |                                                                     | TSTR426 |
| 904    | WRITE(6,694) ISTTYP(KSTOR), KSTOR                                   | TSTR427 |
| 694    | FORMAT('O *** ISTTYP =', I3, ' IN RESERVOIR NO.', I4,               | TSTR428 |
|        | * ', IS OF A TYPE NOT PRESENTLY MODELED')                           | TSTR429 |
|        | STOP                                                                | TSTR430 |
| C      |                                                                     | TSTR431 |
| 905    | WRITE(6,695) ISTMOD(KSTOR), KSTOR                                   | TSTR432 |
| 695    | FORMAT('O *** ISTMOD =', I3, ' IN RESERVOIR NO.', I4,               | TSTR433 |
|        | * ', IS OF A TYPE NOT PRESENTLY MODELED')                           | TSTR434 |
|        | STOP                                                                | TSTR435 |
| C      |                                                                     | TSTR436 |
| 906    | WRITE(6,696) ISTCUT(KSTOR), KSTOR                                   | TSTR437 |
| 696    | FORMAT('O *** ISTCUT =', I3, ' IN RESERVOIR NO.', I4,               | TSTR438 |
|        | * ', IS OF A TYPE NOT PRESENTLY MODELED')                           | TSTR439 |
|        | STOP                                                                | TSTR440 |
| C      |                                                                     | TSTR441 |
| 907    | WRITE(6,697) ISTINF, KSTOR                                          | TSTR442 |
| 697    | FORMAT('O *** ISTINF =', I3, ' IN RESERVOIR NO.', I4,               | TSTR443 |
|        | * ', IS OF A TYPE NOT PRESENTLY MODELED')                           | TSTR444 |
|        | STOP                                                                | TSTR445 |
| C      |                                                                     | TSTR446 |
| C      |                                                                     | TSTR447 |
| 9999   | RETURN                                                              | TSTR448 |
|        | END                                                                 | TSTR449 |
| C===== |                                                                     | TSTR450 |

|                                                                       |          |
|-----------------------------------------------------------------------|----------|
| SUBROUTINE TSTORG                                                     | TSTO 1   |
| COMMON/TABLES/KDEPTH(25),KCLASS(25),PSIMAX(15),ALFMAX(15),            | TSTO 2   |
| 1 NN(25),MM(25),ANORM(15,51),QNORM(15,51),                            | TSTO 3   |
| 2 DNORM(15,51),AFAC(15),RFAC(15)                                      | TSTO 4   |
| COMMON A(160,2,2),Q(160,2,2),CPOLL(160,2,2,3),QMAX(160),              | TSTO 5X  |
| 1 QFULL(160),AFULL(160),DXDT(160),C1(160),SLOPE(160),                 | TSTO 6   |
| 2 DIST(160),GEOM1(160),ROUGH(160),NOE(160),NUE(160,3),                | TSTO 7   |
| 3 INUE(160,3),NTYPE(160),JR(160),NKLASS,NE,NDT,EPSIL,                 | TSTO 8   |
| 4 TIME,DT,M,KFULL,N,NOS,NPOLL,NPRINT,ITER,                            | TSTO 9   |
| 5 QCWF(160),IOLD(160),P1(160),RNUFF(160),QINFIL(160),                 | TSTO 10  |
| 6 WCWF(160,3),PLUTO(160,3),IR(160),P2(160),NIN(1000),                 | TSTO 11X |
| 7 P5(160),P6(160),P7(160),SCF(160),BARREL(160),                       | TSTO 12  |
| 8 TITLE(40),NFE(20),NPN(20),NORDER(70),GEOM2(160),                    | TSTO 13X |
| 9 GEOM3(160),P4(160),SCOUR(160),KSTORE(160)                           | TSTO 14  |
| COMMON BODIN( 2,150),SSIN( 2,150),BUDOUT,SSOUT,CULIN(2,150),          | TSTO 15X |
| * QINST,COLST,QINSTL( 2),QOUSTL( 2),STORL( 2),QOUTO( 2),STORO( 2),    | TSTO 16  |
| * NSTOR,KSTOR,IPRINT( 2),IPOL( 2),IFLOOD( 2),ICOST( 2),DEPMAX( 2),    | TSTO 17  |
| * ATERM( 2,11),AO2DT2( 2,11),BDEPTH( 2,11),BSTOR( 2,11),COLOUT,       | TSTO 18X |
| * DUMSTR(11),DUMDEP(11),                                              | TSTO 19  |
| * KTSTEP,VOLIN( 2,150),VOLOUT( 2,150),STUR,CUMIN( 2),CUMOUT( 2),      | TSTO 20  |
| * SBOD( 2),SSS( 2),SCOL( 2),                                          | TSTO 21X |
| * ISTMOC( 2),ISTTYP( 2),ISTOUT( 2),                                   | TSTO 22  |
| * QPUMP( 2),DSTART( 2),DSTOP( 2),                                     | TSTO 23  |
| * DTOK( 2),STGRMX( 2),DTPUMP( 2),DTMORE( 2),STORF( 2),APLAN( 2),      | TSTO 24  |
| * CLANC( 2),CSTCR( 2),CPS( 2),CTOTAL( 2),CPCUYD( 2),CPACRE( 2),       | TSTO 25  |
| * LP,JP,LPREV( 2),LABEL,DETENT(150),FRAC(150),OUT1(10,200)            | TSTO 26X |
| C                                                                     | TSTO 27X |
| DIMENSION DEPTH1( 2)                                                  | TSTO 28  |
| DIMENSION COL(150),QO2(150)                                           | TSTO 29  |
| EQUIVALENCE (QO1(1),QMAX(1)),(QO2(1),QFULL(1))                        | TSTO 30  |
| C                                                                     | TSTO 31  |
| IF (KTSTEP.GT. 1 .OR. KSTOR.GT. 1) GO TO 1300                         | TSTO 32  |
| C                                                                     | TSTO 33  |
| DO 1050 KSTDUM=1,NSTCR                                                | TSTO 34  |
| IF (IPRINT(KSTDUM).GT. 0) GO TO 1100                                  | TSTO 35  |
| 1050 CONTINUE                                                         | TSTO 36  |
| GO TO 1200                                                            | TSTO 37  |
| C                                                                     | TSTO 38  |
| 1100 WRITE(6,601) NDT                                                 | TSTO 39  |
| 601 FORMAT('1STORAGE SOLUTION FOR', 14, ' TIME-STEPS FOLLOWS',        | TSTO 40  |
| * ' , ON A STEP-BY-STEP BASIS', //)                                   | TSTO 41  |
| WRITE(6,602)                                                          | TSTO 42  |
| 602 FORMAT('OU STP TIME INFLOW OUTFLOW STORAGE DEPTH', 2X, 'IN: BOD', | TSTO 43  |
| * 5X, 'SS STOR: BOD', 7X, 'SS', 4X, 'BOD', 5X, 'SS OUT: BOD',         | TSTO 44  |
| * 5X, 'SS', 4X, 'BOD', 5X, 'SS', 2X, ' J L', /,                       | TSTO 45  |
| * ' A NC (MIN) (CFS) (CFS) (CU.FT) (FT.)', 5X,                        | TSTO 46  |
| * '(LB) (LB)', 6X, '(LB)', 5X, '(LB) (MG/L) (MG/L)', 5X,              | TSTO 47  |
| * '(LB) (LB) (MG/L) (MG/L)', 2X, ' P P', /)                           | TSTO 48  |
| C                                                                     | TSTO 49  |
| 1200 DO 1250 KSTDUM=1,NSTCR                                           | TSTO 50  |
| STORMX(KSTDUM) = 0.0                                                  | TSTO 51  |
| LPREV(KSTDUM) = 1                                                     | TSTO 52  |
| SBOD(KSTDUM) = 0.0                                                    | TSTO 53  |
| SSS(KSTDUM) = 0.0                                                     | TSTO 54  |
| SCUL(KSTDUM) = 0.0                                                    | TSTO 54A |
| STOR = STCRO(KSTDUM)                                                  | TSTO 55  |
| DO 1230 I=1,11                                                        | TSTO 56  |
| DUMSTR(I) = BSTOR(KSTDUM,I)                                           | TSTO 57  |
| 1230 DUMDEP(I) = BDEPTH(KSTDUM,I)                                     | TSTO 58  |
| C                                                                     | TSTO 59  |

|                                                                |          |
|----------------------------------------------------------------|----------|
| CALL TINTERP(DUMSTR,DUMDEP,11,STOR,DEPTH,KFLAG)                | TST0 60  |
| C                                                              | TST0 61  |
| IF (KFLAG .EQ. -10) GO TO 901                                  | TST0 62  |
| IF (KFLAG .EQ. 10) GO TO 8000                                  | TST0 63  |
| C                                                              | TST0 64  |
| DEPTH1(KSTDUM) = DEPTH                                         | TST0 65  |
| IF (DEPTH .GT. DEPMAX(KSTDUM)) GO TO 8000                      | TST0 66  |
| IF (IPRINT(KSTDUM) .LT. 1) GO TO 1250                          | TST0 67  |
| WRITE(6,603) KSTDUM, QOUTO(KSTDUM), STORO(KSTDUM), DEPTH       | TST0 68  |
| 603 FORMAT(' ', 11, ' 0 0.0 0.0', F7.1, F10.0, F6.2)           | TST0 69  |
| 1250 CONTINUE                                                  | TST0 70  |
| WRITE(6,604)                                                   | TST0 71  |
| 604 FORMAT(' ')                                                | TST0 72  |
| C-----                                                         | TST0 73  |
| 1300 DO 1350 II=1,11                                           | TST0 74  |
| DUMSTR(II) = BSTOR(KSTOR,II)                                   | TST0 75  |
| 1350 DUMDEP(II) = BDEPTH(KSTOR,II)                             | TST0 76  |
| RKTSTP = KTSTEP                                                | TST0 77  |
| TIME2M = DT*RKTSTP/60.0                                        | TST0 78  |
| BODOUT = 0.0                                                   | TST0 79  |
| SSOUT = 0.0                                                    | TST0 80  |
| COLOUT = 0.0                                                   | TST0 80A |
| C                                                              | TST0 81  |
| IF (ISTCUT(KSTOR) .EQ. 5 .OR. ISTOUT(KSTOR) .EQ. 6) GO TO 2000 | TST0 82  |
| GO TO 3000                                                     | TST0 83  |
| C-----                                                         | TST0 84  |
| 2000 IF (KTSTEP .GT. 1) GO TO 2100                             | TST0 85  |
| QINSTL(KSTOR) = 0.0                                            | TST0 86  |
| QOUSTL(KSTOR) = QOUTO(KSTOR)                                   | TST0 87  |
| STORL(KSTOR) = STORO(KSTOR)                                    | TST0 88  |
| CUMIN(KSTOR) = 0.0                                             | TST0 89  |
| CUMCUT(KSTOR) = 0.0                                            | TST0 90  |
| STOR = STORO(KSTOR)                                            | TST0 91  |
| LP = 0                                                         | TST0 92  |
| JP = 0                                                         | TST0 93  |
| DTON(KSTOR) = 0.0                                              | TST0 94  |
| C                                                              | TST0 95  |
| 2100 VOLIN(KSTOR,KTSTEP) = 0.5*DT*(QINSTL(KSTOR) + QINST)      | TST0 96  |
| VOLCUZ = DT*QOUSTL(KSTOR)                                      | TST0 97  |
| STORZ = STORL(KSTOR) + VOLIN(KSTOR,KTSTEP) - VOLCUZ            | TST0 98  |
| C                                                              | TST0 99  |
| CALL TINTERP(DUMSTR,DUMDEP,11,STORZ,DEPTHZ,KFLAG)              | TST0100  |
| C                                                              | TST0101  |
| IF (KFLAG .EQ. -10) GO TO 901                                  | TST0102  |
| IF (KFLAG .EQ. 10) GO TO 8000                                  | TST0103  |
| C                                                              | TST0104  |
| *** THE FOLLOWING STATEMENTS, ABOVE 2150, ARE TEMPORARY        | TST0105  |
| C                                                              | TST0106  |
| IF (QINST .GT. QPUMP(KSTOR) .AND. QOUSTL(KSTOR) .EQ. 0.0)      | TST0107  |
| * GO TO 2120                                                   | TST0108  |
| GO TO 2150                                                     | TST0109  |
| 2120 FON = (QINST-QPUMP(KSTOR))/(QINST-QINSTL(KSTOR))          | TST0110  |
| DSTART(KSTOR) = DEPTHZ*(1.0-FON) + FON*DEPTH1(KSTOR)           | TST0111  |
| DTON(KSTOR) = DTON(KSTOR) + FON                                | TST0112  |
| WRITE(6,605) DSTART(KSTOR), KSTOR                              | TST0113  |
| 605 FORMAT(' ', 60X, 'NEW DSTART =', F6.2, ' FT.',             | TST0114  |
| * ' , IN UNIT NO.', 13)                                        | TST0115  |
| C                                                              | TST0116  |
|                                                                | TST0117  |
|                                                                | TST0118  |



|                                                                  |         |
|------------------------------------------------------------------|---------|
| 2150 IF (QOUSTL(KSTOR) .EQ. 0.0 .AND. DEPTHZ .GT. DSTART(KSTOR)) | TST0119 |
| * GO TO 2200                                                     | TST0120 |
| IF (QOUSTL(KSTOR) .GT. 0.0 .AND. DEPTHZ .LT. DSTOP(KSTOR) )      | TST0121 |
| * GO TO 2300                                                     | TST0122 |
| QOUST = QOUSTL(KSTOR)                                            | TST0123 |
| VOLOUT(KSTOR,KTSTEP) = VOLOUZ                                    | TST0124 |
| STOR = STCRZ                                                     | TST0125 |
| DEPTH = DEPTHZ                                                   | TST0126 |
| GO TO 2500                                                       | TST0127 |
| C                                                                | TST0128 |
| 2200 FON = (DEPTHZ - DSTART(KSTOR))/(DEPTHZ - DEPTH(KSTOR))      | TST0129 |
| QOUST = QPUMP(KSTOR)                                             | TST0130 |
| GO TO 2400                                                       | TST0131 |
| C                                                                | TST0132 |
| 2300 FON = (DEPTH(KSTOR) - DSTOP(KSTOR))/(DEPTH(KSTOR) - DEPTHZ) | TST0133 |
| QOUST = 0.0                                                      | TST0134 |
| C                                                                | TST0135 |
| 2400 IF (FON .LT. 0.0) GO TO 8100                                | TST0136 |
| VOLOUT(KSTOR,KTSTEP) = FON*DT*QPUMP(KSTOR)                       | TST0137 |
| STOR = STORL(KSTOR) + VOLIN(KSTOR,KTSTEP)-VOLOUT(KSTOR,KTSTEP)   | TST0138 |
| C                                                                | TST0139 |
| CALL TINTRP(DUMSTR,DUMDEP,11,STOR,DEPTH,KFLAG)                   | TST0140 |
| C                                                                | TST0141 |
| IF (KFLAG .EQ. -10) GO TO 901                                    | TST0142 |
| IF (KFLAG .EQ. 10) GO TO 8000                                    | TST0143 |
| C                                                                | TST0144 |
| 2500 IF (DEPTH .GT. DEPMAX(KSTOR)) GO TO 8000                    | TST0145 |
| CUMIN(KSTOR) = CUMIN(KSTOR) + VOLIN(KSTOR,KTSTEP)                | TST0146 |
| CUMOUT(KSTOR) = CUMOUT(KSTOR) + VOLOUT(KSTOR,KTSTEP)             | TST0147 |
| QINSTL(KSTOR) = QINST                                            | TST0148 |
| QOUSTL(KSTOR) = QOUST                                            | TST0149 |
| DEPTH(KSTOR) = DEPTH                                             | TST0150 |
| STORL(KSTOR) = STOR                                              | TST0151 |
| IF (KTSTEP .EQ. NDT) STORF(KSTOR) = STOR                         | TST0152 |
| IF (STOR .GT. STORMX(KSTOR)) STORMX(KSTOR) = STOR                | TST0153 |
| IF (QOUST .GT. 0.0) DTON(KSTOR) = DTON(KSTOR) + 1.0              | TST0154 |
| GO TO 4000                                                       | TST0155 |
| C-----                                                           | TST0156 |
| C                                                                | TST0157 |
| 3000 CALL TSRCUT                                                 | TST0158 |
| C                                                                | TST0159 |
| IF (STOR .GT. STORMX(KSTOR)) STORMX(KSTOR) = STOR                | TST0160 |
| C                                                                | TST0161 |
| CALL TINTRP(DUMSTR,DUMDEP,11,STOR,DEPTH,KFLAG)                   | TST0162 |
| C                                                                | TST0163 |
| IF (KFLAG .EQ. -10) GO TO 901                                    | TST0164 |
| IF (KFLAG .EQ. 10) GO TO 8000                                    | TST0165 |
| IF (DEPTH .GT. DEPMAX(KSTOR)) GO TO 8000                         | TST0166 |
| C                                                                | TST0167 |
| C                                                                | TST0168 |
| C                                                                | TST0169 |
| C                                                                | TST0170 |
| C                                                                | TST0171 |
| C                                                                | TST0172 |
| C                                                                | TST0173 |
| C                                                                | TST0174 |
| C                                                                | TST0175 |
| C                                                                | TST0176 |
| C                                                                | TST0177 |
| C                                                                | TST0178 |

|   |                                                         |          |
|---|---------------------------------------------------------|----------|
| C | BODOUT = BOD OUTFLOW (LB)                               | TST0179  |
| C | SSOUT = SS OUTFLOW (LB)                                 | TST0180  |
| C | BODCOT = BOD OUTFLOW CONC. (MG/L)                       | TST0181  |
| C | SSCOUT = SS OUTFLOW CONC. (MG/L)                        | TST0182  |
| C | SBODC = BOD CONC. (MG/L) AVG. IN RESERVOIR              | TST0183  |
| C | SSSC = SS CONC. (MG/L) AVG. IN RESERVOIR                | TST0184  |
| C | MG/L = 16050*(LB/CU.FT.)                                | TST0185  |
| C |                                                         | TST0186  |
|   | 3100 IF (KTSTEP .GT. 1) GO TO 3200                      | TST0187  |
|   | VOLIN(KSTOR,1) = 0.5*QINST*DT                           | TST0188  |
|   | VOLOUT(KSTOR,1) = 0.5*(QOUST+QOUSTL(KSTOR))*DT          | TST0189  |
|   | CUMIN(KSTOR) = VOLIN(KSTOR,1)                           | TST0190  |
|   | CUMOUT(KSTOR) = VOLOUT(KSTOR,1)                         | TST0191  |
|   | SBOD(KSTOR) = 0.0                                       | TST0192  |
|   | SSS(KSTOR) = 0.0                                        | TST0193  |
|   | SCOL(KSTOR) = 0.0                                       | TST0193A |
|   | LPREV(KSTOR) = 1                                        | TST0194  |
|   | JP = 0                                                  | TST0195  |
|   | LP = 0                                                  | TST0196  |
|   | GO TO 4000                                              | TST0197  |
|   | 3200 VOLIN(KSTOR,KTSTEP) = 0.5*(QINST+QINSTL(KSTOR))*DT | TST0198  |
|   | VOLOUT(KSTOR,KTSTEP) = 0.5*(QOUST+QOUSTL(KSTOR))*DT     | TST0199  |
|   | CUMIN(KSTOR) = CUMIN(KSTOR) + VOLIN(KSTOR,KTSTEP)       | TST0200  |
|   | CUMOUT(KSTOR) = CUMOUT(KSTOR) + VOLOUT(KSTOR,KTSTEP)    | TST0201  |
| C | -----                                                   | TST0202  |
|   | 4000 IF (IPOL(KSTOR) .EQ. 0) GO TO 4400                 | TST0203  |
|   | IF (VOLCUT(KSTOR,KTSTEP) .LE. 0.0) GO TO 4200           | TST0204  |
| C |                                                         | TST0205  |
|   | CALL TPLUGS                                             | TST0206  |
| C |                                                         | TST0207  |
|   | IF (LABEL .EQ. 5000) GO TO 4200                         | TST0208  |
| C |                                                         | TST0209  |
| C | ACW HAVE, FOR EACH PLUG (FOR KP=JP,LP), ..              | TST0210  |
| C | FRAC(KP),DETENT(KP)                                     | TST0211  |
| C |                                                         | TST0212  |
| C | MODEL ASSUMES..                                         | TST0213  |
| C | NO SCOUR WITHIN BASIN                                   | TST0214  |
| C |                                                         | TST0215  |
| C | BRANCH FOR TYPE OF BASIN FLOW (IPOL)                    | TST0216  |
| C |                                                         | TST0217  |
|   | K = IPOL(KSTOR)                                         | TST0218  |
|   | GO TO (4100,4180), K                                    | TST0219  |
| C |                                                         | TST0220  |
| C | IPOL=1 ASSUMES 100% EFFICIENT PLUG FLOW                 | TST0221  |
| C |                                                         | TST0222  |
|   | 4100 DO 4150 KP=JP,LP                                   | TST0223  |
|   | BODOUT = BODOUT + BODIN(KSTOR,KP)*FRAC(KP)              | TST0224  |
|   | COLOUT = COLOUT + COLIN(KSTOR,KP)*FRAC(KP)              | TST0224A |
|   | 4150 SSOUT = SSOUT + SSIN(KSTOR,KP)*FRAC(KP)            | TST0225  |
| C |                                                         | TST0226  |
|   | BODCOT = BODOUT*16050.0/VOLCUT(KSTOR,KTSTEP)            | TST0227  |
|   | SSCOUT = SSOUT *16050.0/VOLCUT(KSTOR,KTSTEP)            | TST0228  |
|   | COLCOT = COLOUT/(VOLCUT(KSTOR,KTSTEP)*283.2)            | TST0228A |
|   | GO TO 4300                                              | TST0229  |
| C |                                                         | TST0230  |
| C | IPOL=2 ASSUMES 100% MIXING                              | TST0231  |
| C |                                                         | TST0232  |
| C | FOR THIS CASE, DON'T REALLY NEED SUBROUTINE TPLUGS,     | TST0233  |
| C | OR (KTSTEP) ON VOLIN, VOLOUT, SSIN, BODIN               | TST0234  |
| C |                                                         | TST0235  |

|      |                                                                  |          |
|------|------------------------------------------------------------------|----------|
| 4180 | BODCOT = SBODC                                                   | TST0236  |
|      | SSCOUT = SSSC                                                    | TST0237  |
|      | COLCCT = SCOLC                                                   | TST0237A |
|      | BODOUT = BODCCT*VOLCUT(KSTOR,KTSTEP)/16050.0                     | TST0238  |
|      | SSOUT = SSCOUT*VOLCUT(KSTOR,KTSTEP)/16050.0                      | TST0239  |
|      | COLOUT = COLCCT*VOLCUT(KSTOR,KTSTEP)*283.2                       | TST0239A |
|      | GO TO 4300                                                       | TST0240  |
| C    |                                                                  | TST0241  |
| 4200 | BODCCT = 0.0                                                     | TST0242  |
|      | SSCOLT = 0.0                                                     | TST0243  |
|      | COLCCT = 0.0                                                     | TST0243A |
|      | IF (LPREV(KSTOR) .GT. 1) GO TO 4300                              | TST0244  |
|      | LP = 0                                                           | TST0245  |
|      | JP = 0                                                           | TST0246  |
| C    |                                                                  | TST0247  |
| 4300 | SBOD(KSTOR) = SBOD(KSTOR) - BODOUT + BODIN(KSTOR,KTSTEP)         | TST0248  |
|      | SSS(KSTOR) = SSS(KSTOR) - SSOUT + SSIN(KSTOR,KTSTEP)             | TST0249  |
|      | SCOL(KSTOR) = SCOL(KSTOR) - COLOUT + COLIN(KSTOR,KTSTEP)         | TST0249A |
|      | IF (STOR .EQ. 0.0) GO TO 4350                                    | TST0250  |
|      | SBODC = SBOD(KSTOR)*16050.0/STOR                                 | TST0251  |
|      | SSSC = SSS(KSTOR) *16050.0/STOR                                  | TST0252  |
|      | SCOLC = SCOL(KSTOR)/(STOR*283.2)                                 | TST0252A |
|      | GO TO 4400                                                       | TST0253  |
| 4350 | SBODC = 0.0                                                      | TST0254  |
|      | SSSC = 0.0                                                       | TST0255  |
|      | SCOLC = 0.0                                                      | TST0255A |
| C    |                                                                  | TST0256  |
| 4400 | IF (IPRINT(KSTOR) .LT. 1) GO TO 5300                             | TST0257  |
|      | IF (IPOL(KSTOR) .EQ. 0) GO TO 5100                               | TST0258  |
| C    |                                                                  | TST0259  |
| C    | PRINT SOLUTION FOR THIS TIME-STEP                                | TST0260  |
| C    |                                                                  | TST0261  |
| 5000 | WRITE(6,606) KSTOR,KTSTEP,TIME2M,QINST,QOUST,                    | TST0262  |
| *    | STOR,DEPTH,BODIN(KSTOR,KTSTEP),SSIN(KSTOR,KTSTEP),               | TST0263  |
| *    | SBOD(KSTOR),SSS(KSTOR),SBODC,SSSC,BODOUT,SSOUT,                  | TST0264  |
| *    | BODCOT,SSCOUT,JP,LP                                              | TST0265  |
| 606  | FORMAT(' ', I1, I3, F6.1, 2F7.1, F10.0, F6.2, 2X, 2F7.1, F10.1,  | TST0266  |
| *    | F9.1, 2F7.1, F9.1, 3F7.1, 2I4)                                   | TST0267  |
|      | WRITE(6,666) CCLIN(KSTOR,KTSTEP),SCOL(KSTOR),SCOLC,COLOUT,COLCOT | TST0267A |
| 666  | FORMAT(' ', 40X, 'COLIN= ', 1PE9.2, 'SCOL(MPN)=', E9.2,          | TST0267B |
| *    | 'CONC=', E9.2, 'COLCUT=', E9.2, 'CONC=', E9.2)                   | TST0267C |
|      | GO TO 5200                                                       | TST0268  |
| 5100 | WRITE(6,606) KSTOR,KTSTEP,TIME2M,QINST,QOUST,                    | TST0269  |
| *    | STOR,DEPTH                                                       | TST0270  |
| 5200 | IF (KSTOR .EQ. NSTOR) WRITE(6,604)                               | TST0271  |
| 5300 | IF (KSTOR .LT. NSTOR) GO TO 5400                                 | TST0272  |
|      | IF (KTSTEP .LT. NOT) GO TO 5400                                  | TST0273  |
|      | DO 5350 KSTDUM=1,NSTOR                                           | TST0274  |
|      | IF (IFLUC(KSTDUM) .EQ. 1) GO TO 5350                             | TST0275  |
|      | IF (IPRINT(KSTDUM) .LT. 1) GO TO 5350                            | TST0276  |
|      | WRITE(6,607) NCT, KSTDUM, CUMIN(KSTDUM), CUMOUT(KSTDUM)          | TST0277  |
| 607  | FORMAT('C', /, ' ', 26X, 'FOR THESE', I4, ' TIME STEPS ',        | TST0278  |
| *    | 'IN STORAGE UNIT NO.', I3, /,                                    | TST0279  |
| *    | ' ', 36X, 'CUMULATIVE INFLOW =', F12.0, ' CU.FT.', /,            | TST0280  |
| *    | ' ', 36X, 'CUMULATIVE OUTFLOW =', F12.0, ' CU.FT.')              | TST0281  |
| 5350 | CONTINUE                                                         | TST0282  |
| 5400 | GO TO 9999                                                       | TST0283  |
| C    |                                                                  | TST0284  |
| C    | ----- ERROR MESSAGES -----                                       | TST0285  |
| 8000 | WRITE(6,691)                                                     | TST0286  |

|      |                                                                     |         |
|------|---------------------------------------------------------------------|---------|
| 691  | FORMAT(' ', 14X, 'UNIT HAS FLOODED - HYDROGRAPH TERMINATED')        | TSTJ287 |
|      | IFLOOE(KSTOR) = 1                                                   | TST0288 |
|      | GO TO 9999                                                          | TST0289 |
| C    |                                                                     | TST0290 |
| 8100 | WRITE(6,692)                                                        | TST0291 |
| 692  | FORMAT(' ', 14X, 'BUFFER VOLUME BETWEEN LEVELS DSTART AND DSTOP ',  | TST0292 |
| *    | 'IS TOO SMALL,', /,                                                 | TST0293 |
| *    | ' ', 14X, 'RESULTING IN NEGATIVE VOLOUT (AND FUN).')                | TST0294 |
|      | IFLOOE(KSTOR) = 1                                                   | TST0295 |
|      | GO TO 9999                                                          | TST0296 |
| C    |                                                                     | TST0297 |
| 901  | IF (KFLAG .EQ. 10) GO TO 902                                        | TST0298 |
|      | WRITE(6,693)                                                        | TST0299 |
| 693  | FORMAT('0 *** TERMINATE - INPUT TO TINTRP PROCEDURE IS LESS THAN L  | TST0300 |
|      | *OWEST VALUE ON CURVE (IN SUSRT. STORAG)')                          | TST0301 |
|      | STOP 901                                                            | TST0302 |
| C    |                                                                     | TST0303 |
| 902  | WRITE(6,694)                                                        | TST0304 |
| 694  | FORMAT('0 *** TERMINATE - INPUT TO TINTRP PROCEDURE IS GREATER THAT | TST0305 |
|      | *N LARGEST VALUE ON CURVE (IN SUBRT. STORAG)')                      | TST0306 |
|      | STOP 902                                                            | TST0307 |
| C    |                                                                     | TST0308 |
| 9999 | RETURN                                                              | TST0309 |
|      | END                                                                 | TST0310 |
| C    | =====                                                               | TST0311 |

```

SUBROUTINE TSTCST
COMMON/TABLES/KDEPTH(25),KCLASS(25),PSIMAX(15),ALFMAX(15),
1      NN(25),MM(25),ANORM(15,51),QNORM(15,51),
2      DNORM(15,51),AFACT(15),RFACT(15)
COMMON A(160,2,2),Q(160,2,2),CPULL(160,2,2,3),QMAX(160),
1      QFULL(160),AFULL(160),DXDT(160),C1(160),SLOPE(160),
2      DIST(160),GEOM1(160),ROUGH(160),NOE(160),NUE(160,3),
3      INUE(160,3),NTYPE(160),JR(160),NKLASS,NE,NDT,EPSIL,
4      TIME,DT,M,KFULL,N,NOS,NPULL,NPRINT,ITER,
5      QCWF(160),IOLD(160),P1(160),RNOFF(160),QINFIL(160),
6      WDW(160,3),PLUTO(160,3),IR(160),P2(160),NIN(1000),
7      P5(160),P6(160),P7(160),SCF(160),BARREL(160),
8      TITLE(40),NPE(20),NYN(20),NORDER(70),GEOM2(160),
9      GECM3(160),P4(160),SCOUR(160),KSTORE(160)
COMMON BODIN( 2,150),SSIN( 2,150),BODOUT,SSOUT,COLIN(2,150),
* QINST,CCUST,QINSTL( 2),QOUSTL( 2),STORL( 2),QOUTO( 2),STORO( 2),
* NSTOR,KSTOR,IPRINT( 2),IPOL( 2),IFLOOD( 2),ICOST( 2),DEPMAX( 2),
* ATERM( 2,11),AO2DT2( 2,11),BDEPTH( 2,11),BSTOR( 2,11),COLOUT,
* DUMSTR(11),DUMDEP(11),
* KSTEP,VOLIN( 2,150),VOLOUT( 2,150),STOR,CUMIN( 2),CUMOUT( 2),
* SBOD( 2),SSS( 2),SCOL( 2),
* ISTMGD( 2),ISTTYP( 2),ISTOUT( 2),
* QPUMP( 2),DSTART( 2),DSTOP( 2),
* DTOM( 2),STORMX( 2),DTPUMP( 2),DTMORE( 2),STURF( 2),APLAN( 2),
* CLAND( 2),CSTOR( 2),CPS( 2),CTOTAL( 2),CPCUYD( 2),CPACRE( 2),
* LP,JP,LPREV( 2),LABEL,DETENT(150),FRAC(150),OUTL(10,200)
C
DIMENSION QO1(150),QO2(150)
EQU(VALANCE (QO1(1),QMAX(1)),(QO2(1),QFULL(1)))
C
DO 9000 KSTOR=1,NSTOR
IF (ICCST(KSTOR) .LT.1) GO TO 9000
C
WRITE(6,601) KSTOR
601 FORMAT('1',10X,'SUMMARY OF STORAGE AND COSTS FOR UNIT NO.',13,/)
IF (IFLCDD(KSTOR) .EQ. 1) GO TO 902
C
CLAND(KSTOR) = CPACRE(KSTOR)*APLAN(KSTOR)/43560.0
CSTOR(KSTOR) = CPCUYD(KSTOR)*STORMX(KSTOR)/27.0
CTOTAL(KSTOR) = CLAND(KSTOR) + CSTOR(KSTOR) + CPS(KSTOR)
C
C
C      BRANCH TO OUTLET TYPE (ISTOUT)
C
K = ISTCUT(KSTOR)
GO TO (8000,8000, 901, 901,4600,4600, 901, 901), K
C
C----- COMPUTE AND PRINT STORAGE UNIT COSTS -----
C
C      F) OUTLET BY EXISTING PUMPS
C
C      FOLLOWING IS TEMPORARY, DOWN TO 8000
C
4600 DTMORE(KSTOR) = STURF(KSTOR)/(QPUMP(KSTOR)*DT)
DTPUMP(KSTOR) = DTOM(KSTOR) - 1.0 + DTMORE(KSTOR)
WRITE(6,602) DTMORE(KSTOR),DTPUMP(KSTOR)
602 FORMAT('0',15X,'NO. TIMESTEPS PUMPED AFTER STORM, TO EMPTY, =',
*      F12.2,/,
*      ' ',15X,'TOTAL NO. TIMESTEPS OUTFLOW WAS PUMPED   =',
*      F12.2)
C

```

|   |                                                                     |         |
|---|---------------------------------------------------------------------|---------|
| C | Z) ALL OUTLETS                                                      | TSTC 61 |
| C |                                                                     | TSTC 62 |
|   | 8000 CONTINUE                                                       | TSTC 63 |
|   | WRITE(6,603) APLAN(KSTOR),STORMX(KSTOR),CPACRE(KSTOR),              | TSTC 64 |
|   | * CLAND(KSTOR),CPCUYD(KSTOR),CSTOR(KSTOR),CPS(KSTOR),               | TSTC 65 |
|   | * CTOTAL(KSTOR)                                                     | TSTC 66 |
|   | 603 FORMAT(' ', 15X, 'PLAN AREA OF RESERVOIR (LAND REQUIREMENT) =', | TSTC 67 |
|   | * F12.2, ' SQ.FT.', /,                                              | TSTC 68 |
|   | * ' ', 15X, 'MAX. STORAGE VOLUME USED (CONSTRUC. REQ.) =',          | TSTC 69 |
|   | * F12.2, ' CU.FT.', /,                                              | TSTC 70 |
|   | * '0'. 15X, 'COST OF LAND, AT \$',F7.0,'/ACRE . . . . . = \$',      | TSTC 71 |
|   | * F12.2, /,                                                         | TSTC 72 |
|   | * ' ', 15X, 'COST OF RESERVOIR, AT \$',F5.2,                        | TSTC 73 |
|   | * ' /CU.YD. . . . = \$',F12.2, /,                                   | TSTC 74 |
|   | * ' ', 15X, 'COST OF PUMP STN. & SPECIAL STRUCTURES . . = \$',      | TSTC 75 |
|   | * F12.2, /,                                                         | TSTC 76 |
|   | * 'C', 15X, 'TOTAL COST', 32X, '= \$', F12.2)                       | TSTC 77 |
| C |                                                                     | TSTC 78 |
|   | 9000 CONTINUE                                                       | TSTC 79 |
|   | GO TO 9999                                                          | TSTC 80 |
| C |                                                                     | TSTC 81 |
| C | ----- ERROR MESSAGES -----                                          | TSTC 82 |
| C |                                                                     | TSTC 83 |
| C |                                                                     | TSTC 84 |
|   | 901 WRITE(6,691) ISTCUT(KSTOR), KSTOR                               | TSTC 85 |
|   | 691 FORMAT('0 *** ISTOUT =', I3, ' ', IN RESERVOIR NO.', I4,        | TSTC 86 |
|   | * ' ', IS OF A TYPE NOT PRESENTLY MODELED')                         | TSTC 87 |
|   | STOP                                                                | TSTC 88 |
| C |                                                                     | TSTC 89 |
|   | 902 WRITE(6,692)                                                    | TSTC 90 |
|   | 692 FORMAT('C', 10X, '*** UNIT FLOODED, SO SUMMARY IS INVALID')     | TSTC 91 |
| C |                                                                     | TSTC 92 |
|   | 9999 RETURN                                                         | TSTC 93 |
|   | END                                                                 | TSTC 94 |
| C | =====                                                               | TSTC 95 |

|                                                                    |                                                           |          |
|--------------------------------------------------------------------|-----------------------------------------------------------|----------|
| SUBROUTINE TPLUGS                                                  |                                                           | TPLU 1   |
| COMMON/TABLES/KDEPTH(25),KLASS(25),PS1MAX(15),ALFMAX(15),          |                                                           | TPLU 2   |
| 1                                                                  | NN(25),MM(25),ANORM(15,51),QNORM(15,51),                  | TPLU 3   |
| 2                                                                  | DNORM(15,51),AFAC(15),RFAC(15)                            | TPLU 4   |
| COMMON A(160,2,2),Q(160,2,2),CPOLL(160,2,2,3),QMAX(160),           |                                                           | TPLU 5X  |
| 1                                                                  | QFULL(160),AFULL(160),DXDT(160),C1(160),SLOPE(160),       | TPLU 6   |
| 2                                                                  | DIST(160),GEOM1(160),ROUGH(160),NUE(160),NUE(160,3),      | TPLU 7   |
| 3                                                                  | INUE(160,3),NTYPE(160),JR(160),NKLASS,NE,NDT,EPSIL,       | TPLU 8   |
| 4                                                                  | TIME,DT,M,KFULL,N,NDS,NPOLL,NPRINT,ITER,                  | TPLU 9   |
| 5                                                                  | QCHF(160),IOLD(160),P1(160),RNOFF(160),QINFIL(160),       | TPLU 10  |
| 6                                                                  | WCHF(160,3),PLUTO(160,3),IR(160),P2(160),NIN(1000),       | TPLU 11X |
| 7                                                                  | P5(160),P6(160),P7(160),SCF(160),BARREL(160),             | TPLU 12  |
| 8                                                                  | TITLE(40),NPE(20),NYN(20),NORDER(70),GEOM2(160),          | TPLU 13X |
| 9                                                                  | GEOM3(160),P4(160),SCOUR(160),KSTORE(160)                 | TPLU 14  |
| COMMON BODIN( 2,150),SSIN( 2,150),BODOUT,SSOUT,COLIN(2,150),       |                                                           | TPLU 15X |
| * QINST,COLST,QINSTL( 2),QOUSTL( 2),STORL( 2),QOUTO( 2),STORO( 2), |                                                           | TPLU 16  |
| * NSTOR,KSTOR,IPRINT( 2),IPOL( 2),IFLOOD( 2),ICOST( 2),DEPMAX( 2), |                                                           | TPLU 17  |
| * ATERM( 2,11),AOZDT2( 2,11),BDEPTH( 2,11),BSTOR( 2,11),COLOUT,    |                                                           | TPLU 18X |
| * DUMSTR(11),DUMDEP(11),                                           |                                                           | TPLU 19  |
| * KTSTEP,VOLIN( 2,150),VOLOUT( 2,150),STOR,CUMIN( 2),CUMOUT( 2),   |                                                           | TPLU 20  |
| * SBOD( 2),SSS( 2),SCCL( 2),                                       |                                                           | TPLU 21X |
| * ISTMOD( 2),ISTTYP( 2),ISTOUT( 2),                                |                                                           | TPLU 22  |
| * QPUMP( 2),DSTART( 2),DSTOP( 2),                                  |                                                           | TPLU 23  |
| * OTON( 2),STORMX( 2),DTPUMP( 2),DTMORE( 2),STORF( 2),APLAN( 2),   |                                                           | TPLU 24  |
| * CLAND( 2),CSTOR( 2),CPS( 2),CTOTAL( 2),CPCUYD( 2),CPACRE( 2),    |                                                           | TPLU 25  |
| * LP,JP,LPREV( 2),LABEL,DETENT(150),FRAC(150),OUTI(10,200)         |                                                           | TPLU 26X |
| C                                                                  |                                                           | TPLU 27X |
| DIMENSION QO1(150),QO2(150)                                        |                                                           | TPLU 28  |
| EQUIVALENCE (QO1(1),QMAX(1)),(QO2(1),QFULL(1))                     |                                                           | TPLU 29  |
| C                                                                  |                                                           | TPLU 30  |
| C                                                                  | GIVEN KTSTEP, VOLIN, VOLOUT & STORE ARRAYS, KTSTEP & TPLU | TPLU 31  |
| C                                                                  | FINDS LP AND JP (NUMBERS OF FIRST AND LAST PART-PLUG TPLU | TPLU 32  |
| C                                                                  | COMPUTES DETENT AND FRAC ARRAYS (KP=JP,LP)                | TPLU 33  |
| C                                                                  | (SUBPROGRAM BY G.T.)                                      | TPLU 34  |
| C                                                                  |                                                           | TPLU 35  |
| LABEL = 0                                                          |                                                           | TPLU 36  |
| LP = KTSTEP                                                        |                                                           | TPLU 37  |
| JP = LPREV(KSTOR)                                                  |                                                           | TPLU 38  |
| VIKK = VOLIN(KSTOR,KTSTEP)                                         |                                                           | TPLU 39  |
| VOKK = VOLOUT(KSTOR,KTSTEP)                                        |                                                           | TPLU 40  |
| SUM1=C.0                                                           |                                                           | TPLU 40A |
| SUM = 0.0                                                          |                                                           | TPLU 41  |
| 1000                                                               | SUM = SUM + VOLIN(KSTOR,LP)                               | TPLU 42  |
| SUM1= SUM + VOKK - VIKK                                            |                                                           | TPLU 42A |
| BACK = SUM1 - STOR                                                 |                                                           | TPLU 43  |
| IF (BACK .GT. VCKK) GO TO 1100                                     |                                                           | TPLU 44  |
| GO TO 1200                                                         |                                                           | TPLU 45  |
| 1100                                                               | BACK = VCKK                                               | TPLU 46  |
| 1200                                                               | IF(BACK) 1300,1400,2000                                   | TPLU 47  |
| 1300                                                               | LP = LP - 1                                               | TPLU 48  |
| IF(LP-1) 5000,1000,1000                                            |                                                           | TPLU 49  |
| C                                                                  |                                                           | TPLU 50  |
| 1400                                                               | WRITE(6,601)                                              | TPLU 51  |
| 601 FORMAT('G A PART-PLUG = 0. EXECUTION TERMINATED.')             |                                                           | TPLU 52  |
| STOP 1400                                                          |                                                           | TPLU 53  |
| C                                                                  |                                                           | TPLU 54  |
| 2000                                                               | IF(VIKK .LE. 0.0) GO TO 4000                              | TPLU 55  |
| IF(LP .EQ. KTSTEP) GO TO 3000                                      |                                                           | TPLU 56  |
| C                                                                  |                                                           | TPLU 57  |
| C                                                                  | GENERAL CASE                                              | TPLU 58  |

|      |                                 |                                        |         |
|------|---------------------------------|----------------------------------------|---------|
| C    |                                 | A NUMBER OF PLUGS AND PART-PLUGS LEAVE | TPLU 59 |
| C    |                                 |                                        | TPLU 60 |
|      | IF(LP .EQ. 1) GO TO 2500        |                                        | TPLU 61 |
|      | IF(JP .GT. 1) GO TO 2800        |                                        | TPLU 62 |
|      | NF = JP + 1                     |                                        | TPLU 63 |
|      | IF(NF .EQ. LP) GO TO 2600       |                                        | TPLU 64 |
|      | SOUT = 0.0                      |                                        | TPLU 65 |
|      | ML = LP - 1                     |                                        | TPLU 66 |
|      | NF = JP + 1                     |                                        | TPLU 67 |
|      | DO 2050 L=NF,ML                 |                                        | TPLJ 68 |
| 2050 | SOUT = SOUT + VOLIN(KSTOR,L)    |                                        | TPLU 69 |
|      | SOUT = SOUT + BACK              |                                        | TPLU 70 |
|      | STOT = SOUT + VOLIN(KSTOR,JP)   |                                        | TPLU 71 |
|      | IF(STOT .GE. VOKK) GO TO 2400   |                                        | TPLJ 72 |
|      | FRAC(JP)= 1.0                   |                                        | TPLU 73 |
| 2100 | RIJP = KTSTEP - JP              |                                        | TPLJ 74 |
|      | DETENT(JP) = RIJP*DT            |                                        | TPLU 75 |
|      | DO 2200 L=NF,ML                 |                                        | TPLJ 76 |
|      | FRAC(L)= 1.0                    |                                        | TPLJ 77 |
|      | RIL = KTSTEP - L                |                                        | TPLU 78 |
| 2200 | DETENT(L) = RIL*DT              |                                        | TPLJ 79 |
| 2300 | FRAC(LP)= BACK/VOLIN(KSTOR,LP)  |                                        | TPLU 80 |
|      | RILP = KTSTEP - LP              |                                        | TPLJ 81 |
|      | DETENT(LP) = RILP*DT            |                                        | TPLJ 82 |
|      | LPREV(KSTOR) = LP               |                                        | TPLU 83 |
|      | GO TO 5555                      |                                        | TPLU 84 |
| C    |                                 |                                        | TPLU 85 |
| 2400 | FRONT = VOKK - SOUT             |                                        | TPLU 86 |
|      | FRAC(JP)= FRONT/VOLIN(KSTOR,JP) |                                        | TPLJ 87 |
|      | GO TO 2100                      |                                        | TPLU 88 |
| C    |                                 |                                        | TPLJ 89 |
| 2500 | FRONT = VOKK - BACK             |                                        | TPLU 90 |
|      | GO TO 2300                      |                                        | TPLU 91 |
| C    |                                 |                                        | TPLJ 92 |
| 2600 | SOUT = BACK + VOLIN(KSTOR,JP)   |                                        | TPLU 93 |
|      | IF(SOUT .GE. VOKK) GO TO 2700   |                                        | TPLU 94 |
|      | FRAC(JP)= 1.0                   |                                        | TPLU 95 |
|      | RIJP = KTSTEP - JP              |                                        | TPLU 96 |
|      | DETENT(JP) = RIJP*DT            |                                        | TPLJ 97 |
|      | GO TO 2300                      |                                        | TPLU 98 |
| C    |                                 |                                        | TPLU 99 |
| 2700 | FRONT = VOKK - BACK             |                                        | TPLJ100 |
|      | FRAC(JP)= FRONT/VOLIN(KSTOR,JP) |                                        | TPLJ101 |
|      | RIJP = KTSTEP - JP              |                                        | TPLJ102 |
|      | DETENT(JP) = RIJP*DT            |                                        | TPLJ103 |
|      | GO TO 2300                      |                                        | TPLU104 |
| C    |                                 |                                        | TPLU105 |
| 2800 | IF (LP .EQ. JP) GO TO 4100      |                                        | TPLU106 |
|      | NF = JP + 1                     |                                        | TPLU107 |
|      | IF(NF .EQ. LP) GO TO 2700       |                                        | TPLU108 |
|      | SOUT = 0.0                      |                                        | TPLU109 |
|      | ML = LP - 1                     |                                        | TPLU110 |
|      | DO 2850 L=NF,ML                 |                                        | TPLJ111 |
| 2850 | SOUT = SOUT + VOLIN(KSTOR,L)    |                                        | TPLU112 |
|      | SOUT = SOUT + BACK              |                                        | TPLJ113 |
|      | GO TO 2400                      |                                        | TPLU114 |
| C    |                                 |                                        | TPLU115 |
| C    |                                 | SPECIAL CASE                           | TPLU116 |
| C    |                                 | INFLOW PLUG LARGER THAN BASIN VOLUME   | TPLU117 |
| C    |                                 |                                        | TPLJ118 |



|      |                                 |         |
|------|---------------------------------|---------|
| 3000 | IF(LP .EQ. 1) GO TO 3700        | TPLU119 |
|      | IF(JP .GT. 1) GO TO 3800        | TPLU120 |
|      | NF = JP + 1                     | TPLU121 |
|      | IF(NF .EQ. LP) GO TO 3500       | TPLU122 |
|      | SOUT = 0.0                      | TPLU123 |
|      | ML = LP - 1                     | TPLU124 |
|      | NF = JP + 1                     | TPLU125 |
|      | DO 3050 L=NF,ML                 | TPLU126 |
| 3050 | SOUT = SOUT + VOLIN(KSTOR,L)    | TPLU127 |
|      | SOUT = SOUT + BACK              | TPLU128 |
|      | STOT = SCUT + VOLIN(KSTOR,JP)   | TPLU129 |
|      | IF(STOT .GE. VOKK) GO TO 3400   | TPLU130 |
|      | FRAC(JP)= 1.0                   | TPLU131 |
| 3100 | RIJP = KTSTEP - JP              | TPLU132 |
|      | DETENT(JP) = RIJP*DT            | TPLU133 |
|      | DO 3200 L=NF,ML                 | TPLU134 |
|      | FRAC(L)= 1.0                    | TPLU135 |
|      | RIL = KTSTEP - L                | TPLU136 |
| 3200 | DETENT(L) = RIL*DT              | TPLU137 |
| 3300 | FRAC(LP)= BACK/VOLIN(KSTOR,LP)  | TPLU138 |
|      | DETENT(LP) = DT*STOR/VIKK       | TPLU139 |
|      | LPREV(KSTOR) = LP               | TPLU140 |
|      | GO TO 5999                      | TPLU141 |
| C    |                                 | TPLU142 |
| 3400 | FRONT = VOKK - SOUT             | TPLU143 |
|      | FRAC(JP)= FRONT/VOLIN(KSTOR,JP) | TPLU144 |
|      | GO TO 3100                      | TPLU145 |
| C    |                                 | TPLU146 |
| 3500 | SOUT = BACK + VOLIN(KSTOR,JP)   | TPLU147 |
|      | RIJP = KTSTEP - JP              | TPLU148 |
|      | DETENT(JP) = RIJP*DT            | TPLU149 |
|      | IF(SOUT .GE. VOKK) GO TO 3600   | TPLU150 |
|      | FRAC(JP)= 1.0                   | TPLU151 |
|      | GO TO 3300                      | TPLU152 |
| C    |                                 | TPLU153 |
| 3600 | FRONT = VOKK - BACK             | TPLU154 |
|      | FRAC(JP)= FRONT/VOLIN(KSTOR,JP) | TPLU155 |
|      | GO TO 3300                      | TPLU156 |
| C    |                                 | TPLU157 |
| 3700 | FRONT = VOKK - BACK             | TPLU158 |
|      | FRAC(LP)= BACK/VIKK             | TPLU159 |
|      | DETENT(LP) = DT*STOR/VIKK       | TPLU160 |
|      | LPREV(KSTOR) = LP               | TPLU161 |
|      | GO TO 9999                      | TPLU162 |
| C    |                                 | TPLU163 |
| 3800 | IF (LP .EQ. JP) GO TO 4100      | TPLU164 |
|      | NF = JP + 1                     | TPLU165 |
|      | IF(NF .EQ. LP) GO TO 3600       | TPLU166 |
|      | SOUT = 0.0                      | TPLU167 |
|      | ML = LP - 1                     | TPLU168 |
|      | DO 3850 L=NF,ML                 | TPLU169 |
| 3850 | SOUT = SOUT + VOLIN(KSTOR,L)    | TPLU170 |
|      | SOUT = SOUT + BACK              | TPLU171 |
|      | FRONT = VOKK - SOUT             | TPLU172 |
|      | FRAC(JP)= FRONT/VOLIN(KSTOR,JP) | TPLU173 |
|      | GO TO 3100                      | TPLU174 |
| C    |                                 | TPLU175 |
| C    |                                 | TPLU176 |
| C    | SPECIAL END CONDITION           | TPLU177 |
| C    | NO INFLOW, OUTFLOW FROM STORAGE | TPLU178 |

|                                     |         |
|-------------------------------------|---------|
| 4000 IF (LP .EQ. JP) GO TO 4100     | TPLU179 |
| NF = JP + 1                         | TPLU180 |
| IF(NF .EQ. LP) GO TO 4200           | TPLU181 |
| SOUT = 0.0                          | TPLU182 |
| ML = LP - 1                         | TPLU183 |
| DO 4050 L=NF,ML                     | TPLU184 |
| 4050 SOUT = SOUT + VOLIN(KSTOR,L)   | TPLU185 |
| SOUT = SOUT + SUM - STOR            | TPLU186 |
| GO TO 2400                          | TPLU187 |
| C                                   | TPLU188 |
| 4100 FRAC(LP)= BACK/VOLIN(KSTOR,LP) | TPLU189 |
| RILP = KTSTEP - LP                  | TPLU190 |
| DETENT(LP) = RILP*DT                | TPLU191 |
| LPREV(KSTOR) = LP                   | TPLU192 |
| GO TO 9999                          | TPLU193 |
| C                                   | TPLU194 |
| 4200 FRONT = VOKK - BACK            | TPLU195 |
| FRAC(JP)= FRONT/VOLIN(KSTOR,JP)     | TPLU196 |
| RILP = KTSTEP - LP                  | TPLU197 |
| DETENT(JP) = RILP*DT                | TPLU198 |
| GO TO 2300                          | TPLU199 |
| C                                   | TPLU200 |
| 5000 LABEL = 5000                   | TPLU201 |
| GO TO 9999                          | TPLU202 |
| C                                   | TPLU203 |
| 9999 RETURN                         | TPLU204 |
| END                                 | TPLU205 |
| C=====                              | TPLU206 |

```

SUBROUTINE TSROUT
COMMON/TABLES/KDEPTH(25),KLASS(25),PSIMAX(15),ALFMAX(15),
1      NN(25),MM(25),ANORM(15,51),QNORM(15,51),
2      DNORM(15,51),AFAC(15),RFAC(15)
COMMON A(160,2,2),Q(160,2,2),CPOLL(160,2,2,3),QMAX(160),
1      QFULL(160),AFULL(160),DXDT(160),C1(160),SLOPE(160),
2      DIST(160),GEOM1(160),ROUGH(160),NOE(160),NUE(160,3),
3      INUE(160,3),NTYPE(160),JR(160),NKLASS,NE,NDT,EPSIL,
4      TIME,DT,M,KFULL,N,NUS,NPULL,NPRINT,ITER,
5      QCWF(160),IOLD(160),P1(160),RNOFF(160),QINFIL(160),
6      WCWF(160,3),PLUTO(160,3),IR(160),P2(160),NIN(1000),
7      P5(160),P6(160),P7(160),SCF(160),BARREL(160),
8      TITLE(40),NPE(20),NYN(20),NORDER(70),GECM2(160),
9      GECM3(160),P4(160),SCUUR(160),KSTORE(160)
COMMON BOCIN(2,150),SSIN(2,150),BODOUT,SSOUT,COLIN(2,150),
* QINST,COLST,QINSTL(2),QOUSTL(2),STORL(2),QOUTO(2),STORO(2),
* NSTOR,KSTOR,IPRINT(2),IPOL(2),IFLOOD(2),ICOST(2),DEPMAX(2),
* ATERM(2,11),AO2DT2(2,11),BDEPTH(2,11),BSTOR(2,11),COLOUT,
* DUMSTR(11),DUMDEP(11),
* KTSTEP,VOLIN(2,150),VOLOUT(2,150),STOR,CUMIN(2),CUMOUT(2),
* SBOD(2),SSS(2),SCCL(2),
* ISTMOC(2),ISTTYP(2),ISTOUT(2),
* QPUMP(2),DSTART(2),DSTOP(2),
* DTUN(2),STORMX(2),DTPLMP(2),DTMORE(2),STURF(2),APLAN(2),
* CLANC(2),CSTOR(2),CPS(2),CTOTAL(2),CPCUYD(2),CPACRE(2),
* LP,JP,LPREV(2),LABEL,DETENT(150),FRAC(150),OUT1(10,200)
DIMENSION DUMTRM(11),DUMAO2(11)
DIMENSION COL(150),QO2(150)
EQUIVALENCE (COL(1),QMAX(1)),(QO2(1),QFULL(1))
ONLY WORKS FOR ONE UNIT (KSTOR) AT A TIME
INPUTS ARE DT, QINST, KSTOR, KTSTEP, STORO(2),
          QOUTO(2), ATERM(2,11), AO2DT2(2,11)
OUTPUTS ARE QOUST, STORO(100)
NOTE.. ALL FLOWS ARE IN CFS.
IF (KTSTEP .GT. 1) GO TO 1000
INITIALISE
QINSTL(KSTOR) = 0.0
QOUSTL(KSTOR) = QOUTO(KSTOR)
STORL(KSTOR) = STORO(KSTOR)
DT2 = DT/2.0
1000 TERM = (QINSTL(KSTOR) + QINST)*DT2 - (QOUSTL(KSTOR)*DT2
*          - STORL(KSTOR))
IF (TERM .GE. 0.0) GO TO 2000
WRITE(6,601) TERM, KTSTEP
601 FORMAT(' ', '** RESET TERM = ', F12.5, ' TO ZERO ', I10)
TERM = 0.0
FIND 'O2CT2' CORRESPONDING TO 'TERM' ABOVE,
BY LINEAR INTRPOLATION.
2000 DO 2050 II=1,11
DUMTRM(II) = ATERM(KSTOR,II)
2050 DUMAO2(II) = AO2DT2(KSTOR,II)

```

|        |                                                                    |         |
|--------|--------------------------------------------------------------------|---------|
| C      | CALL TINTRP(DUMTRM,DUMAO2,11,TERM,O2DT2,KFLAG)                     | TSR0 61 |
| C      | IF (KFLAG.NE.0) GO TO 901                                          | TSR0 62 |
|        | STOR = TERM - O2DT2                                                | TSR0 63 |
|        | QOUST = O2DT2/DT2                                                  | TSR0 64 |
| CC     | WRITE(6,61)                                                        | TSR0 65 |
|        | IF (ISTCUT(KSTOR).NE.9) GO TO 2200                                 | TSR0 66 |
| C      |                                                                    | TSR0 67 |
| C      | COMPUTE OUTFLOW THROUGH ORIFICE.                                   | TSR0 68 |
| C      |                                                                    | TSR0 69 |
|        | DO 2100 II=1,11                                                    | TSR0 70 |
| 2100   | DUMAO2(II) = OUT1(KSTCR,II)                                        | TSR0 71 |
|        | CALL TINTRP(DUMTRM,DUMAO2,11,TERM,O2DT2,KFLAG)                     | TSR0 72 |
|        | IF (KFLAG.NE.0) GO TO 901                                          | TSR0 73 |
|        | QO1(M) = O2DT2/DT2                                                 | TSR0 74 |
|        | QO2(M) = QOUST-QO1(M)                                              | TSR0 75 |
| C      |                                                                    | TSR0 76 |
| C      | INITIALISE FOR NEXT TIME-STEP                                      | TSR0 77 |
| C      |                                                                    | TSR0 78 |
| 2200   | QINSTL(KSTCR) = QINST                                              | TSR0 79 |
|        | QOUSTL(KSTCR) = QOUST                                              | TSR0 80 |
|        | STORL(KSTCR) = STOR                                                | TSR0 81 |
|        | GO TO 9999                                                         | TSR0 82 |
| C      |                                                                    | TSR0 83 |
| C----- | ERROR MESSAGES                                                     | TSR0 84 |
| C      |                                                                    | TSR0 85 |
| 901    | IF (KFLAG.EQ.10) GO TO 902                                         | TSR0 86 |
|        | WRITE(6,6S1)                                                       | TSR0 87 |
| 691    | FORMAT('O *** TERMINATE - INPUT TO TINTRP PROCEDURE IS LESS THAN L | TSR0 88 |
|        | *QWEST VALUE ON CURVE (IN SUBRT. ROUTE)')                          | TSR0 89 |
|        | STOP                                                               | TSR0 90 |
| C      |                                                                    | TSR0 91 |
| 902    | WRITE(6,6S2)                                                       | TSR0 92 |
| 692    | FORMAT('C *** TERMINATE - INPUT TO TINTRP PROCEDURE IS GREATER THA | TSR0 93 |
|        | *N LARGEST VALUE ON CURVE (IN SUBRT. ROUTE)')                      | TSR0 94 |
|        | STOP                                                               | TSR0 95 |
| C      |                                                                    | TSR0 96 |
| 9999   | RETURN                                                             | TSR0 97 |
|        | END                                                                | TSR0 98 |
| C===== |                                                                    | TSR0 99 |
|        |                                                                    | TSR0100 |
|        |                                                                    | TSR0101 |

|                                                            |         |
|------------------------------------------------------------|---------|
| SUBROUTINE TINTRP(X, Y, L, XE, YE, K)                      | TINT 1  |
| DIMENSION X(L),Y(L)                                        | TINT 2  |
| C                                                          | TINT 3  |
| C                                                          | TINT 4  |
| C                                                          | TINT 5  |
| C                                                          | TINT 6  |
| C                                                          | TINT 7  |
| IF (XE .GE. X(1) ) GO TO 2000                              | TINT 8  |
| C                                                          | TINT 9  |
| C                                                          | TINT 10 |
| 1000 K = -10                                               | TINT 11 |
| GO TO 9999                                                 | TINT 12 |
| C                                                          | TINT 13 |
| C                                                          | TINT 14 |
| 2000 DO 2050 J=2,L                                         | TINT 15 |
| IF (XE-X(J) ) 4000,3000,2050                               | TINT 16 |
| 2050 CONTINUE                                              | TINT 17 |
| C                                                          | TINT 18 |
| C                                                          | TINT 19 |
| K = 10                                                     | TINT 20 |
| GO TO 9999                                                 | TINT 21 |
| C                                                          | TINT 22 |
| C                                                          | TINT 23 |
| 3000 YE = Y(J)                                             | TINT 24 |
| GO TO 5000                                                 | TINT 25 |
| C                                                          | TINT 26 |
| C                                                          | TINT 27 |
| C                                                          | TINT 28 |
| C                                                          | TINT 29 |
| 4000 YE = Y(J-1) + (Y(J)-Y(J-1))*(XE-X(J-1))/(X(J)-X(J-1)) | TINT 30 |
| 5000 K = 0                                                 | TINT 31 |
| C                                                          | TINT 32 |
| 9999 RETURN                                                | TINT 33 |
| END                                                        | TINT 34 |

```
FUNCTION ACOS(X)
  ARG = (1.0 - X) / (1.0 + X)
  TARG = SCRT(ARG)
  ACOS = 2.0 * ATAN(TARG)
  RETURN
END
```

```
ACOS 1
ACOS 2
ACOS 3
ACOS 4
ACOS 5
ACOS 6
```

| BLOCK DATA |                                                           | DATA |    |
|------------|-----------------------------------------------------------|------|----|
| C          | TYPE=01 CIRCULAR CONDUIT                                  | DATA | 1  |
| C          | TYPE=02 RECTANGULAR CONDUIT                               | DATA | 2  |
| C          | TYPE=03 EGG-SHAPED CONDUIT                                | DATA | 3  |
| C          | TYPE=04 HORSESHOE CONDUIT                                 | DATA | 4  |
| C          | TYPE=05 GOTHIC SHAPED CONDUIT                             | DATA | 5  |
| C          | TYPE=06 CATENARY CONDUIT                                  | DATA | 6  |
| C          | TYPE=07 SEMI ELLIPTICAL CONDUIT                           | DATA | 7  |
| C          | TYPE=08 BASKET-HANDLE CONDUIT                             | DATA | 8  |
| C          | TYPE=09 SEMI-CIRCULAR CONDUIT                             | DATA | 9  |
| C          | TYPE=10 MODIFIED BASKET-HANDLE CONDUIT                    | DATA | 10 |
| C          | TYPE=11 RECTANGULAR CONDUIT (TRIANGULAR BOTTOM)           | DATA | 11 |
| C          | TYPE=12 RECTANGULAR CONDUIT (ROUND BOTTOM)                | DATA | 12 |
| C          | TYPE=13 USER SUPPLIED                                     | DATA | 13 |
| C          | TYPE=14 USER SUPPLIED                                     | DATA | 14 |
| C          | TYPE=15 USER SUPPLIED                                     | DATA | 15 |
| C          | TYPE=16 MANHOLE                                           | DATA | 16 |
| C          | TYPE=17 LIFT STATION                                      | DATA | 17 |
| C          | TYPE=18 FLOW DIVIDER                                      | DATA | 18 |
| C          | TYPE=19 STORAGE UNIT                                      | DATA | 19 |
| C          | TYPE=20 FLOW DIVIDER                                      | DATA | 20 |
| C          | TYPE=21 FLOW DIVIDER                                      | DATA | 21 |
| C          | TYPE=22 BACKWATER ELEMENT                                 | DATA | 22 |
| C          | TYPE=23                                                   | DATA | 23 |
| C          | TYPE=24                                                   | DATA | 24 |
| C          | TYPE=25                                                   | DATA | 25 |
|            | COMMON/NAMES/ NAME(4,20),NAME2(4,5),GNO,YES,BLANK         | DATA | 26 |
|            | COMMON/TABLES/KDEPTH(25),KLASS(25),PSIMAX(15),ALFMAX(15), | DATA | 27 |
| 1          | NN(25),NM(25),ANORM(15,51),QN1(120),QN2(120),             | DATA | 28 |
| 2          | QN3(120),QN4(120),QN5(120),QN6(120),QN7(45),              | DATA | 29 |
| 3          | DN1(120),DN2(120),DN3(120),DN4(120),DN5(120),             | DATA | 30 |
| 4          | DN6(120),DN7(45),AFAC(15),RFAC(15)                        | DATA | 31 |
|            | CATA NAME/4HCIRC,4HULAR,4H SHA,4HPED ,                    | DATA | 32 |
| *          | 4HRECT,4HANGU,4HLAR ,4H ,                                 | DATA | 33 |
| *          | 4HEGG-,4HSHAP,4HED ,4H ,                                  | DATA | 34 |
| *          | 4HFCRS,4HESH,4HOE ,4H ,                                   | DATA | 35 |
| *          | 4HGOth,4HIC S,4HHAPE,4HOD ,                               | DATA | 36 |
| *          | 4HCATE,4HNARY,4H SHA,4HPED ,                              | DATA | 37 |
| *          | 4HSEMI,4H ELL,4HIPTI,4HCAL ,                              | DATA | 38 |
| *          | 4HBASK,4HET H,4HANDL,4HE ,                                | DATA | 39 |
| *          | 4HSEMI,4H CIR,4HCULA,4HR ,                                | DATA | 40 |
| *          | 4HMODI,4HFIED,4H B. ,4HH. ,                               | DATA | 41 |
| *          | 4HRECT,4H. - ,4HTRIA,4HNG. ,                              | DATA | 42 |
| *          | 4HRECT,4H. - ,4HROUN,4HOD ,                               | DATA | 43 |
| *          | 4H USE,4HR SU,4HPPLI,4HED ,                               | DATA | 44 |
| *          | 4H USE,4HR SU,4HPPLI,4HED ,                               | DATA | 45 |
| *          | 4H USE,4HR SU,4HPPLI,4HED ,                               | DATA | 46 |
| *          | 4HMANH,4HOLE ,4H ,4H ,                                    | DATA | 47 |
| *          | 4HLIFT,4H STA,4HTION,4H ,                                 | DATA | 48 |
| *          | 4HFLOW,4H DIV,4HIDER,4H ,                                 | DATA | 49 |
| *          | 4HSTOR,4HAGE ,4HUNIT,4H ,                                 | DATA | 50 |
| *          | 4HFLOW,4H DIV,4HIDER,4H /                                 | DATA | 51 |
|            | CATA NAME2/4HFLOW,4H DIV,4HIDER,4H ,                      | DATA | 52 |
| *          | 4HBACK,4HWATE,4HR UN,4HIT ,                               | DATA | 53 |
| *          | 4H ,4H ,4H ,4H ,                                          | DATA | 54 |
| *          | 4H ,4H ,4H ,4H ,                                          | DATA | 55 |
| *          | 4H ,4H ,4H ,4H /                                          | DATA | 56 |
|            | CATA GNO/4HNO /, YES/4HYES /, BLANK/4H                    | DATA | 57 |
|            | DATA KDEPTH/2,1,7*2,3*1,3*2,10*3/                         | DATA | 58 |
|            | DATA KLASS/2,1,7*2,3*1,3*2,10*3/                          | DATA | 59 |
|            |                                                           | DATA | 60 |

|                                                                    |         |
|--------------------------------------------------------------------|---------|
| DATA AN/51,1,7*51,6*1,10*0/                                        | DATA 61 |
| DATA MM/51,1,7*51,6*1,10*0/                                        | DATA 62 |
| DATA ALFMAX/.96,.97,.96,.96,.98,.98,.96,.96,3*.98,3*.96/           | DATA 63 |
| DATA PSIMAX/1.08,.9505,1.065,1.077,1.065,1.05,1.045,1.06078,       | DATA 64 |
| 1 1.06637,6*1.0/                                                   | DATA 65 |
| DATA AFACT/0.7853982,0.0,0.5105,0.8293,0.6554,0.70277,0.785,0.7862 | DATA 66 |
| 1,1.2697,6*C.0/,RFAC1/0.25,0.0,0.1931,0.2538,0.2269,0.23172,0.242, | DATA 67 |
| 2 0.2464,0.2946,6*0.0/                                             | DATA 68 |
| DATA ANCRM/15*0.0,15*.02,15*.04,15*.06,15*.08,15*.10,15*.12,       | DATA 69 |
| * 15*.14,15*.16,15*.18,15*.20,15*.22,15*.24,15*.26,15*.28,         | DATA 70 |
| * 15*.30,15*.32,15*.34,15*.36,15*.38,15*.40,15*.42,                | DATA 71 |
| * 15*.44,15*.46,15*.48,15*.50,15*.52,15*.54,15*.56,                | DATA 72 |
| * 15*.58,15*.60,15*.62,15*.64,15*.66,15*.68,15*.70,                | DATA 73 |
| * 15*.72,15*.74,15*.76,15*.78,15*.80,15*.82,15*.84,                | DATA 74 |
| * 15*.86,15*.88,15*.90,15*.92,15*.94,15*.96,15*.98,                | DATA 75 |
| * 15*1.0/                                                          | DATA 76 |
| DATA CN1/0.0 ,0.0 ,0.0 ,0.0 ,0.0 ,0.0 ,0.0 ,                       | DATA 77 |
| * 0.0 ,0.0 ,0.0 ,0.0 ,4*0.0,                                       | DATA 78 |
| * 0.00526,0.0 ,0.00295,0.00467,0.00500,0.00605,0.00438,            | DATA 79 |
| * 0.00758,0.00757,0.0 ,0.0 ,4*0.0,                                 | DATA 80 |
| * 0.01414,0.0 ,0.01331,0.01237,0.01740,0.01455,0.01227,            | DATA 81 |
| * 0.01812,0.01815,0.0 ,0.0 ,4*0.0,                                 | DATA 82 |
| * 0.02553,0.0 ,0.02629,0.02268,0.03098,0.02540,0.02312,            | DATA 83 |
| * 0.03000,0.03000,0.0 ,0.0 ,4*0.0,                                 | DATA 84 |
| * 0.03862,0.0 ,0.04000,0.03515,0.04272,0.03863,0.03638,            | DATA 85 |
| * 0.03966,0.03580,0.0 ,0.0 ,4*0.0,                                 | DATA 86 |
| * 0.05315,0.0 ,0.05657,0.04943,0.05500,0.05430,0.05145,            | DATA 87 |
| * 0.04957,0.04037,0.0 ,0.0 ,4*0.0,                                 | DATA 88 |
| * 0.06877,0.0 ,0.07500,0.06525,0.06980,0.07127,0.06783,            | DATA 89 |
| * 0.06230,0.04601,0.0 ,0.0 ,4*0.0,                                 | DATA 90 |
| * 0.08551,0.0 ,0.09432,0.08212,0.08620,0.08778,0.08500,            | DATA 91 |
| * 0.07849,0.05500,0.0 ,0.0 ,4*0.0/                                 | DATA 92 |
| DATA QN2/0.10326,0.0 ,0.11473,0.10005,0.10461,0.10372,0.10093,     | DATA 93 |
| * 0.09618,0.07475,0.0 ,0.0 ,4*0.0,                                 | DATA 94 |
| * 0.12195,0.0 ,0.13657,0.11891,0.12463,0.12081,0.11752,            | DATA 95 |
| * 0.11416,0.09834,0.0 ,0.0 ,4*0.0,                                 | DATA 96 |
| * 0.14144,0.0 ,0.15894,0.13856,0.14500,0.14082,0.13530,            | DATA 97 |
| * 0.13094,0.12500,0.0 ,0.0 ,4*0.0,                                 | DATA 98 |
| * 0.16162,0.0 ,0.18030,0.15896,0.16309,0.16375,0.15626,            | DATA 99 |
| * 0.14808,0.15570,0.0 ,0.0 ,4*0.0,                                 | DATA100 |
| * 0.18251,0.0 ,0.20036,0.18004,0.18118,0.18779,0.17917,            | DATA101 |
| * 0.16583,0.18588,0.0 ,0.0 ,4*0.0,                                 | DATA102 |
| * 0.20410,0.0 ,0.22000,0.20172,0.20000,0.21157,0.20296,            | DATA103 |
| * 0.18381,0.20883,0.0 ,0.0 ,4*0.0,                                 | DATA104 |
| * 0.22636,0.0 ,0.23919,0.22397,0.22181,0.23478,0.22654,            | DATA105 |
| * 0.20294,0.22300,0.0 ,0.0 ,4*0.0,                                 | DATA106 |
| * 0.24918,0.0 ,0.25896,0.24677,0.24487,0.25818,0.24962,            | DATA107 |
| * 0.22500,0.23472,0.0 ,0.0 ,4*0.0/                                 | DATA108 |
| DATA CN3/0.27246,0.0 ,0.28000,0.27006,0.26888,0.28244,0.27269,     | DATA109 |
| * 0.25470,0.24667,0.0 ,0.0 ,4*0.0,                                 | DATA110 |
| * 0.29614,0.0 ,0.30504,0.29380,0.29380,0.30741,0.29568,            | DATA111 |
| * 0.28532,0.26758,0.0 ,0.0 ,4*0.0,                                 | DATA112 |
| * 0.32027,0.0 ,0.33082,0.31790,0.31901,0.33204,0.31848,            | DATA113 |
| * 0.31006,0.29346,0.0 ,0.0 ,4*0.0,                                 | DATA114 |
| * 0.34485,0.0 ,0.35551,0.34237,0.34389,0.35505,0.34152,            | DATA115 |
| * 0.32804,0.32124,0.0 ,0.0 ,4*0.0,                                 | DATA116 |
| * 0.36989,0.0 ,0.37692,0.36720,0.36564,0.37465,0.36500,            | DATA117 |
| * 0.34555,0.35000,0.0 ,0.0 ,4*0.0,                                 | DATA118 |
| * 0.39531,0.0 ,0.39809,0.39239,0.38612,0.39404,0.38941,            | DATA119 |
| * 0.36944,0.37720,0.0 ,0.0 ,4*0.0,                                 | DATA120 |



```

*      0.42105,0.0      ,0.42000,0.41792,0.40720,0.41426,0.41442,DATA121
*      0.40032,0.40540,0.0      ,0.0      ,4*0.0,      DATA122
*      0.44704,0.0      ,0.44625,0.44374,0.43000,0.43804,0.44000,DATA123
*      C.43203,0.43541,0.0      ,0.0      ,4*0.0/      DATA124
DATA CN4/0.47329,0.0      ,0.47321,0.46984,0.45868,0.46531,0.46636,DATA125
*      0.46004,0.46722,0.0      ,0.0      ,4*0.0,      DATA126
*      0.49980,0.0      ,0.50000,0.49619,0.48895,0.49357,0.49309,DATA127
*      0.47849,0.50000,0.0      ,0.0      ,4*0.0,      DATA128
*      C.52658,0.0      ,0.52255,0.52276,0.52000,0.52187,0.52000,DATA129
*      0.49591,C.53532,0.0      ,0.0      ,4*0.0,      DATA130
*      0.55354,0.0      ,0.54481,0.54950,0.55032,0.54925,0.54628,DATA131
*      0.51454,0.56935,0.0      ,0.0      ,4*0.0,      DATA132
*      0.58064,0.0      ,0.56785,0.57640,0.58040,0.57647,0.57285,DATA133
*      C.53810,0.60000,0.0      ,0.0      ,4*0.0,      DATA134
*      0.60777,0.0      ,0.59466,0.60345,0.61000,0.60321,0.60000,DATA135
*      C.56711,0.61544,0.0      ,0.0      ,4*0.0,      DATA136
*      0.63499,0.0      ,0.62485,0.63065,0.63762,0.62964,0.62949,DATA137
*      0.60000,0.62811,0.0      ,0.0      ,4*0.0,      DATA138
*      0.66232,0.0      ,0.65518,0.65795,0.66505,0.65639,0.65877,DATA139
*      C.64092,0.64170,0.0      ,0.0      ,4*0.0/      DATA140
DATA QN5/0.68995,0.0      ,0.68181,0.68531,0.69290,0.68472,0.68624,DATA141
*      0.68136,0.66598,0.0      ,0.0      ,4*0.0,      DATA142
*      C.71770,0.0      ,0.70415,0.71271,0.72342,0.71425,0.71017,DATA143
*      C.71259,0.70010,0.0      ,0.0      ,4*0.0,      DATA144
*      0.74538,0.0      ,0.72585,0.74009,0.75467,0.74303,0.73304,DATA145
*      0.73438,0.73413,0.0      ,0.0      ,4*0.0,      DATA146
*      0.77275,0.0      ,0.74819,0.76738,0.78500,0.76827,0.75578,DATA147
*      0.75500,0.76068,0.0      ,0.0      ,4*0.0,      DATA148
*      0.79979,0.0      ,0.77482,0.79451,0.81165,0.79168,0.77925,DATA149
*      C.78625,0.78027,0.0      ,0.0      ,4*0.0,      DATA150
*      0.82658,0.0      ,0.80515,0.82144,0.83654,0.81500,0.80368,DATA151
*      0.81880,0.80000,0.0      ,0.0      ,4*0.0,      DATA152
*      C.85320,0.0      ,0.83534,0.84814,0.86000,0.84094,0.83114,DATA153
*      0.85000,0.82891,0.0      ,0.0      ,4*0.0,      DATA154
*      C.87954,0.0      ,0.86193,0.87450,0.88253,0.86707,0.85950,DATA155
*      0.86790,0.85564,0.0      ,0.0      ,4*0.0/      DATA156
DATA CN6/0.90546,0.0      ,0.88465,0.90057,0.90414,0.89213,0.88592,DATA157
*      0.88483,0.89000,0.0      ,0.0      ,4*0.0,      DATA158
*      0.93095,0.0      ,0.90690,0.92652,0.92500,0.91607,0.90848,DATA159
*      C.90431,0.91270,0.0      ,0.0      ,4*0.0,      DATA160
*      0.95577,0.0      ,0.93000,0.95244,0.94486,0.94000,0.93000,DATA161
*      0.93690,0.93664,0.0      ,0.0      ,4*0.0,      DATA162
*      0.97976,0.0      ,0.95866,0.97724,0.96475,0.96604,0.95292,DATA163
*      0.97388,0.96677,0.0      ,0.0      ,4*0.0,      DATA164
*      1.00291,0.0      ,0.98673,0.99988,0.98567,0.99000,0.97481,DATA165
*      1.00747,1.00000,0.0      ,0.0      ,4*0.0,      DATA166
*      1.02443,0.0      ,1.01238,1.02048,1.00833,1.00714,0.99374,DATA167
*      1.03300,1.02661,0.0      ,0.0      ,4*0.0,      DATA168
*      1.04465,0.0      ,1.03396,1.03989,1.03000,1.02158,1.01084,DATA169
*      1.05000,1.04631,0.0      ,0.0      ,4*0.0,      DATA170
*      1.06135,0.0      ,1.05000,1.05698,1.05360,1.03814,1.02858,DATA171
*      1.05464,1.05726,0.0      ,0.0      ,4*0.0/      DATA172
DATA CN7/1.08208,0.0      ,1.06517,1.07694,1.06500,1.05000,1.04543,DATA173
*      1.06078,1.06637,0.0      ,0.0      ,4*0.0,      DATA174
*      1.07662,0.0      ,1.05380,1.07562,1.05500,1.05000,1.05000,DATA175
*      1.05500,1.06000,0.0      ,0.0      ,4*0.0,      DATA176
*      1.00000,0.0      ,1.00000,1.00000,1.00000,1.00000,1.00000,DATA177
*      1.00000,1.00000,0.0      ,0.0      ,4*0.0/      DATA178
DATA CN1/0.0      ,0.0      ,0.0      ,0.0      ,0.0      ,0.0      ,0.0      ,DATA179
*      0.0      ,0.0      ,0.0      ,0.0      ,4*0.0,      DATA180

```

```

*      0.05273,0.0      ,0.05584,0.04146,0.05112,0.04500,0.04808,DATA181
*      0.04112,0.04102,0.0      ,0.0      ,4*0.0,      DATA182
*      0.08574,0.0      ,0.01952,0.07033,0.02896,0.04686,0.05584,DATA183
*      0.07380,0.07407,0.0      ,0.0      ,4*0.0,      DATA184
*      0.24194,0.0      ,0.05590,0.09098,0.02312,0.02576,0.04056,DATA185
*      0.10000,0.10000,0.0      ,0.0      ,4*0.0,      DATA186
*      0.41581,0.0      ,0.16479,0.10962,0.07904,0.01584,0.01952,DATA187
*      0.12236,0.11769,0.0      ,0.0      ,4*0.0,      DATA188
*      0.15280,0.0      ,0.19490,0.12921,0.15000,0.05192,0.01000,DATA189
*      0.14141,0.13037,0.0      ,0.0      ,4*0.0,      DATA190
*      0.16653,0.0      ,0.20000,0.14813,0.17576,0.10024,0.05952,DATA191
*      0.15857,0.14036,0.0      ,0.0      ,4*0.0,      DATA192
*      0.18558,0.0      ,0.22500,0.16701,0.19288,0.15000,0.12056,DATA193
*      0.17462,0.15000,0.0      ,0.0      ,4*0.0/      DATA194
DATA DN2/0.20799,0.0      ,0.25000,0.18565,0.21000,0.17042,0.16510,DATA195
*      0.18946,0.16546,0.0      ,0.0      ,4*0.0,      DATA196
*      0.23186,0.0      ,0.27312,0.20401,0.23000,0.18599,0.18659,DATA197
*      0.20315,0.18213,0.0      ,0.0      ,4*0.0,      DATA198
*      0.25386,0.0      ,0.29482,0.22211,0.25000,0.20000,0.20000,DATA199
*      0.21557,0.20000,0.0      ,0.0      ,4*0.0,      DATA200
*      0.27118,0.0      ,0.31454,0.23998,0.27067,0.21995,0.21524,DATA201
*      0.22833,0.22018,0.0      ,0.0      ,4*0.0,      DATA202
*      0.28900,0.0      ,0.33276,0.25769,0.29058,0.24011,0.22872,DATA203
*      0.24230,0.24030,0.0      ,0.0      ,4*0.0,      DATA204
*      0.30658,0.0      ,0.35000,0.27524,0.30754,0.25872,0.24108,DATA205
*      0.25945,0.25788,0.0      ,0.0      ,4*0.0,      DATA206
*      0.32349,0.0      ,0.36697,0.29265,0.32099,0.27595,0.25296,DATA207
*      0.27936,0.27216,0.0      ,0.0      ,4*0.0,      DATA208
*      0.34017,0.0      ,0.38357,0.30990,0.33297,0.29214,0.26500,DATA209
*      0.30000,0.28500,0.0      ,0.0      ,4*0.0/      DATA210
DATA CN3/0.35666,0.0      ,0.40000,0.32704,0.34431,0.30802,0.27784,DATA211
*      0.32040,0.29704,0.0      ,0.0      ,4*0.0,      DATA212
*      0.37298,0.0      ,0.41697,0.34406,0.35653,0.32372,0.29212,DATA213
*      0.34034,0.30892,0.0      ,0.0      ,4*0.0,      DATA214
*      0.38915,0.0      ,0.43372,0.36101,0.37009,0.33894,0.30970,DATA215
*      0.35852,0.32128,0.0      ,0.0      ,4*0.0,      DATA216
*      0.40521,0.0      ,0.45000,0.37790,0.38451,0.35315,0.32982,DATA217
*      0.37595,0.33476,0.0      ,0.0      ,4*0.0,      DATA218
*      0.42117,0.0      ,0.46374,0.39471,0.40000,0.36557,0.35000,DATA219
*      0.39214,0.35000,0.0      ,0.0      ,4*0.0,      DATA220
*      0.43704,0.0      ,0.47747,0.41147,0.41810,0.37833,0.36738,DATA221
*      0.40802,0.36927,0.0      ,0.0      ,4*0.0,      DATA222
*      0.45284,0.0      ,0.49209,0.42818,0.43648,0.39230,0.38390,DATA223
*      0.42372,0.38963,0.0      ,0.0      ,4*0.0,      DATA224
*      0.46858,0.0      ,0.50989,0.44484,0.45374,0.40970,0.40000,DATA225
*      0.43894,0.41023,0.0      ,0.0      ,4*0.0/      DATA226
DATA CN4/0.48430,0.0      ,0.53015,0.46147,0.46805,0.42982,0.41667,DATA227
*      0.45315,0.43045,0.0      ,0.0      ,4*0.0,      DATA228
*      0.50000,0.0      ,0.55000,0.47807,0.48195,0.45000,0.43333,DATA229
*      0.46557,0.45000,0.0      ,0.0      ,4*0.0,      DATA230
*      0.51572,0.0      ,0.56429,0.49468,0.49626,0.46769,0.45000,DATA231
*      0.47833,0.46769,0.0      ,0.0      ,4*0.0,      DATA232
*      0.53146,0.0      ,0.57675,0.51134,0.51352,0.48431,0.45697,DATA233
*      0.49230,0.48431,0.0      ,0.0      ,4*0.0,      DATA234
*      0.54723,0.0      ,0.58834,0.52803,0.53190,0.50000,0.48372,DATA235
*      0.50945,0.50000,0.0      ,0.0      ,4*0.0,      DATA236
*      0.56305,0.0      ,0.60000,0.54474,0.55000,0.51466,0.50000,DATA237
*      0.52936,0.51443,0.0      ,0.0      ,4*0.0,      DATA238
*      0.57892,0.0      ,0.61441,0.56138,0.56416,0.52886,0.51374,DATA239
*      0.55000,0.52851,0.0      ,0.0      ,4*0.0,      DATA240

```

```

*      0.59487,0.0      ,0.62967,0.57804,0.57787,0.54292,0.52747,DATA241
*      C.57000,0.54271,0.0      ,0.0      ,4*0.0/      DATA242
DATA CN5/0.61093,0.0      ,0.64582,0.59478,0.59224,0.55729,0.54209,DATA243
*      0.59000,0.55774,0.0      ,0.0      ,4*0.0,      DATA244
*      0.62710,0.0      ,0.66368,0.61171,0.60950,0.57223,0.55950,DATA245
*      0.61023,0.57388,0.0      ,0.0      ,4*0.0,      DATA246
*      0.64342,0.0      ,0.68209,0.62881,0.62941,0.58780,0.57941,DATA247
*      0.63045,0.59101,0.0      ,0.0      ,4*0.0,      DATA248
*      0.65991,0.0      ,0.70000,0.64609,0.65000,0.60428,0.60000,DATA249
*      0.65000,0.60989,0.0      ,0.0      ,4*0.0,      DATA250
*      0.67659,0.0      ,0.71463,0.66350,0.67064,0.62197,0.62000,DATA251
*      0.66756,0.63005,0.0      ,0.0      ,4*0.0,      DATA252
*      0.69350,0.0      ,0.72807,0.68111,0.69055,0.64047,0.64000,DATA253
*      0.68413,0.65000,0.0      ,0.0      ,4*0.0,      DATA254
*      0.71068,0.0      ,0.74074,0.69901,0.70721,0.65980,0.66000,DATA255
*      0.70000,0.66682,0.0      ,0.0      ,4*0.0,      DATA256
*      0.72816,0.0      ,0.75296,0.71722,0.72031,0.67976,0.68000,DATA257
*      0.71481,0.68318,0.0      ,0.0      ,4*0.0/      DATA258
DATA DN6/0.74602,0.0      ,0.76500,0.73583,0.73286,0.70000,0.70000,DATA259
*      0.72984,0.70000,0.0      ,0.0      ,4*0.0,      DATA260
*      0.76424,0.0      ,0.77784,0.75490,0.74632,0.71731,0.71843,DATA261
*      0.74579,0.71675,0.0      ,0.0      ,4*0.0,      DATA262
*      0.78297,0.0      ,0.79212,0.77447,0.76432,0.73769,0.73865,DATA263
*      0.76417,0.73744,0.0      ,0.0      ,4*0.0,      DATA264
*      0.80235,0.0      ,0.80945,0.79471,0.78448,0.76651,0.76365,DATA265
*      0.78422,0.76651,0.0      ,0.0      ,4*0.0,      DATA266
*      0.82240,0.0      ,0.82936,0.81564,0.80421,0.80000,0.79260,DATA267
*      0.80477,0.80000,0.0      ,0.0      ,4*0.0,      DATA268
*      0.84353,0.0      ,0.85000,0.83759,0.82199,0.82090,0.82088,DATA269
*      0.82532,0.82090,0.0      ,0.0      ,4*0.0,      DATA270
*      0.86563,0.0      ,0.86731,0.86067,0.84363,0.84311,0.85000,DATA271
*      0.85000,0.84311,0.0      ,0.0      ,4*0.0,      DATA272
*      0.88970,0.0      ,0.88769,0.88557,0.87423,0.87978,0.88341,DATA273
*      0.88277,0.87978,0.0      ,0.0      ,4*0.0/      DATA274
DATA DN7/0.91444,0.0      ,0.91400,0.91159,0.90617,0.91576,0.90998,DATA275
*      0.91500,0.91576,0.0      ,0.0      ,4*0.0,      DATA276
*      0.94749,0.0      ,0.95000,0.94520,0.93827,0.95000,0.93871,DATA277
*      0.95000,0.95000,0.0      ,0.0      ,4*0.0,      DATA278
*      1.00000,0.0      ,1.00000,1.00000,1.00000,1.00000,1.00000,DATA279
*      1.00000,1.00000,0.0      ,0.0      ,4*0.0/      DATA280
END      DATA281

```

## Section 4

### STORAGE BLOCK

|                   | <u>Page</u> |
|-------------------|-------------|
| Subroutine STORAG | 127         |
| Subroutine TRTDAT | 130         |
| Subroutine TRCHEK | 140         |
| Subroutine STRDAT | 142         |
| Subroutine TREAT  | 150         |
| Subroutine BYPASS | 169         |
| Subroutine TRLINK | 170         |
| Subroutine SEDIM  | 171         |
| Subroutine HIGHRF | 173         |
| Subroutine KILL   | 176         |
| Subroutine STRAGE | 177         |
| Subroutine PLUGS  | 182         |
| Subroutine SROUTE | 186         |
| Subroutine SPRINT | 188         |
| Subroutine TRCOST | 190         |
| Subroutine INTERP | 198         |
| BLOCK DATA        |             |

|   |                                                                    |         |
|---|--------------------------------------------------------------------|---------|
|   | SUBROUTINE STORAG                                                  | STOR 1  |
| C |                                                                    | STOR 2  |
| C | ME TREATMENT MODEL 3, 23 AUG. 1970. (EJF) (T-STEP BASIS)           | STOR 3  |
| C |                                                                    | STOR 4  |
| C | AVAILABLE SYSVOLS: WATFOR = 08-12,14-16                            | STOR 5  |
| C | FURTH = 08-12,14-15,17-99                                          | STOR 6  |
| C |                                                                    | STOR 7  |
| C | THIS M/PROG TO BE CALLED SUBROUTINE STORAG BY EXEC M               | STOR 8  |
| C |                                                                    | STOR 9  |
|   | COMMON CCNVER,KHOUR,KMIN,L,KMOD,NFLAG,BIG,HEAD1,HEAD2,             | STOR 10 |
|   | * QDESYN,COIF,WAOU(7),WAIN(7),QQUU(7),QQIN(7),WARM(7),QORL,        | STOR 11 |
|   | * BDIF,QCRM(7),BDQU(7),BDIN(7),BCOU(7),BCIN(7),BDRL,               | STOR 12 |
|   | * SSIF,SSIN(7),SSOU(7),SCOU(7),SCIN(7),SSRM(7),SSRL,COIF,CORL,     | STOR 13 |
|   | * ADEPTH(11),AASURF(11),ITREA(7),ISTUR,IPRINT,ICOST,HPPD,          | STOR 14 |
|   | * MODSIZ,ICHEM,ICL2,SCREEN,QQIFMX,DESF,IRANGE,KNTDF,TRIBA,SEDA,    | STOR 15 |
|   | * SQM,SREFFH,BREFFH,NUNITH,UAREAH,UPRAMA,ICHEMH,HM,VOLDAF,ITABLE,  | STOR 16 |
|   | * MODCST,TCTCST,RECIRC,OVRDAF,TSURFA,OVRSED,NSED,JM(7),WTRMT5,     | STOR 17 |
|   | * NSCRN,SCRCAP,SUAREA,FAREAB,NMS,AREAMS,VOLCON,VOLSED,ALJMT,       | STOR 18 |
|   | * BDINT(7),SSINT(7),WAIN(7),BDOUT(7),SSOUT(7),WAOUT(7),            | STOR 19 |
|   | * WARM(7),BDRMT(7),SSRMT(7),CHEMUT(8),CL2UT(8),QMOD(20),WTRMT1,    | STOR 20 |
|   | * QQRMT(7),QQOUT(7),QQRMMX(7),QQQUMX(7),QQRMMN(7),QQQJMN(7),       | STOR 21 |
|   | * BCRMT(7),BCOUT(7),BCRMMX(7),BCQUMX(7),BCRMMN(7),BCQJMN(7),       | STOR 22 |
|   | * SCRMT(7),SCOUT(7),SCRMMX(7),SCQUMX(7),SCRMMN(7),SCQJMN(7),       | STOR 23 |
|   | * BDRM(7),BCRM(7),SCRM(7)                                          | STOR 24 |
|   | COMMON /BLK/DT,NDT,KDT                                             | STOR 25 |
|   | COMMON /BLK1/NAME(4,21)                                            | STOR 26 |
|   | COMMON /BLK2/QCARR(150),BDARR(150)                                 | STOR 27 |
|   | COMMON /BLK3/JS,JNS,NPOLL,SSARR(150),COARR(150),POLL(160,5,5),     | STOR 28 |
|   | * QQ(160,5)                                                        | STOR 29 |
|   | DIMENSION TIME(160),TITLE(40),JN(5),TYTLE(20)                      | STOR 30 |
|   | DATA TYTLE / 4HOUTP, 4HUT F, 4HROM, 4HEXTE,                        | STOR 31 |
|   | * 4HRNAL, 4H STO, 4HRAGE, 4H/TRE,                                  | STOR 32 |
|   | * 4HATME, 4HNT M, 4HODEL, 4HS ,                                    | STOR 33 |
|   | * 4H , 4H , 4H , 4H ,                                              | STOR 34 |
|   | * 4H , 4H , 4H , 4H /                                              | STOR 35 |
|   | NSTIN = 10                                                         | STOR 36 |
|   | NSTOUT = 11                                                        | STOR 37 |
| C | READ FROM DISK                                                     | STOR 38 |
|   | REWIND NSTIN                                                       | STOR 39 |
|   | READ(NSTIN) (TITLE(I),I=1,40)                                      | STOR 40 |
|   | DO 50 I=21,40                                                      | STOR 41 |
|   | 50 TITLE(I) = TYTLE(I-20)                                          | STOR 42 |
|   | WRITE(06,601) (TITLE(I),I=1,40)                                    | STOR 43 |
|   | 601 FORMAT(' ', 20A4)                                              | STOR 44 |
|   | WRITE(NSTOUT) (TITLE(I), I=1,40)                                   | STOR 45 |
|   | READ(NSTIN) NDT,NOUTS,NPOLL,DT,TZERO,TRIBA                         | STOR 46 |
|   | WRITE(NSTOUT) NDT,NOUTS,NPOLL,DT,TZERO,TRIBA                       | STOR 47 |
|   | READ(NSTIN) (JN(N),N=1,NOUTS)                                      | STOR 48 |
|   | WRITE(06,603) (JN(J),J=1,NOUTS)                                    | STOR 49 |
|   | 603 FORMAT('0', 20X, 'TRANSPORT MODEL OUTFALLS AT THE FOLLOWING ', | STOR 50 |
|   | * 'ELEMENT NUMBERS:', /, ' ', 25X, 517)                            | STOR 51 |
|   | WRITE(NSTOUT) (JN(J),J=1,NOUTS)                                    | STOR 52 |
| C | READ FROM CARD                                                     | STOR 53 |
|   | READ(C5,500) JNS                                                   | STOR 54 |
|   | 500 FORMAT(11C)                                                    | STOR 55 |
| C | FIND OUTFLOW DATA FOR SELECTED EXT. ELEM. OF TRANSP.               | STOR 56 |
|   | DO 1000 J=1,NOUTS                                                  | STOR 57 |
|   | 1000 IF(JN(J) .EQ. JNS) GO TO 2000                                 | STOR 58 |
|   | WRITE(06,699) JNS                                                  | STOR 59 |
|   | 699 FORMAT('THE SELECTED EXTERNAL ELEMENT NUMBER,',                | STOR 60 |

```

*      I3, ' IS NOT AVAILABLE FROM THE TRANSPORT MODEL',      STOR 61
*      ' OUTPUT. EXECUTION TERMINATED.')                      STOR 62
STOP                                                         STOR 63
2000 JS = J                                                  STOR 64
      WRITE(6,611) JNS                                       STOR 65
611 FORMAT('C', 20X, 'INPUT TO TREATMENT MODEL SUPPLIED FROM ', STOR 66
*      'TRANSPORT MODEL EXTERNAL ELEMENT NUMBER', I3)      STOR 67
C      JN = EXT. ELEM. NOS. OF TRANSPORT MODEL OUTFALLS     STOR 68
C      NOUTS = NO. OF OUTFALLS FROM TRANSPORT MODEL          STOR 69
C      JNS = JN OF OUTFALL TO BE TREATED                     STOR 70
C      JS = NO. OF OUTFALL FROM TRANS. MOD. TO BE TREATED   STOR 71
C      NPOLL = NO. OF POLLUTANTS                             STOR 72
C      TRIBA = TOTAL TRIBUTARY AREA (ACRES)                   STOR 73
      READ(5,501) NRUNS, DESF                                STOR 74
501 FORMAT(I10, F10.2)                                       STOR 75
      DT = DT/60.0                                           STOR 76
      CONVER = 1000000./(DT*60.*62.4)                       STOR 77
      QQIFMX = 0.0                                           STOR 78
C      NRUNS = NO. OF DIFF. TREATMENT EXECS. TO BE RUN      STOR 79
C      WITH TRANSPORT OUTPUT DATA                           STOR 80
C      DESF = QDESYN/QQIFMX                                  STOR 81
C      FOR DESIGN STORM, SPECIFY DESF (PROGRAM FINDS QDESYN STOR 82
C      FOR OTHER STORMS, READ IN QDESYN (SET DESF .LE. QDES STOR 83
C                                                         STOR 84
      WRITE(6,600) NRUNS, DT, NDT, TRIBA, NOUTS, NPOLL, TZERO STOR 85
600 FORMAT('O', //, 'NUMBER OF RUNS =', I10, /,           STOR 86
*      'TIME-STEP SIZE =', F10.2, ' MIN', /,              STOR 87
*      'NO. TIME-STEPS MODELED =', I10, /,                STOR 88
*      'OTRIBUTARY AREA =', F10.2, ' ACRES', /,           STOR 89
*      'ONO. TRANSP. MOD. OUTFALLS =', I10, /,            STOR 90
*      'ONO. OF POLLUTANTS =', I10, /,                    STOR 91
*      'OTIME ZERO =', F10.1, ' SEC')                     STOR 92
C                                                         STOR 93
C      NRUNS = NO. OF TREATMENT SIMULATION RUNS MADE, THIS STOR 94
C      DT = TIME-STEP SIZE (MIN)                            STOR 95
C      NDT = NO. OF TIME-STEPS TO BE MODELED                STOR 96
C      KDT = TIME-STEP NUMBER                                STOR 97
C                                                         STOR 98
C----- READ INPUT HYDROGRAPHS AND POLLUTUGRAPHS ----- STOR 99
C                                                         STOR100
C      STANDARD UNITS FOR INPUT ARE CFS, LB/MIN & MPN/MIN    STOR101
1100 DO 2222 KDT=1,NDT                                       STOR102
      READ(NSTIN) TIME(KDT),(QQ(KDT,J),J=1,NOUTS),          STOR103
*      ((POLL(KDT,K,J),J=1,NOUTS),K=1,NPOLL)                STOR104
C      STANDARD UNITS FOR INTERNAL COMPUTATIONS ARE          STOR105
C      CFS, MG/L, CU.FT/DT, LB/DT, MPN/100ML, MPN/DT       STOR106
      QQARR(KDT) = QQ(KDT,JS)                                STOR107
      BDARR(KDT) = POLL(KDT,1,JS)*DT                         STOR108
      SSARR(KDT) = POLL(KDT,2,JS)*DT                         STOR109
      GOARR(KDT) = POLL(KDT,3,JS)*DT                         STOR110
      IF (QQARR(KDT) .GT. QQIFMX) QQIFMX = QQARR(KDT)        STOR111
2222 CONTINUE                                                STOR112
3000 DO 7777 KRUN=1,NRUNS                                     STOR113
      WRITE(6,677) KRUN                                       STOR114
677 FORMAT('1----- RUN NO.', 12, ' -----')             STOR115
C                                                         STOR116
C----- READ, CALC. AND WRITE ALL TREATMENT UNIT CHARACTERISTICS STOR117
C                                                         STOR118
C      CALL TRTEAT                                           STOR119
C                                                         STOR120

```

|                                                                 |         |
|-----------------------------------------------------------------|---------|
| IF (NFLAG .EQ. 0) GO TO 9999                                    | STOR121 |
| C                                                               | STOR122 |
| C-----F--- FEED HYDROGRAPH(S) TO STORAGE UNIT(S) -----          | STOR123 |
| C                                                               | STOR124 |
| C                                                               | STOR125 |
| C                                                               | STOR126 |
| C                                                               | STOR127 |
| C                                                               | STOR128 |
| C                                                               | STOR129 |
| 4000 DO 6666 KDT=1,NDT                                          | STOR130 |
| CCC                                                             | STOR131 |
| IF STORAGE UNIT HAS FLOODED, TREAT SHOULDN'T BE CALL            | STOR132 |
| QQIF = QGARR(KDT)                                               | STOR133 |
| BDIF = BDARR(KDT)                                               | STOR134 |
| SSIF = SSARR(KDT)                                               | STOR135 |
| COIF = CGARR(KDT)                                               | STOR136 |
| C                                                               | STOR137 |
| CALL TREAT                                                      | STOR138 |
| C                                                               | STOR139 |
| QQ(KDT,JS) = QQRL                                               | STOR140 |
| POLL(KDT,1,JS) = BDRL/DT                                        | STOR141 |
| POLL(KDT,2,JS) = SSRL/DT                                        | STOR142 |
| POLL(KDT,3,JS) = CORL/DT                                        | STOR143 |
| CCC                                                             | STOR144 |
| PRINT TABLES & WRITE OUTPUTS ON TAPE (DISC - IN ORIG            | STOR145 |
| C                                                               | STOR146 |
| *** - PRESENTLY ONLY FOR RUN 1                                  | STOR147 |
| IF (KRUN .EQ. 1) WRITE(NSTOUT) TIME(KDT),(QO(KDT,J),J=1,NOUTS), | STOR148 |
| * ((POLL(KDT,K,J),J=1,NOUTS),K=1,NPOLL)                         | STOR149 |
| 6666 CONTINUE                                                   | STOR150 |
| IF (ITABLE .EQ. 1) CALL SPRINT                                  | STOR151 |
| C                                                               | STOR152 |
| CALL TRCOST                                                     | STOR153 |
| C                                                               | STOR154 |
| 7777 CONTINUE                                                   | STOR155 |
| C                                                               | STOR156 |
| 8000 WRITE(6,900)                                               | STOR157 |
| 900 FORMAT('1 ')                                                | STOR158 |
| C                                                               | STOR159 |
| GO TO 9999                                                      | STOR160 |
| C                                                               | STOR161 |
| 9999 RETURN                                                     | STOR162 |
| END                                                             | STOR163 |
| C=====                                                          | STOR164 |
| C----- ERROR MESSAGES -----                                     |         |
| C                                                               |         |
| CC GO TO 9999                                                   |         |
| C                                                               |         |
| 9999 RETURN                                                     |         |
| END                                                             |         |
| C=====                                                          |         |

|                                                                   |         |
|-------------------------------------------------------------------|---------|
| SUBROUTINE TRTDAT                                                 | TRTD 1  |
| COMMON CCNVER,KHOUR,KMIN,L,KMOD,NFLAG,BIG,HEAD1,HEAD2,            | TRTD 2  |
| * QUESYA,CQIF,WAOU(7),WAIN(7),QQUO(7),QQIN(7),WARM(7),QQLL,       | TRTD 3  |
| * BOLF,QQRF(7),BDOU(7),BDIN(7),BCOU(7),BCIN(7),BURL,              | TRTD 4  |
| * SSIF,SSIN(7),SSOU(7),SCOU(7),SCIN(7),SSRM(7),SSRL,CQIF,CORL,    | TRTD 5  |
| * ADEPTH(11),AASURF(11),ITREAT(7),ISTOR,IPRINT,ICOST,HRFD,        | TRTD 6  |
| * MODSIZ,ICHEM,ICL2,SCREEN,QQIFMX,DESF,IRANGE,KNTOF,TRIBA,SEDA,   | TRTD 7  |
| * SQM,SREFFH,BREFFH,NUNITH,UAREAH,OPRAMA,ICHEMH,HM,VOLDAF,ITABLE, | TRTD 8  |
| * MODCST,TUTCST,RECIRC,OVRDAF,TSURFA,OVRSED,NSED,JM(7),WTRMT5,    | TRTD 9  |
| * NSCRN,SCRCAP,SUAREA,FAREAB,NMS,AREAMS,VOLCON,VOLSED,ALJMT,      | TRTD 10 |
| * BDINT(7),SSINT(7),WAIN(7),BDOUT(7),SSOUT(7),WAOUT(7),           | TRTD 11 |
| * WARM(7),BDRMT(7),SSRMT(7),CHEMUT(8),CL2UT(8),QMOD(20),WTRMT1,   | TRTD 12 |
| * QQRMT(7),QQOUT(7),QQRMMX(7),QQOUMX(7),QQRMMN(7),QQOUMN(7),      | TRTD 13 |
| * BCRMT(7),BCOUT(7),BCRMMX(7),BCOUMX(7),BCRMMN(7),BCOUMN(7),      | TRTD 14 |
| * SCRMT(7),SCOUT(7),SCRMMX(7),SCOUMX(7),SCRMMN(7),SCOUMN(7),      | TRTD 15 |
| * BDRM(7),BCRM(7),SCRM(7)                                         | TRTD 16 |
| COMMON /IBLK/DT,NDT,KDT                                           | TRTD 17 |
| COMMON /BLK1/NAME(4,21)                                           | TRTD 18 |
| COMMON /BLK2/QQARR(150),BDARR(150)                                | TRTD 19 |
| COMMON /STBK/QIN(150),BODIN(150),SUSIN(150),COLIN(150),           | TRTD 20 |
| * QINST,QOUST,QINSTL,COUSTL,STORL,QOUTO,STORO,                    | TRTD 21 |
| * ISPRIN,IPOL,DEPMAX,QCHAX,DEPTH,                                 | TRTD 22 |
| * ATERM(11),AQ2DT2(11),BDEPTH(11),BSTOR(11),                      | TRTD 23 |
| * DUMSTR(11),DUMDEP(11),                                          | TRTD 24 |
| * VOLIN(150),VOLOUT(150),STOR,CUMIN,CUMOUT,                       | TRTD 25 |
| * SBOD,SSS,SCOL,BODOUT,SUSOUT,COLOUT,                             | TRTD 26 |
| * ISTMCD,ISTTYP,ISTOUT,                                           | TRTD 27 |
| * QPUMP,DSTART,DSTOP,                                             | TRTD 28 |
| * DTON,STORMX,DTPUMP,DTMORE,STORF,APLAN,                          | TRTD 29 |
| * CLXRC,CSTOR,CPS,CTOTAL,CPCUYD,CPACRE,                           | TRTD 30 |
| * LP,JP,LPREV,LABEL,DETENT(150),FRAC(150)                         | TRTD 31 |
| C                                                                 | TRTD 32 |
| WRITE(6,601)                                                      | TRTD 33 |
| 601 FORMAT('OINPUT DATA FOR TREATMENT PACKAGE FOLLOWS')           | TRTD 34 |
| READ(5,501) ISTOR, (ITREAT(K), K=1,7)                             | TRTD 35 |
| 501 FORMAT(10I5)                                                  | TRTD 36 |
| C                                                                 | TRTD 37 |
| C                                                                 | TRTD 38 |
| C                                                                 | TRTD 39 |
| C                                                                 | TRTD 40 |
| C                                                                 | TRTD 41 |
| C                                                                 | TRTD 42 |
| C                                                                 | TRTD 43 |
| C                                                                 | TRTD 44 |
| C                                                                 | TRTD 45 |
| C                                                                 | TRTD 46 |
| C                                                                 | TRTD 47 |
| C                                                                 | TRTD 48 |
| C                                                                 | TRTD 49 |
| C                                                                 | TRTD 50 |
| C                                                                 | TRTD 51 |
| C                                                                 | TRTD 52 |
| C                                                                 | TRTD 53 |
| C                                                                 | TRTD 54 |
| C                                                                 | TRTD 55 |
| C                                                                 | TRTD 56 |
| C                                                                 | TRTD 57 |
| C                                                                 | TRTD 58 |
| C                                                                 | TRTD 59 |
| C                                                                 | TRTD 60 |

NOTE.. OPERATIONAL OPTIONS INDICATED BY '\*'

ISTOR = INDICATOR TO SEPARATE STORAGE MODELING

\* 01 = NONE

\* 02 = FLOW THRU STORAGE MODELED SEPARATELY

ITREAT(1) = TREATMENT PARAMETER (LEVEL 1 = BAR RACKS)

\* 11 = NO BAR RACKS (BYPASS)

\* 12 = BAR RACKS

ITREAT(2) = TREATMENT PARAMETER (LEVEL 2 = INLET PUMP)

\* 21 = GRAVITY = NO PUMPING (BYPASS)

\* 22 = PUMPING (STATION)

ITREAT(3) = TREATMENT PARAMETER (LEVEL 3 = PRIMARY)

\* 31 = NONE (BYPASS)

\* 32 = DISSOLVED AIR FLOATATION

\* 33 = FINE SCREENS & DISS AIR FLOATATION

\* 34 = FINE SCREENS (ONLY)

\* 35 = SEDIMENTATION

ITREAT(4) = TREATMENT PARAMETER (LEVEL 4 = SECONDARY)

\* 41 = NONE (BYPASS)

\* 42 = MICROSTRAINERS

\* 43 = HIGH RATE FILTERS

ITREAT(5) = TREATMENT PARAMETER (LEVEL 5 = EFFLUENT)

\* 51 = NO EFFLUENT SCREENS



```

C          *          = 52 = EFFLUENT SCREENS          TRTD 61
C          ITREAT(6) = TREATMENT PARAMETER (LEVEL 6 = OUTLET PUTRTD 62
C          *          = 61 = GRAVITY = NO PUMPING (BYPASS) TRTD 63
C          *          = 62 = PUMPING (STATION)          TRTD 64
C          ITREAT(7) = TREATMENT PARAMETER (LEVEL 7 = CL2 CONTATRTD 65
C          *          = 71 = NO CONTACT TANK (BYPASS)   TRTD 66
C          *          = 72 = CONTACT TANK              TRTD 67
C                                                    TRTD 68
C          WRITE(6,602)  ISTR, (NAME(I,ISTR), I=1,4)    TRTD 69
602 FORMAT('0',   ///,
C          *          '0', 27X, 'CHARACTERISTICS OF THE TREATMENT PACKAGE ARE', /,TRTD 71
C          *          'C', 32X, 'LEVEL', 5X, 'MODE', 10X, 'PROCESS', //, TRTD 72
C          *          'C', 34X, '0', 8X, '0', 11, 8X, 4A4) TRTD 73
C          DO 20 L=1,7 TRTD 74
C          LR = ITREAT(L) - L*10 TRTD 75
C          GO TO (1,2,3,4,5,6,7), L TRTD 76
C          TRTD 77
C          TRTD 78
C          1 J1 = LR + 2 TRTD 79
C          JM(1) = J1 TRTD 80
C          GO TO 8 TRTD 81
C          2 J2 = LR + 4 TRTD 82
C          JM(2) = J2 TRTD 83
C          GO TO 8 TRTD 84
C          3 J3 = LR + 6 TRTD 85
C          IF (LR .EQ. 5 .AND. ISTR .EQ. 2) J3 = 21 TRTD 86
C          JM(3) = J3 TRTD 87
C          GO TO 8 TRTD 88
C          4 J4 = LR + 11 TRTD 89
C          JM(4) = J4 TRTD 90
C          GO TO 8 TRTD 91
C          5 J5 = LR + 14 TRTD 92
C          JM(5) = J5 TRTD 93
C          GO TO 8 TRTD 94
C          6 J6 = LR + 16 TRTD 95
C          JM(6) = J6 TRTD 96
C          GO TO 8 TRTD 97
C          7 J7 = LR + 18 TRTD 98
C          JM(7) = J7 TRTD 99
C          TRTD100
C          8 J = JM(L) TRTD101
C          TRTD102
C          IF (LR .EQ. 1) GO TO 10 TRTD103
C          WRITE(6,603) L, ITREAT(L), (NAME(I,J), I=1,4) TRTD104
603 FORMAT(' ', 34X, 11, 8X, 12, 8X, 4A4) TRTD105
C          GO TO 20 TRTD106
C          10 WRITE(6,604) L, ITREAT(L) TRTD107
604 FORMAT(' ', 34X, 11, 8X, 12, 29X, '(BYPASS)') TRTD108
C          20 CONTINUE TRTD109
C          TRTD110
C          CALL TRCHEK TRTD111
C          TRTD112
C          30 IF (NFLAG .EQ. 0) GO TO 9999 TRTD113
C          TRTD114
C          IPRINT = 0 = NO PRINTOUT EACH TIME-STEP (SUMMARY POSTRTD115
C          = 1 = PRINTOUT SOLUTION EACH TIME-STEP (QUANTRTD116
C          = 2 = PRINTOUT SOLUTION EACH TIME-STEP (QUALITRTD117
C          ICOST = 0 = NO COST COMPUTATIONS AND SUMMARY TRTD118
C          = 1 = COMPUTE COSTS AND SUMMARIZE TRTD119
C          IRANGE = 0 = QUANTITY RANGES (MAX,AV,MIN) NOT SUMMARTRTD120

```

|   |                                                                              |                                                   |         |
|---|------------------------------------------------------------------------------|---------------------------------------------------|---------|
| C |                                                                              | = 1 = QUANTITY RANGES (MAX,AV,MIN) SUMMARIZED     | TRTD121 |
| C | ITABLE = 0                                                                   | = INFLOWS,OUTFLOWS NOT SUMMARIZED IN FINAT        | TRTD122 |
| C |                                                                              | = 1 = INFLOWS AND OUTFLOWS SUMMARIZED IN FINAT    | TRTD123 |
| C |                                                                              |                                                   | TRTD124 |
|   | READ(5,502) IPRINT, ICOST, IRANGE, ITABLE                                    |                                                   | TRTD125 |
|   | 502 FURMAT(4I10)                                                             |                                                   | TRTD126 |
|   | WRITE(6,605) IPRINT, ICOST, IRANGE, ITABLE                                   |                                                   | TRTD127 |
|   | 605 FORMAT('0', 75X, ' IPRINT =', I2, ', ' , ICOST =', I2, ', ' , IRANGE =', |                                                   | TRTD128 |
|   | * I2, ', ' , ITABLE =', I2)                                                  |                                                   | TRTD129 |
|   | KPASS = 0                                                                    |                                                   | TRTD130 |
|   | IF (DESF .LE. 0.0) GO TO 50                                                  |                                                   | TRTD131 |
|   | QDESYN = QQIFMX*DESF                                                         |                                                   | TRTD132 |
|   | WRITE(6,608) QDESYN, DESF, QQIFMX                                            |                                                   | TRTD133 |
|   | 608 FORMAT('0', //, '0', 5X, 'DESIGN STORM USED.',                           |                                                   | TRTD134 |
|   | * ' TREATMENT CAPACITY WILL BE SELECTED TO SUIT.', //                        |                                                   | TRTD135 |
|   | * '0', 15X, 'DESIGN FLOWRATE =', F10.2, ' CFS.', /,                          |                                                   | TRTD136 |
|   | * ' ', 20X, '(=' F6.3, ' TIMES MAXIMUM ARRIVAL RATE OF',                     |                                                   | TRTD137 |
|   | * F10.2, ' CFS.))                                                            |                                                   | TRTD138 |
|   | GO TO 60                                                                     |                                                   | TRTD139 |
|   | 50 READ(5,503) QDESYN                                                        |                                                   | TRTD140 |
|   | 503 FORMAT(F10.2)                                                            |                                                   | TRTD141 |
|   | 55 WRITE(6,609) QDESYN                                                       |                                                   | TRTD142 |
|   | 609 FORMAT('C', 5X, 'SPECIFIED TREATMENT CAPACITY USED.',//,                 |                                                   | TRTD143 |
|   | * '0', 15X, 'DESIGN FLOWRATE =', F10.2, ' CFS.))                             |                                                   | TRTD144 |
|   | 60 IF (ITREAT(3) .GT. 31) GO TO 70                                           |                                                   | TRTD145 |
|   | IF (ITREAT(4) .GT. 41) GO TO 70                                              |                                                   | TRTD146 |
|   | IF (ITREAT(5) .GT. 51) GO TO 70                                              |                                                   | TRTD147 |
|   | KMOD = 1                                                                     |                                                   | TRTD148 |
|   | QMOD(1) = QDESYN/1.547                                                       |                                                   | TRTD149 |
|   | GO TO 90                                                                     |                                                   | TRTD150 |
|   | 70 QDSMGD = QDESYN/1.547                                                     |                                                   | TRTD151 |
| C |                                                                              |                                                   | TRTD152 |
| C |                                                                              | SEARCH FOR SMALLEST MODULE SIZE (MGD) .GT. QDSMGD | TRTD153 |
| C |                                                                              |                                                   | TRTD154 |
|   | QMOD(1) = 5.0                                                                |                                                   | TRTD155 |
|   | QMOD(2) = 10.0                                                               |                                                   | TRTD156 |
|   | QMOD(3) = 15.0                                                               |                                                   | TRTD157 |
|   | QMOD(4) = 20.0                                                               |                                                   | TRTD158 |
|   | QMOD(5) = 25.0                                                               |                                                   | TRTD159 |
|   | QMOD(6) = 30.0                                                               |                                                   | TRTD160 |
|   | QMOD(7) = 35.0                                                               |                                                   | TRTD161 |
|   | QMOD(8) = 50.0                                                               |                                                   | TRTD162 |
|   | QMOD(9) = 75.0                                                               |                                                   | TRTD163 |
|   | QMOD(10) = 100.0                                                             |                                                   | TRTD164 |
|   | QMOD(11) = 125.0                                                             |                                                   | TRTD165 |
|   | QMOD(12) = 150.0                                                             |                                                   | TRTD166 |
|   | QMOD(13) = 200.0                                                             |                                                   | TRTD167 |
|   | QMOD(14) = 250.0                                                             |                                                   | TRTD168 |
|   | QMOD(15) = 300.0                                                             |                                                   | TRTD169 |
|   | QMOD(16) = 350.0                                                             |                                                   | TRTD170 |
|   | QMOD(17) = 400.0                                                             |                                                   | TRTD171 |
|   | QMOD(18) = 450.0                                                             |                                                   | TRTD172 |
|   | QMOD(19) = 500.0                                                             |                                                   | TRTD173 |
|   | DO 75 K=1,19                                                                 |                                                   | TRTD174 |
|   | IF (QMOD(K) - QDSMGD) 75,80,80                                               |                                                   | TRTD175 |
|   | 75 CONTINUE                                                                  |                                                   | TRTD176 |
|   | GO TO 903                                                                    |                                                   | TRTD177 |
|   | 80 QDESYN = 1.547*QMOD(K)                                                    |                                                   | TRTD178 |
|   | KMOD = K                                                                     |                                                   | TRTD179 |
|   | IF (KMOD .LT. 8) GO TO 49                                                    |                                                   | TRTD180 |

|                                                                   |         |
|-------------------------------------------------------------------|---------|
| IF (KMCD .LT. 13) GO TO 51                                        | TRTD181 |
| IF (KMCD .LT. 20) GO TO 52                                        | TRTD182 |
| GO TO 903                                                         | TRTD183 |
| C                                                                 | TRTD184 |
| 49 MODSIZ = 5                                                     | TRTD185 |
| NOUNIT = QMOD(KMOD)/5.0 + 0.1                                     | TRTD186 |
| GO TO 53                                                          | TRTD187 |
| C                                                                 | TRTD188 |
| 51 MODSIZ = 25                                                    | TRTD189 |
| NOUNIT = QMOD(KMOD)/25.0 + 0.1                                    | TRTD190 |
| GO TO 53                                                          | TRTD191 |
| C                                                                 | TRTD192 |
| 52 MODSIZ = 50                                                    | TRTD193 |
| NOUNIT = QMOD(KMOD)/50.0 + 0.1                                    | TRTD194 |
| C                                                                 | TRTD195 |
| 53 CONTINUE                                                       | TRTD196 |
| WRITE(6,680) QDESYN, QMOD(KMOD), KMOD                             | TRTD197 |
| 680 FORMAT(' ', 15X, 'TREATMENT SYSTEM INCLUDES MODULE UNITS', /, | TRTD198 |
| * ' ', 20X, 'DESIGN FLOW IS THEREFORE INCREASED TO NEXT',         | TRTD199 |
| * ' LARGEST MODULE SIZE', /,                                      | TRTD200 |
| * ' ', 20X, 'ADJUSTED DESIGN FLOWRATE =', F10.2, ' CFS., =',      | TRTD201 |
| * F10.2, ' MGD.', /, ' ', 20X, '(KMOD =', I3, ')')                | TRTD202 |
| 90 CONTINUE                                                       | TRTD203 |
| C                                                                 | TRTD204 |
| C                                                                 | TRTD205 |
| C                                                                 | TRTD206 |
| FIND CHLORINATOR CAPACITY REQUIRED = PCL2MX (LB/DAY)              | TRTD207 |
| PCL2MX = 0.0                                                      | TRTD208 |
| DO 96 KDT=1,NDT                                                   | TRTD209 |
| IF (QQARR(KDT) .GT. 0.0) GO TO 92                                 | TRTD210 |
| BCIF = 0.0                                                        | TRTD211 |
| GO TO 94                                                          | TRTD212 |
| 92 BCIF = 16050.0*BDARR(KDT)/(QQARR(KDT)*DT*60.0)                 | TRTD213 |
| 94 CL2DEM = BCIF/10.0                                             | TRTD214 |
| IF (CL2DEM .LT. 6.0) CL2DEM = 6.0                                 | TRTD215 |
| IF (CL2DEM .GT. 25.0) CL2DEM = 25.0                               | TRTD216 |
| PCL2DM = CL2DEM*QDESYN*0.646*8.35                                 | TRTD217 |
| 96 IF (PCL2DM .GT. PCL2MX) PCL2MX = PCL2DM                        | TRTD218 |
| C                                                                 | TRTD219 |
| C----- READ TREATMENT PARAMETERS                                  | TRTD220 |
| C                                                                 | TRTD221 |
| C----- LEVEL 0 BRANCH (FOR ASSOCIATED STORAGE)                    | TRTD222 |
| C                                                                 | TRTD223 |
| IF (KPASS .EQ. 1) GO TO 1000                                      | TRTD224 |
| GO TO ( ECC, 500), I\$TOR                                         | TRTD225 |
| C                                                                 | TRTD226 |
| C                                                                 | TRTD227 |
| C                                                                 | TRTD228 |
| 800 WRITE(6,6C6)                                                  | TRTD229 |
| 606 FORMAT('C', 15X, 'NO STORAGE FROM A SEPARATE STORAGE MODEL',  | TRTD230 |
| * ' IS ASSOCIATED WITH THIS TREATMENT MODEL')                     | TRTD231 |
| GO TO 1000                                                        | TRTD232 |
| C                                                                 | TRTD233 |
| C                                                                 | TRTD234 |
| C                                                                 | TRTD235 |
| 900 CALL STREAT                                                   | TRTD236 |
| C                                                                 | TRTD237 |
| QRAT = QDESYN/QCQMAX                                              | TRTD238 |
| IF (QRAT .LT. 1.20) GO TO 1000                                    | TRTD239 |
| WRITE(6,6E1) QCQMAX                                               | TRTD240 |
| 681 FORMAT('O', 5X, 'DESIGN FLOW INPUT TO TREATMENT WILL BE '     |         |

|                                                                           |         |
|---------------------------------------------------------------------------|---------|
| *        'CONSIDERABLY RESTRICTED BY MAXIMUM POSSIBLE OUTFLOW',           | TRTD241 |
| *        ' FROM STORAGE =' , F10.2, ' CFS', /,                            | TRTD242 |
| *        ' ', 20X, 'THEREFORE REDUCE TREATMENT DESIGN FLW')               | TRTD243 |
| QDESYN = QCMAX                                                            | TRTD244 |
| KPASS = 1                                                                 | TRTD245 |
| GO TO 55                                                                  | TRTD246 |
| C-----                                                                    | TRTD247 |
| LEVEL 1 BRANCH (FOR TREATMENT BY BAR RACKS)                               | TRTD248 |
| C                                                                         | TRTD249 |
| 1000    K = ITREAT(1) - 10                                                | TRTD250 |
| L = 1                                                                     | TRTD251 |
| GO TO (1100,1200), K                                                      | TRTD252 |
| C                                                                         | TRTD253 |
| C                            11) FOR NO BAR RACKS                         | TRTD254 |
| C                                                                         | TRTD255 |
| 1100 WRITE(6,611)                                                         | TRTD256 |
| 611 FORMAT('0', 15X, 'BAR RACKS NOT INCLUDED (LEVEL 1)')                  | TRTD257 |
| GO TO 20C0                                                                | TRTD258 |
| C                                                                         | TRTD259 |
| C                            12) FOR BAR RACKS                            | TRTD260 |
| C                                                                         | TRTD261 |
| 1200    NSCRN = QDESYN/240.0                                              | TRTD262 |
| IF (NSCRN .LT. 2) NSCRN = 2                                               | TRTD263 |
| SCRCAP = QDESYN/NSCRN                                                     | TRTD264 |
| SUAREA = SCRCAP/3.0                                                       | TRTD265 |
| FAREAB = 1.4*SUAREA                                                       | TRTD266 |
| WRITE(6,612) NSCRN,SCRCAP,SUAREA,FAREAB                                   | TRTD267 |
| 612 FORMAT('0', 15X, 'PRELIMINARY TREATMENT BY MECHANICALLY CLEANED ',    | TRTD268 |
| *        'BAR RACKS (LEVEL 1)',                                           | TRTD269 |
| * /, ' ', 20X, 'NUMBER OF SCREENS =', I7, /, ' ',                         | TRTD270 |
| *        20X, 'CAPACITY PER SCREEN =' , F10.2, ' CFS', /, ' ',            | TRTD271 |
| *        20X, 'SUBMERGED AREA =', F10.2,                                  | TRTD272 |
| *        ' SQ.FT. (PERPENDICULAR TO THE FLOW)', /,                        | TRTD273 |
| * ' ', 20X, 'FACE AREA OF BARS =', F10.2, ' SQ.FT.')                      | TRTD274 |
| C-----                                                                    | TRTD275 |
| LEVEL 2 BRANCH (FOR INLET PUMPING)                                        | TRTD276 |
| C                                                                         | TRTD277 |
| 2000    K = ITREAT(2) - 20                                                | TRTD278 |
| L = 2                                                                     | TRTD279 |
| GO TO (2100,2200), K                                                      | TRTD280 |
| C                                                                         | TRTD281 |
| C                            21) FOR GRAVITY INLET (NO PUMPING, = BYPASS) | TRTD282 |
| C                                                                         | TRTD283 |
| 2100 WRITE(6,621)                                                         | TRTD284 |
| 621 FORMAT('0', 15X, 'INFLOW BY GRAVITY (NO PUMPING) (LEVEL 2)')          | TRTD285 |
| GO TO 3000                                                                | TRTD286 |
| C                                                                         | TRTD287 |
| C                            22) FOR INLET PUMPING (STATION)              | TRTD288 |
| C                                                                         | TRTD289 |
| 2200 READ(5,522) HEAD1                                                    | TRTD290 |
| 522 FORMAT(F10.2)                                                         | TRTD291 |
| WRITE(6,622) HEAD1                                                        | TRTD292 |
| 622 FORMAT('0', 15X, 'INFLOW BY INLET PUMPING (LEVEL 2)', /,              | TRTD293 |
| *        ' ', 20X, 'PUMPED HEAD =' , F7.2, ' FT. WATER')                  | TRTD294 |
| C                                                                         | TRTD295 |
| C-----                                                                    | TRTD296 |
| LEVEL 3 BRANCH (FOR PRIMARY TREATMENT)                                    | TRTD297 |
| C                                                                         | TRTD298 |
| 3000    K = ITREAT(3) - 30                                                | TRTD299 |
| L = 3                                                                     | TRTD300 |
| GO TO (3100,3200,3200,3400,3500), K                                       |         |

|     |                                                                       |         |
|-----|-----------------------------------------------------------------------|---------|
| C   |                                                                       | TRTD301 |
| C   | 31) FOR NO PRIMARY TREATMENT (BYPASS)                                 | TRTD302 |
| C   |                                                                       | TRTD303 |
|     | 3100 WRITE(6,631)                                                     | TRTD304 |
|     | 631 FORMAT('0', 15X, 'NO PRIMARY TREATMENT INCLUDED (LEVEL 3)')       | TRTD305 |
|     | GO TO 4000                                                            | TRTD306 |
| C   |                                                                       | TRTD307 |
| C   | 32) & 33) FOR DISSOLVED AIR FLOATATION                                | TRTD308 |
| C   |                                                                       | TRTD309 |
|     | 3200 READ (5,450) ICHEM, ICL2, OVRDAF, RECIRC, DEEP                   | TRTD310 |
|     | 450 FORMAT (2I5, 3F10.2)                                              | TRTD311 |
|     | TSURFA = (1.0+0.01*RECIRC)*QDESYN*1000000.0/(1.547*OVRDAF)            | TRTD312 |
| C?? | DETMIN = TSURFA*DEEP/((1.0+0.01*RECIRC)*QDESYN*60.0)                  | TRTD313 |
| C   | OVRDAF = DESIGN OVERFLOW RATE, GPD/SF (5000.00 SUGGESTED)             | TRTD314 |
| C   | RECIRC = RECIRCULATION FLOW, PERCENT (15.00 SUGGESTED)                | TRTD315 |
| C   | TSURFA = TOTAL SURFACE AREA, SQ.FT                                    | TRTD316 |
| C   | DETMIN = DETENTION TIME, MIN                                          | TRTD317 |
|     | VOLDAF = TSURFA*DEEP                                                  | TRTD318 |
|     | WRITE (6,453) MODSIZ,NOUNIT,QMOD(KMOD),QDESYN,OVRDAF,RECIRC,          | TRTD319 |
|     | * CEEP,TSURFA                                                         | TRTD320 |
|     | 453 FORMAT( '0',15X,'TREATMENT BY DISSOLVED AIR FLOATATION (LEVEL 3)' | TRTD321 |
|     | .,/21X,'MODULE SIZE =', 17, ' MGD',/21X,'NUMBER OF UNITS =',          | TRTD322 |
|     | .17/21X,'TOTAL DESIGN FLOW =',F7.2,' MGD, =', F7.2,' CFS', /,         | TRTD323 |
|     | * ' ', 20X, 'DESIGN OVERFLOW RATE =', F10.2,                          | TRTD324 |
|     | * ' GPD/SF, (5000 SUGGESTED)', /,                                     | TRTD325 |
|     | * ' ', 20X, 'RECIRCULATION FLOW =', F10.2,                            | TRTD326 |
|     | * ' PERCENT (15 SUGGESTED)', /,                                       | TRTD327 |
|     | * ' ', 20X, 'TANK DEPTH =', F10.2, ' FEET', /,                        | TRTD328 |
|     | * ' ', 20X, 'TOTAL SURFACE AREA =', F10.2, ' SQ.FT.')                 | TRTD329 |
|     | IF (ICHEM .GT. 0) WRITE(6,455)                                        | TRTD330 |
|     | 455 FORMAT(' ', 20X, 'CHEMICALS WILL BE ADDED')                       | TRTD331 |
|     | IF (ICHEM .EQ. 0) WRITE(6,456)                                        | TRTD332 |
|     | 456 FORMAT(' ', 20X, 'NO CHEMICALS ADDED')                            | TRTD333 |
|     | IF (ICL2 .GT. 0) WRITE(6,457)                                         | TRTD334 |
|     | 457 FORMAT(' ', 20X, 'CHLORINE WILL BE ADDED')                        | TRTD335 |
|     | IF (ICL2 .EQ. 0) WRITE(6,458)                                         | TRTD336 |
|     | 458 FURMAT(' ', 20X, 'NO CHLORINE ADDED')                             | TRTD337 |
|     | IF (ITREAT(3) .EQ. 33) GO TO 3400                                     | TRTD338 |
|     | GO TO 4000                                                            | TRTD339 |
| C   |                                                                       | TRTD340 |
| C   | 33) & 34) FOR FINE SCREENS                                            | TRTD341 |
| C   |                                                                       | TRTD342 |
|     | 3400 SCREEN=QDESYN*449./50.                                           | TRTD343 |
| C   | SCREEN=AREA OF SCREENS ASSUMING DESIGN HYDRAULIC LOADING OF 50 GPM/SQ | TRTD344 |
|     | IF (ITREAT(3) .EQ. 33) WRITE(6,461)                                   | TRTD345 |
|     | 461 FORMAT('0',15X,'TREATMENT BY FINE SCREENS',                       | TRTD346 |
|     | * ' (AHEAD OF DISSOLVED AIR FLOATATION) (LEVEL 3)')                   | TRTD347 |
|     | IF (ITREAT(3) .EQ. 34) WRITE(6,462)                                   | TRTD348 |
|     | 462 FORMAT('0',15X,'TREATMENT BY FINE SCREENS')                       | TRTD349 |
|     | WRITE (6,460) SCREEN                                                  | TRTD350 |
|     | 460 FORMAT (' ',20X,'TOTAL SCREEN AREA =', F10.0,' SQUARE FEET')      | TRTD351 |
|     | GO TO 4000                                                            | TRTD352 |
| C   |                                                                       | TRTD353 |
| C   | 35) FOR SEDIMENTATION                                                 | TRTD354 |
| C   |                                                                       | TRTD355 |
|     | 3500 GO TO (3550,3600), ISTORE                                        | TRTD356 |
| C   | SED TANKS TO BE INSTALLED (NO ASSOCIATED STORAGE)                     | TRTD357 |
|     | 3550 READ(5,535) OVRSED, SEDEP, ICL2                                  | TRTD358 |
|     | 535 FORMAT(2F10.2, I10)                                               | TRTD359 |
| C   | NOTE: SEDEP MAY VARY AS TANK FILLS                                    | TRTD360 |

|     |                                                                   |         |
|-----|-------------------------------------------------------------------|---------|
| C   | ASSUME 10000 SQ. FT. PER UNIT (70 BY 140 FT)                      | TRTD361 |
| C   | OVRSED = OVERFLOW RATE IN SED TANK, (GPD/SF)                      | TRTD362 |
| C   | NO MODULES REQUIRED HERE                                          | TRTD363 |
|     | SEDNUM = QMCD(KMOD)*1000000.0/(OVRSED*10000.0)                    | TRTD364 |
|     | NSED = SEDNUM + 1.0                                               | TRTD365 |
|     | SEDA = 10000.0*SEDNUM/NSED                                        | TRTD366 |
|     | VOLSED = SEDA*SEDEP*NSED                                          | TRTD367 |
|     | WRITE(6,635) OVRSED, SEDEP, NSED, SEDA                            | TRTD368 |
| 635 | FORMAT('O', 15X, 'TREATMENT BY SEDIMENTATION (LEVEL 3)',          | TRTD369 |
| *   | ' - (NO ASSOCIATED STORAGE)', /,                                  | TRTD370 |
| *   | ' ', 20X, 'DESIGN OVERFLOW RATE =', F10.2, ' GPD/SQ.FT.',         | TRTD371 |
| *   | ' (1600 SUGGESTED)', /,                                           | TRTD372 |
| *   | ' ', 20X, 'SED TANK DEPTH =', F10.2, ' FEET',                     | TRTD373 |
| *   | 4X, ' (8 FEET SUGGESTED)', /,                                     | TRTD374 |
| *   | ' ', 20X, 'NUMBER OF SED TANKS =', I10, /,                        | TRTD375 |
| *   | ' ', 20X, 'SURFACE AREA =', F10.2, ' SQ.FT./TANK') TRTD376        |         |
|     | GO TO 3900                                                        | TRTD377 |
| C   | SEDIMENTATION IN ASSOCIATED STORAGE UNIT                          | TRTD378 |
| C   | INPUT HANDLED IN LEVEL 0                                          | TRTD379 |
|     | 3600 WRITE(6,636)                                                 | TRTD380 |
|     | 636 FORMAT('O', 15X, 'TREATMENT BY SEDIMENTATION IN ASSOCIATED',  | TRTD391 |
|     | * ' STORAGE - SEE LEVEL 0 ABOVE')                                 | TRTD382 |
|     | 3900 IF (ICL2 .GT. 0) WRITE(6,457)                                | TRTD383 |
|     | IF (ICL2 .EQ. 0) WRITE(6,458)                                     | TRTD384 |
| C   |                                                                   | TRTD385 |
| C   | ----- LEVEL 4 BRANCH (FOR SECONDARY TREATMENT)                    | TRTD386 |
| C   |                                                                   | TRTD387 |
|     | 4000 K = ITREAT(4) - 40                                           | TRTD388 |
|     | L = 4                                                             | TRTD389 |
|     | GO TO (4100,4200,4300), K                                         | TRTD390 |
| C   |                                                                   | TRTD391 |
| C   | 41) FOR NO SECONDARY TREATMENT (BYPASS)                           | TRTD392 |
| C   |                                                                   | TRTD393 |
|     | 4100 WRITE(6,641)                                                 | TRTD394 |
|     | 641 FORMAT('O', 15X, 'NO SECCNDARY TREATMENT INCLUDED (LEVEL 4)') | TRTD395 |
|     | GO TO 5000                                                        | TRTD396 |
| C   |                                                                   | TRTD397 |
| C   | 42) FOR MICROSTRAINERS                                            | TRTD398 |
| C   |                                                                   | TRTD399 |
| C   | DESIGN LOADING = 40 GPM/SQ.FT OF SUBMERGED AREA                   | TRTD400 |
|     | 4200 IF (QMOC(KMOD) .GE. 25.0) GO TO 4210                         | TRTD401 |
| C   | QDESYN .LT. 25 MGD                                                | TRTD402 |
|     | NMS = KMOD                                                        | TRTD403 |
|     | CAPMS = 5.0                                                       | TRTD404 |
|     | AREAMS = QMOD(KMOD)*(1000000.0/1440.0)/(40.0*NMS)                 | TRTD405 |
| C   | NMS = NO. OF MICROSTRAINER UNITS                                  | TRTD406 |
| C   | CAPMS = CAPACITY PER UNIT (MGD)                                   | TRTD407 |
| C   | AREAMS = SUBMERGED SCREEN AREA (SQ.FT/UNIT)                       | TRTD408 |
|     | GO TO 4220                                                        | TRTD409 |
| C   | QDESYN .GE. 25 MGD                                                | TRTD410 |
|     | 4210 NMS = (QDESYN + 0.1)/19.34                                   | TRTD411 |
|     | CAPMS = 12.5                                                      | TRTD412 |
|     | AREAMS = 12.5*(1000000.0/1440.0)/40.0                             | TRTD413 |
|     | 4220 CONTINUE                                                     | TRTD414 |
|     | WRITE(6,642) NMS, CAPMS, AREAMS                                   | TRTD415 |
|     | 642 FORMAT('O', 15X, 'TREATMENT BY MICROSTRAINERS', /,            | TRTD416 |
| *   | ' ', 20X, 'NUMBER OF UNITS =', I10, /,                            | TRTD417 |
| *   | ' ', 20X, 'CAPACITY PER UNIT =', F10.2, ' MGD', /,                | TRTD418 |
| *   | ' ', 20X, 'SUBMERGED SCREEN AREA =', F10.2,                       | TRTD419 |
| *   | ' SQ.FT. PER UNIT')                                               | TRTD420 |

|                                                                     |         |
|---------------------------------------------------------------------|---------|
| GO TO 5000                                                          | TRTD421 |
| C                                                                   | TRTD422 |
| C                                                                   | TRTD423 |
| C                                                                   | TRTD424 |
| C                                                                   | TRTD425 |
| C                                                                   | TRTD426 |
| C                                                                   | TRTD427 |
| 4300 READ(5,543) OPRAMA, ICHMH, HM, SQM                             | TRTD428 |
| 543 FORMAT(F10.2, I10, 2F10.2)                                      | TRTD429 |
| CCC TEST FOR U/S EQUIP. IF ONLY BAR-RACKS, WARN RE SPAC             | TRTD430 |
| SREFFH = (0.80 + 0.15*ICHMH)*100.0                                  | TRTD431 |
| BREFFH = (0.50 + 0.30*ICHMH)*100.0                                  | TRTD432 |
| NUNITH = 4                                                          | TRTD433 |
| QU = QDESYN/NUNITH                                                  | TRTD434 |
| IF (QU .GT. 62.0) NUNITH = QDESYN/62.0 + 1.0                        | TRTD435 |
| NEVEN = (NUNITH/2)*2                                                | TRTD436 |
| IF (NUNITH .NE. NEVEN) NUNITH = NUNITH + 1                          | TRTD437 |
| QU = QDESYN*449.0/NUNITH                                            | TRTD438 |
| UAREAH = QU/OPRAMA                                                  | TRTD439 |
| C                                                                   | TRTD440 |
| C                                                                   | TRTD441 |
| C                                                                   | TRTD442 |
| C                                                                   | TRTD443 |
| C                                                                   | TRTD444 |
| C                                                                   | TRTD445 |
| C                                                                   | TRTD446 |
| C                                                                   | TRTD447 |
| C                                                                   | TRTD448 |
| C                                                                   | TRTD449 |
| C                                                                   | TRTD450 |
| C                                                                   | TRTD451 |
| C                                                                   | TRTD452 |
| C                                                                   | TRTD453 |
| WRITE(6,643) NUNITH, UAREAH, OPRAMA, SREFFH, BREFFH, HM, SQM        | TRTD454 |
| 643 FORMAT('0', 15X, 'TREATMENT BY HIGH RATE FILTERS (LEVEL 4)', /, | TRTD455 |
| * ' ', 20X, 'NUMBER OF UNITS =', I10, /,                            | TRTD456 |
| * ' ', 20X, 'FILTER AREA PER UNIT =', F10.2, ' SQ.FT', /,           | TRTD457 |
| * ' ', 20X, 'MAX. OPERATING RATE =', F10.2,                         | TRTD458 |
| * ' GPM/SQ.FT.', /,                                                 | TRTD459 |
| * ' ', 20X, 'SOLIDS REMOVAL EFFICIENCY =', F10.2, ' PERCENT', /,    | TRTD460 |
| * ' ', 20X, 'BOD REMOVAL EFFICIENCY =', F10.2, ' PERCENT', /,       | TRTD461 |
| * ' ', 20X, 'MAX. DESIGN HEAD LOSS =', F10.2, ' FT.', /,            | TRTD462 |
| * ' ', 20X, 'MAX. SOLIDS HOLDING CAP. =', F10.2,                    | TRTD463 |
| * ' LB/SQ.FT. (AT MAX H AND Q)' )                                   | TRTD464 |
| IF (ICHMH .EQ. 0) WRITE(6,456)                                      | TRTD465 |
| IF (ICHMH .EQ. 1) WRITE(6,455)                                      | TRTD466 |
| SREFFH = SREFFH/100.0                                               | TRTD467 |
| BREFFH = BREFFH/100.0                                               | TRTD468 |
| GO TO 5000                                                          | TRTD469 |
| C                                                                   | TRTD470 |
| C-----                                                              | TRTD471 |
| C                                                                   | TRTD472 |
| 5000 K = ITREAT(5) - 50                                             | TRTD473 |
| L = 5                                                               | TRTD474 |
| GO TO (5100, 5200), K                                               | TRTD475 |
| C                                                                   | TRTD476 |
| C                                                                   | TRTD477 |
| C                                                                   | TRTD478 |
| 5100 WRITE(6,651)                                                   | TRTD479 |
| 651 FORMAT('C', 15X, 'NO EFFLUENT SCREENS (LEVEL 5)')               | TRTD480 |

|                                                                        |         |
|------------------------------------------------------------------------|---------|
| GO TO 6000                                                             | TRTD481 |
| C                                                                      | TRTD482 |
| C                                                                      | TRTD483 |
| C                                                                      | TRTD484 |
| C                                                                      | TRTD485 |
| C                                                                      | TRTD486 |
| C                                                                      | TRTD487 |
| C                                                                      | TRTD488 |
| C                                                                      | TRTD489 |
| 5200 UNESN = QDESUN/100.0                                              | TRTD490 |
| NOESUN = INT(UNESN)                                                    | TRTD491 |
| UNESNC = FLOAT(NOESUN)                                                 | TRTD492 |
| IF (UNESNO .GE. UNESN) GO TO 5210                                      | TRTD493 |
| UNESNC = UNESNO + 1.0                                                  | TRTD494 |
| NOESUN = NOESUN + 1                                                    | TRTD495 |
| 5210 IF (NOESUN .GT. 1) GO TO 5220                                     | TRTD496 |
| UNESNO = 2.0                                                           | TRTD497 |
| NOESUN = 2                                                             | TRTD498 |
| 5220 QCESUN = QDESUN*0.646/UNESNO                                      | TRTD499 |
| WRITE(6,652) QCESUN,NOESUN                                             | TRTD500 |
| 652 FORMAT('0', 15X, 'TREATMENT BY EFFLUENT SCREENS (LEVEL 5)',        | TRTD501 |
| * ' (FOR AESTHETIC IMPROVEMENTS)', /,                                  | TRTD502 |
| * ' ', 20X, 'MODULE SIZE =', F10.2,                                    | TRTD503 |
| * ' MGD, (MAX = 64.6 MGD.)', /,                                        | TRTD504 |
| * ' ', 20X, 'NO. UNITS =', I7)                                         | TRTD505 |
| C                                                                      | TRTD506 |
| C-----                                                                 | TRTD507 |
| C                                                                      | TRTD508 |
| 6000 K = ITREAT(6) - 60                                                | TRTD509 |
| L = 6                                                                  | TRTD510 |
| GO TO (6100,6200), K                                                   | TRTD511 |
| C                                                                      | TRTD512 |
| C                                                                      | TRTD513 |
| C                                                                      | TRTD514 |
| 6100 WRITE(6,661)                                                      | TRTD515 |
| 661 FORMAT('0', 15X, 'OUTFLOW BY GRAVITY (NO PUMPING) (LEVEL 6)')      | TRTD516 |
| GO TO 7000                                                             | TRTD517 |
| C                                                                      | TRTD518 |
| C                                                                      | TRTD519 |
| C                                                                      | TRTD520 |
| 6200 READ(5,522) HEAD2                                                 | TRTD521 |
| WRITE(6,662) HEAD2                                                     | TRTD522 |
| 662 FORMAT('0', 15X, 'OUTFLOW BY OUTLET PUMPING (LEVEL 6)', /,         | TRTD523 |
| * ' ', 20X, 'PUMPED HEAD =', F7.2, ' FT. WATER')                       | TRTD524 |
| C                                                                      | TRTD525 |
| C-----                                                                 | TRTD526 |
| C                                                                      | TRTD527 |
| 7000 K = ITREAT(7) - 70                                                | TRTD528 |
| L = 7                                                                  | TRTD529 |
| GO TO (7100,7200), K                                                   | TRTD530 |
| C                                                                      | TRTD531 |
| C                                                                      | TRTD532 |
| C                                                                      | TRTD533 |
| 7100 WRITE(6,671)                                                      | TRTD534 |
| 671 FORMAT('0', 15X, 'NO CHLORINE CONTACT TANK FOR OUTFLOW (LEVEL 7)') | TRTD535 |
| GO TO 8000                                                             | TRTD536 |
| C                                                                      | TRTD537 |
| C                                                                      | TRTD538 |
| C                                                                      | TRTD539 |
| 7200 CAPUCL = 2000.0                                                   | TRTD540 |



```

      IF (PCL2MX .GT. 8000.0) CAPUCL = 8000.0
      NUNITC = PCL2MX/CAPUCL + 1.0
      VOLCCN = QDESYN*60.0*15.0
      WRITE(6,672) NUNITC,CAPUCL,PCL2MX,VOLCCN
672  FORMAT('O', 15X, 'TREATMENT BY CHLORINE CONTACT TANK (LEVEL 7)',/,
*      ' ', 20X, 'NUMBER OF DOSING UNITS =', I10, '/',
*      ' ', 20X, 'DOSING RATE PER UNIT   =', F10.2, ' LB/DAY', /,
*      ' ', 20X, 'MAXIMUM DEMAND RATE    =', F10.2, ' LB/DAY', /,
*      ' ', 20X, 'VOLUME OF CONTACT TANK =', F10.0,
*      ' CU.FT, AT 15 MIN. DETENTION TIME')
C
C----- READ INITIAL CONDITIONS-----
C
8000 READ (5,570) KHCUR,KMIN
570  FORMAT (2I5)
C&&&
      GO TO 9999
C
C----- ERROR MESSAGES -----
C
9000 WRITE(6,690) L, ITREAT(L)
690  FORMAT('C*** ITREAT(', I1, ') = ', I2,
*      ' IS OF A TYPE NOT PRESENTLY MODELED.  EXECUTION STOPPED')
      STOP
C
901  IF (KFLAG .EQ. 10) GO TO 902
      WRITE(6,691)
691  FORMAT('O *** TERMINATE - INPUT TO INTERP PROCEDURE IS LESS THAN L
*QWEST VALUE ON CURVE (IN SUBRT. TRTDAT)')
      STOP
C
902  WRITE(6,692)
692  FORMAT('C *** TERMINATE - INPUT TO INTERP PROCEDURE IS GREATER THA
*N LARGEST VALUE ON CURVE (IN SUBRT. TRTDAT)')
      STOP
903  WRITE(6,693) QDSMGD, CMOD(19)
693  FORMAT('O***', 5X, 'QDESYN =', F10.2, ' MGD IS LARGER THAN MAXIMJ
* AVAILABLE COMBINATION OF MODULE SIZES =', F10.2, ' MGD.', /,
*      'O', 8X, 'MCDL IS INADEQUATE, SO TERMINATE EXECUTION.')
      STOP 693
C
C
9999 RETURN
      END
C=====

```

TRTD541  
 TRTD542  
 TRTD543  
 TRTD544  
 TRTD545  
 TRTD546  
 TRTD547  
 TRTD548  
 TRTD549  
 TRTD550  
 TRTD551  
 TRTD552  
 TRTD553  
 TRTD554  
 TRTD555  
 TRTD556  
 TRTD557  
 TRTD558  
 TRTD559  
 TRTD560  
 TRTD561  
 TRTD562  
 TRTD563  
 TRTD564  
 TRTD565  
 TRTD566  
 TRTD567  
 TRTD568  
 TRTD569  
 TRTD570  
 TRTD571  
 TRTD572  
 TRTD573  
 TRTD574  
 TRTD575  
 TRTD576  
 TRTD577  
 TRTD578  
 TRTD579  
 TRTD580  
 TRTD581  
 TRTD582  
 TRTD583  
 TRTD584  
 TRTD585

|   |                                                                       |         |
|---|-----------------------------------------------------------------------|---------|
|   | SUBROUTINE TRCHK                                                      | TRCH 1  |
|   | COMMON CONVER,KHOUR,KMIN,L,KMOD,NFLAG,BIG,HEAD1,HEAD2,                | TRCH 2  |
|   | * QUESYN,QQIF,WAOU(7),WAIN(7),QQOU(7),QQIN(7),WARM(7),QQRL,           | TRCH 3  |
|   | * BDIF,QCRM(7),BDOU(7),BDIN(7),BCOU(7),BCIN(7),BDRL,                  | TRCH 4  |
|   | * SSIF,SSIN(7),SSOU(7),SCOU(7),SCIN(7),SSRM(7),SSKL,COIF,CORL,        | TRCH 5  |
|   | * ADEPT(11),AASURF(11),ITREAT(7),ISTOR,IPRINT,ICOST,HREFD,            | TRCH 6  |
|   | * MODSIZ,ICHEM,ICL2,SCREEN,QQIFMX,DESF,IRANGE,KNTOF,TRIBA,SEDA,       | TRCH 7  |
|   | * SQM,SREFFH,BREFFH,NUNITH,UAREAH,UPRMA,ICHEMH,HM,VOLDAF,ITABLE,      | TRCH 8  |
|   | * MUDCST,TOTCST,RECIRC,OVRDAF,TSURFA,OVRSED,NSED,JM(7),WTRMT5,        | TRCH 9  |
|   | * NSCRN,SCRCAP,SUAREA,FAREAB,NMS,AREAMS,VULCON,VOLSED,ALJMT,          | TRCH 10 |
|   | * BDINT(7),SSINT(7),WAIN(7),BDOUT(7),SSOUT(7),WAOUT(7),               | TRCH 11 |
|   | * WARM(7),BDRMT(7),SSRM(7),CHEMUT(8),CL2UT(8),QMOD(20),WTRMT1,        | TRCH 12 |
|   | * QCRM(7),QQOUT(7),QCRM(7),QQUMX(7),QCRM(7),QQUMN(7),                 | TRCH 13 |
|   | * BCRM(7),BCOUT(7),BCRM(7),BCUMX(7),BCRM(7),BCUMN(7),                 | TRCH 14 |
|   | * SCRMT(7),SCOUT(7),SCRMT(7),SCUMX(7),SCRMT(7),SCUMN(7),              | TRCH 15 |
|   | * BDRM(7),BCRM(7),SCRMT(7)                                            | TRCH 16 |
|   | DIMENSION NOCOMB(15), NOCON(10)                                       | TRCH 17 |
| C |                                                                       | TRCH 18 |
| C | NOCOMB = NO. OF ILLEGAL COMBINATIONS                                  | TRCH 19 |
| C | NOCOMB = AN ILLEGAL COMBINATION PAIR                                  | TRCH 20 |
| C |                                                                       | TRCH 21 |
|   | NFLAG = 1                                                             | TRCH 22 |
|   | KNCCMB = 0                                                            | TRCH 23 |
|   | NNCCMB = 11                                                           | TRCH 24 |
|   | NOCCMB(1) = 2211                                                      | TRCH 25 |
|   | NOCCMB(2) = 3111                                                      | TRCH 26 |
|   | NOCCMB(3) = 3211                                                      | TRCH 27 |
|   | NOCCMB(4) = 3311                                                      | TRCH 28 |
|   | NOCCMB(5) = 3411                                                      | TRCH 29 |
|   | NOCCMB(6) = 4211                                                      | TRCH 30 |
|   | NOCCMB(7) = 4311                                                      | TRCH 31 |
|   | NOCCMB(8) = 7211                                                      | TRCH 32 |
|   | NOCCMB(9) = 5233                                                      | TRCH 33 |
|   | NOCCMB(10) = 5234                                                     | TRCH 34 |
|   | NOCCMB(11) = 5243                                                     | TRCH 35 |
| C |                                                                       | TRCH 36 |
|   | 1000 DO 4200 I=1,NNCOMB                                               | TRCH 37 |
|   | 2000 DO 4100 K=1,7                                                    | TRCH 38 |
|   | ITR100 = ITREAT(K)*100                                                | TRCH 39 |
|   | ITR = NOCOMB(I) - ITR100                                              | TRCH 40 |
|   | 3000 DO 4000 KK=1,7                                                   | TRCH 41 |
|   | IF (ITREAT(KK) .NE. ITR) GO TO 4000                                   | TRCH 42 |
|   | KNCCMB = KNCCMB + 1                                                   | TRCH 43 |
|   | IF (KNCCMB .EQ. 1) WRITE(6,630)                                       | TRCH 44 |
|   | 630 FORMAT('0 THE FOLLOWING COMBINATIONS OF TREATMENT OPTIONS HAVE ', | TRCH 45 |
|   | * 'BEEN SPECIFIED AS INADMISSABLE',                                   | TRCH 46 |
|   | * ' * SIMULATION WILL THEREFORE BE DISCONTINUED', /,                  | TRCH 47 |
|   | * '0 ITREAT WITH ITREAT', /)                                          | TRCH 48 |
|   | WRITE(6,631) ITREAT(K), ITR                                           | TRCH 49 |
|   | 631 FORMAT(' ', 9X, I2, 6X, 'WITH', 6X, I2)                           | TRCH 50 |
|   | 4000 CONTINUE                                                         | TRCH 51 |
|   | 4100 CONTINUE                                                         | TRCH 52 |
|   | 4200 CONTINUE                                                         | TRCH 53 |
| C |                                                                       | TRCH 54 |
|   | IF (KNCCMB .EQ. 0) GO TO 5000                                         | TRCH 55 |
|   | NFLAG = 0                                                             | TRCH 56 |
| C |                                                                       | TRCH 57 |
|   | 5000 IF (ISTOR .EQ. 2 .AND. ITREAT(3) .NE. 35) GO TO 25               | TRCH 58 |
|   | GO TO 6000                                                            | TRCH 59 |
|   | 25 WRITE(6,610)                                                       | TRCH 60 |

|      |                                                                  |         |
|------|------------------------------------------------------------------|---------|
| 610  | FORMAT('OISTOR = 02 IN THE INPUT DATA REQUIRES THAT ITREAT(3) ', | TRCH 61 |
| *    | 'BE SET TO 35 * NOT SO, SO SIMULATION IS DISCONTINUED')          | TRCH 62 |
|      | NFLAG = 0                                                        | TRCH 63 |
| C    |                                                                  | TRCH 64 |
| C    | NNECCN = NO. OF UNECONOMIC COMBINATIONS                          | TRCH 65 |
| C    | NCECON = AN UNECONOMIC COMBINATION PAIR                          | TRCH 66 |
| C    |                                                                  | TRCH 67 |
| 6000 | KNECCN = 0                                                       | TRCH 68 |
|      | NNECCN = 8                                                       | TRCH 69 |
|      | NOECCN(1) = 4233                                                 | TRCH 70 |
|      | NOECCN(2) = 4234                                                 | TRCH 71 |
|      | NOECCN(3) = 5232                                                 | TRCH 72 |
|      | NOECCN(4) = 5235                                                 | TRCH 73 |
|      | NOECCN(5) = 5242                                                 | TRCH 74 |
|      | NOECCN(6) = 7232                                                 | TRCH 75 |
|      | NOECCN(7) = 7233                                                 | TRCH 76 |
|      | NOECCN(8) = 7235                                                 | TRCH 77 |
|      | DO 8200 I=1,NNECCN                                               | TRCH 78 |
|      | DO 8100 K=1,7                                                    | TRCH 79 |
|      | ITR100 = ITREAT(K)*100                                           | TRCH 80 |
|      | ITR = NOECCN(I) - ITR100                                         | TRCH 81 |
|      | DO 8000 KK=1,7                                                   | TRCH 82 |
|      | IF (ITREAT(KK) .NE. ITR) GO TO 8000                              | TRCH 83 |
|      | KNECCN = KNECCN + 1                                              | TRCH 84 |
|      | IF (KNECCN .EQ. 1) WRITE(6,640)                                  | TRCH 85 |
| 640  | FORMAT('O**** WARNING ****', /,                                  | TRCH 86 |
| *    | 'O THE FOLLOWING COMBINATIONS OF TREATMENT ',                    | TRCH 87 |
| *    | 'PTIONS ARE CONSIDERED ECONOMICALLY INADVISABLE',                | TRCH 88 |
| *    | ' - SIMULATION CONTINUES', /,                                    | TRCH 89 |
| *    | 'O ITREAT WITH ITREAT', /)                                       | TRCH 90 |
|      | WRITE(6,631) ITREAT(K), ITR                                      | TRCH 91 |
| 8000 | CONTINUE                                                         | TRCH 92 |
| 8100 | CONTINUE                                                         | TRCH 93 |
| 8200 | CONTINUE                                                         | TRCH 94 |
| C    |                                                                  | TRCH 95 |
|      | IF (ITREAT(4) .NE. 43) GO TO 9999                                | TRCH 96 |
| C    |                                                                  | TRCH 97 |
| C    | CHECK FOR DT-SIZE WITH HIGH RATE FILTERS                         | TRCH 98 |
|      | E = 0.0001                                                       | TRCH 99 |
|      | IF (ABS(DT-0.5) .LT. E) GO TO 8500                               | TRCH100 |
|      | IF (ABS(DT-1.0) .LT. E) GO TO 8500                               | TRCH101 |
|      | IF (ABS(DT-2.0) .LT. E) GO TO 8500                               | TRCH102 |
|      | IF (ABS(DT-2.5) .LT. E) GO TO 8500                               | TRCH103 |
|      | IF (ABS(DT-5.0) .LT. E) GO TO 8500                               | TRCH104 |
|      | IF (ABS(DT-10.) .LT. E) GO TO 8500                               | TRCH105 |
|      | WRITE(6,650) DT                                                  | TRCH106 |
| 650  | FORMAT('O', 5X, 'DT =', F10.2, ' MIN. IS AN UNACCEPTABLE',       | TRCH107 |
| *    | ' VALUE WITH HIGH RATE FILTERS', /,                              | TRCH108 |
| *    | ' ', 10X, '(REQUIRED TO BE EITHER 0.5, 1.0, 2.0,',               | TRCH109 |
| *    | ' 2.5, 5.0, OR 10.0 MIN.)' )                                     | TRCH110 |
|      | NFLAG = 0                                                        | TRCH111 |
| 8500 | HRFD = 10.0/DT                                                   | TRCH112 |
| C    | ADD CHECK FOR CL2 ADDED AT ONE LOCATION ONLY                     | TRCH113 |
| C    |                                                                  | TRCH114 |
| 9999 | RETURN                                                           | TRCH115 |
|      | END                                                              | TRCH116 |
| C    | =====                                                            | TRCH117 |

|            |                                                                   |         |
|------------|-------------------------------------------------------------------|---------|
|            | SUBROUTINE STRDAT                                                 | STRD 1  |
|            | COMMON CONVER,KHOUR,KMIN,L,KMOD,NFLAG,BIG,HEAD1,HEAD2,            | STRD 2  |
|            | * QDESYN,CCIF,WAQU(7),WAIN(7),QQOU(7),QQIN(7),WARM(7),QQRL,       | STRD 3  |
|            | * BDIF,CCRM(7),BDCU(7),BDIN(7),BCOU(7),BCIN(7),BDRL,              | STRD 4  |
|            | * SSIF,SSIN(7),SSOU(7),SCOU(7),SCIN(7),SSRM(7),SSRL,COIF,CORL,    | STRD 5  |
|            | * ADEPTH(11),AASURF(11),ITREAT(7),ISTOR,IPRINT,ICUST,HFFD,        | STRD 6  |
|            | * MODSI2,ICHEM,ICL2,SCREEN,QQIFMX,DESF,IRANGE,KNTOF,TRIBA,SEDA,   | STRD 7  |
|            | * SQM,SREFFH,BREFFH,NUNITH,UAREAH,OPRAMA,ICHEMH,HM,VOLDAF,ITABLE, | STRD 8  |
|            | * MUDCST,TCTCST,RECIRC,UVRDAF,TSURFA,UVRSED,NSED,JM(7),WTRMT5,    | STRD 9  |
|            | * NSCRN,SCRCAP,SUAREA,FAREAB,NMS,AREAMS,VOLCON,VOLSED,ALUMUT,     | STRD 10 |
|            | * BDMT(7),SSINT(7),WAIMT(7),BDOUT(7),SSOUT(7),WAOUT(7),           | STRD 11 |
|            | * WARM(7),BCRMT(7),SSRMT(7),CHEMUT(8),CL2UT(8),QMOD(20),WTRMT1,   | STRD 12 |
|            | * QQRMT(7),QQOUT(7),QQRMMX(7),QQOUMX(7),QQRMMN(7),QQOUMV(7),      | STRD 13 |
|            | * BCRMT(7),BCOU(7),BCRMMX(7),BCOUMX(7),BCRMMN(7),BCOUMV(7),       | STRD 14 |
|            | * SCRMT(7),SCOUT(7),SCRMMX(7),SCOUMX(7),SCRMMN(7),SCOUMV(7),      | STRD 15 |
|            | * BDRM(7),BCRM(7),SCRM(7)                                         | STRD 16 |
|            | COMMON /STBK/QIN(150),BODIN(150),SUSIN(150),COLIN(150),           | STRD 17 |
|            | * QINST,CCUST,CINSTL,QOUSTL,STORL,QOUTO,STORO,                    | STRD 18 |
|            | * ISPRIN,IPCL,DEPMAX,QOMAX,DEPTH,                                 | STRD 19 |
|            | * ATERM(11),AOZDT2(11),BDEPTH(11),BSTOR(11),                      | STRD 20 |
|            | * DUMSTR(11),DUMDEP(11),                                          | STRD 21 |
|            | * VOLIN(150),VOLOUT(150),STOR,CUMIN,CUMOUT,                       | STRD 22 |
|            | * SBOD,SSS,SCCL,BODOUT,SUSOUT,COLOUT,                             | STRD 23 |
|            | * ISTMOD,ISTTYP,ISTOUT,                                           | STRD 24 |
|            | * QPUMP,CSTART,CSTOP,                                             | STRD 25 |
|            | * DTON,STORMX,DTPUMP,DTMORE,STORF,APLAN,                          | STRD 26 |
|            | * CLANC,CSTOR,CPS,CTOTAL,CPCUYD,CPCACRE,                          | STRD 27 |
|            | * LP,JP,LPREV,LABEL,DETENT(150),FRAC(150)                         | STRD 28 |
|            | COMMON /TBLK/DT,NOT,KDT                                           | STRD 29 |
|            | DIMENSION NUE(3)                                                  | STRD 30 |
|            | DT=DT*60.C                                                        | STRD 31 |
| C          |                                                                   | STRD 32 |
|            | READ(5,501) ISTMOD, ISTTYP, ISTOUT                                | STRD 33 |
|            | 501 FORMAT(10I5)                                                  | STRD 34 |
| C          |                                                                   | STRD 35 |
| C          | NOTE.. OPERATIONAL OPTIONS INDICATED BY '*'                       | STRD 36 |
| C          |                                                                   | STRD 37 |
| C          | ISTMOD = STORAGE MODE (ROUTING, HOLDING, ETC)                     | STRD 38 |
| C          | * = 1 = IN-LINE                                                   | STRD 39 |
| C          | = 2 = OFF-LINE                                                    | STRD 40 |
| C          | = 3 = INTRASYSTEM                                                 | STRD 41 |
| C          | = 4 = REROUTING                                                   | STRD 42 |
| C          | ISTTYP = STORAGE STRUCTURE (NATURAL, TANK, BAG)                   | STRD 43 |
| C          | * = 1 = IRREGULAR (NATURAL) RESERVOIR                             | STRD 44 |
| C          | * = 2 = GEOMETRIC (REGULAR) RESERVOIR - COVERED                   | STRD 45 |
| C          | * = 3 = GEOMETRIC (REGULAR) RESERVOIR - UNCOVERED                 | STRD 46 |
| C          | = 4 = INTRASYSTEM                                                 | STRD 47 |
| C          | = 5 = RUBBER BAG                                                  | STRD 48 |
| C-EXCLUDED | ISTEXS = EXCESS FLOW HANDLING (BYPASS, BACK-UP, FLOOD)            | STRD 49 |
| C-EXCLUDED | = 1 = BYPASS FRACTION CONTINUOUSLY, UNDER GRASTOR                 | STRD 50 |
| C-EXCLUDED | = 2 = BYPASS FRACTION CONTINUOUSLY, CONTROLLED                    | STRD 51 |
| C-EXCLUDED | = 3 = BYPASS ALL, AFTER SURCHARGE BEGINS (CONSTOR                 | STRD 52 |
| C-EXCLUDED | = 4 = BACK UP                                                     | STRD 53 |
| C-EXCLUDED | = 5 = PASS THROUGH (UNDER HEAD)                                   | STRD 54 |
| C-EXCLUDED | * = 6 = FLOODS (AND SIMULATION TERMINATES)                        | STRD 55 |
| C          | ISTOUT = OUTLET TYPE (GRAVITY, PUMP)                              | STRD 56 |
| C          | * = 1 = GRAVITY WITH FIXED ORIFICE                                | STRD 57 |
| C          | * = 2 = GRAVITY WITH FIXED WEIR                                   | STRD 58 |
| C          | = 3 = GRAVITY WITH FIXED SIDE-WEIR                                | STRD 59 |
| C          | = 4 = GRAVITY WITH FIXED SIPHON                                   | STRD 60 |

```

C          = 5 = NEW PUMPS STRD 61
C          *          = 6 = EXISTING PUMPS STRD 62
C          = 7 = ADJUSTABLE VALVE WITH GRAVITY STRD 63
C          = 8 = SLUICE GATE WITH GRAVITY STRD 64
C-EXCLUDED ISTBUP = BACK-UP EFFECTS (YES/NO) STRD 65
C-EXCLUDED = 1 = NO STRD 66
C-EXCLUDED = 2 = YES, UPSTREAM OF INLET POINT STRD 67
C-EXCLUDED = 3 = YES, UPSTREAM OF OUTLET POINT STRD 68
C STRD 69
C WRITE(6,602) ISTOUT, ISTMOD, ISTTYP STRD 70
602 FORMAT(' ', 20X, 'CHARACTERISTICS OF STORAGE UNIT ARE', STRD 71
* /, ' ', 25X, 'OUTLET TYPE =', I3, STRD 72
* /, ' ', 25X, 'STORAGE MODE =', I3, STRD 73
* /, ' ', 25X, 'STORAGE TYPE =', I3) STRD 74
C STRD 75
C IPOL = 1 = SOLIDS PRESENT (100% EFFICIENT PLUG FLOW) STRD 76
C = 2 = SOLIDS PRESENT (100% EFFICIENT MIXING) STRD 77
C IPRINT = 0 = NO PRINTOUT EACH TIME-STEP (SUMMARY POSSTRD 78
C = 1 = PRINTOUT ALL TIME-STEP SOLUTIONS (EACH STRD 79
C ICOST = 0 = NO COST SUMMARY STRD 80
C ICOST = 1 = COST SUMMARY INCLUDED STRD 81
C STRD 82
C READ(5,502) IPOL, ISPRIN STRD 83
502 FORMAT(2I10) STRD 84
C WRITE(6,603) IPOL, ISPRIN STRD 85
603 FORMAT('C', 20X, 'IPOL =', I2, ' ', PRINT CONTROL (ISPRIN) =', I2) STRD 86
C STRD 87
C----- READ RESERVOIR PARAMETERS STRD 88
C STRD 89
C STRD 90
C BRANCH TO STORAGE TYPE (ISTTYP) STRD 91
C STRD 92
1000 IF (ISTTYP .EQ. 1) GO TO 1100 STRD 93
IF (ISTTYP .EQ. 2) GO TO 1200 STRD 94
IF (ISTTYP .EQ. 3) GO TO 1200 STRD 95
CC IF (ISTTYP .EQ. 4) GO TO STRD 96
CC IF (ISTTYP .EQ. 5) GO TO STRD 97
GO TO 904 STRD 98
C STRD 99
C A) FOR IRREGULAR (NATURAL) RESERVOIR STRD100
C STRD101
1100 READ(5,511) DEPMAX, ICL2 STRD102
511 FORMAT(F10.2, I10) STRD103
C DEPMAX = MAX. ALLOWABLE DEPTH IN RESERVOIR (FT.) STRD104
C WRITE(6,611) DEPMAX STRD105
611 FORMAT('C', 20X, 'NATURAL RESERVOIR, WITH MAX. DEPTH =', F7.2, STRD106
* ' FT.', 10X, ' 11 DEPTH/AREA PARAMETERS ARE', /, STRD107
* ' ', 25X, 4('DEPTH(FT) AREA(SQ.FT)', 4X)) STRD108
C READ(5,512) (ADEPTH(I), AASURF(I), I=1,11) STRD109
512 FORMAT(4(F10.2, F10.0)) STRD110
C ADEPTH = DATA DEPTH (FT), 0 - DEPMAX. STRD111
C AASURF = DATA SURF. AREA OF NAT. RESERVOIR (SQ. FT.) STRD112
C WRITE(6,612) (ADEPTH(I), AASURF(I), I=1,11) STRD113
612 FORMAT(' ', 23X, F10.2, F10.0, 5X, F10.2, F10.0, 5X, STRD114
* F10.2, F10.0, 5X, F10.2, F10.0) STRD115
C IF (ADEPTH(11) .LT. DEPMAX) GO TO 903 STRD116
C STRD117
C CALL INTERP(ADEPTH,AASURF,11,DEPMAX,APLAN,KFLAG) STRD118
C STRD119
C IF (KFLAG .NE. 0) GO TO 901 STRD120

```

|                                                                               |         |
|-------------------------------------------------------------------------------|---------|
| APLAN = APLAN*1.25                                                            | STRD121 |
| GO TO 2000                                                                    | STRD122 |
| C                                                                             | STRD123 |
| C                                                                             | STRD124 |
| C                                                                             | STRD125 |
| 1200 READ(5,511) DEPMAX, ICL2                                                 | STRD126 |
| READ(5,513) BASEA, BASEC, COTSLO                                              | STRD127 |
| 513 FORMAT(2F10.0, F10.5)                                                     | STRD128 |
| C                                                                             | STRD129 |
| C                                                                             | STRD130 |
| C                                                                             | STRD131 |
| BASEA = BASE AREA OF RESERVOIR (SQ.FT.)                                       | STRD132 |
| BASEC = BASE CIRCUMF. OF RESERVOIR (FT.)                                      | STRD133 |
| COTSLO = COTAN OF SIDESLOPE ANGLE (HORIZ/VERT)                                | STRD134 |
| ADEPTH(1)=0.0                                                                 | STRD135 |
| AASURF(1)=BASEA                                                               | STRD136 |
| DO 1 I=1,2,11                                                                 | STRD137 |
| ADEPTH(MMM)=ADEPTH(MMM-1)+DEPMAX/10.0                                         | STRD138 |
| AASURF(MMM)=BASEA                                                             | STRD139 |
| 1 CONTINUE                                                                    | STRD140 |
| APLAN = (BASEA + BASEC*COTSLO*DEPMAX)*1.25                                    | STRD141 |
| WRITE(6,613) DEPMAX, BASEA, BASEC, COTSLO                                     | STRD142 |
| 613 FORMAT('0', 15X, 'MAN-MADE RESERVOIR, WITH MAX. DEPTH =', F7.2,           | STRD143 |
| * ' FT., AND CHARACTERISTICS', /, ' ', 25X,                                   | STRD144 |
| * 'BASE AREA =', F10.0, ' SQ.FT.', /, 7X,                                     | STRD145 |
| * 'BASE CIRCUMF. =', F10.0, ' FT.', /, 6X,                                    | STRD146 |
| * 'COT(SIDESLOPE) =', F10.5)                                                  | STRD147 |
| C                                                                             | STRD148 |
| C-----                                                                        | STRD149 |
| C                                                                             | STRD150 |
| C                                                                             | STRD151 |
| C                                                                             | STRD152 |
| C                                                                             | STRD153 |
| 2000 IF (ISTCUT .EQ. 1) GO TO 2100                                            | STRD154 |
| IF (ISTOUT .EQ. 2) GO TO 2200                                                 | STRD155 |
| CC IF (ISTCLT .EQ. 3) GO TO                                                   | STRD156 |
| CC IF (ISTCLT .EQ. 4) GO TO                                                   | STRD157 |
| CC IF (ISTCLT .EQ. 5) GO TO                                                   | STRD158 |
| IF (ISTOUT .EQ. 6) GO TO 2600                                                 | STRD159 |
| CC IF (ISTCLT .EQ. 7) GO TO                                                   | STRD160 |
| CC IF (ISTOUT .EQ. 8) GO TO                                                   | STRD161 |
| GO TO 906                                                                     | STRD162 |
| C                                                                             | STRD163 |
| C                                                                             | STRD164 |
| C                                                                             | STRD165 |
| 2100 READ(5,521) CDAOUT                                                       | STRD166 |
| 521 FORMAT(F10.3)                                                             | STRD167 |
| C                                                                             | STRD168 |
| CDAOUT = OUTLET ORIFICE AREA * DISCHARGE CJEFFICIENT                          | STRD169 |
| WRITE(6,621) CDAOUT, DT                                                       | STRD170 |
| 621 FORMAT(' ', 20X, 'RESERVOIR OUTLET CONTROL BY GRAVITY WITH FIXED OSTRD166 | STRD171 |
| *RIFICE', /, ' ', 20X, 'ORIFICE AREA*CD =', F10.3, ' SQ.FT, DT =',            | STRD172 |
| *F10.2, ' SEC. (FROM INPUT HYDROGRAPH)', /, ' ', 20X, 'NOTE..', /,            | STRD173 |
| *' ', 23X, 'ORIFICE CENTERLINE ASSUMED AT TANK DEPTH = 0, WHEN STORSTRD169    | STRD174 |
| *AGE = 0')                                                                    | STRD175 |
| QDMAX = CDAOUT*SQRT(64.4*DEPMAX)                                              | STRD176 |
| C                                                                             | STRD177 |
| GO TO 3000                                                                    | STRD178 |
| C                                                                             | STRD179 |
| C                                                                             | STRD180 |
| C                                                                             | STRD181 |
| C                                                                             | STRD182 |
| 2200 READ(5,522) WEIRHT, WEIRL                                                | STRD183 |
| 522 FORMAT(2F10.3)                                                            | STRD184 |
| C                                                                             | STRD185 |
| C                                                                             | STRD186 |
| WEIRHT = RES. DEPTH (FT) WHEN SURF. AT WEIR ELEV.                             | STRD187 |
| WEIRL = WEIR LENGTH (FT)                                                      | STRD188 |

```

C          WEIRQ = 3.33*(DEPTH-WEIRHT)**1.5 (CFS/FT) STRD181
C          WRITE(6,622) WEIRHT, WEIRL, DT STRD182
622 FORMAT(' ', 20X, 'RESERVOIR OUTLET CONTROL BY GRAVITY WITH FIXED WSTRD183
*EIR', /, ' ', 20X, 'WEIR HEIGHT =', F6.2, ' FT, WEIR LENGTH =', STRD184
* F7.2, ' FT, DT =', F10.2, ' SEC. (FROM INPUT HYDROGRAPH)') STRD185
C          QUMAX = WEIRL*3.33*(DEPMAX-WEIRHT)**1.5 STRD186
C STRD187
C          GO TO 3000 STRD188
C STRD189
C          F) OUTLET BY EXISTING PUMPS STRD190
C STRD191
6200 READ(5,526) QPUMP,DSTART,DSTOP STRD192
526 FORMAT(3F10.3) STRD193
C          QPUMP = CONSTANT PUMPED OUTFLOW RATE (CFS) STRD194
C          DSTART = RESERVOIR DEPTH AT START OF PUMPING (FT) STRD195
C          DSTOP = RESERVOIR DEPTH AT END OF PUMPING (FT) STRD196
C          WRITE(6,626) QPUMP,DSTART,DSTOP STRD197
626 FORMAT(' ', 20X, 'RESERVOIR OUTFLOW BY FIXED-RATE PUMPING', /, STRD198
* ' ', 20X, 'PUMPING RATE =', F7.2, ' CFS, PUMPING START', STRD199
* ' DEPTH =', F6.2, ' FT, PUMPING STOP DEPTH =', F6.2, ' FT') STRD200
C          QUMAX = QPUMP STRD201
C          IF (DSTART .GT. DSTOP) GO TO 3000 STRD202
C STRD203
C          DSTOP = DSTART - 0.5 STRD204
C          WRITE(6,627) STRD205
627 FORMAT('***** WARNING. STOP DEPTH SHOULD BE LESS THAN START', STRD206
* ' DEPTH', /, ' ', 10X, 'RESET STOP DEPTH TO', F6.2, ' FT') STRD207
C          GO TO 3000 STRD208
C STRD209
C          END OF INPUT PARAMETERS FOR STORAGE UNITS STRD210
C STRD211
C STRD212
C----- COMPUTE DEPTH/STORAGE RELATION (ARRAY) ----- STRD213
C STRD214
C          NOTE STRD215
C          RUBBER BAG AND INTRASYSTEM DO NOT HAVE 'DEPTH' STRD216
C STRD217
C          BRANCH TO STORAGE TYPE (ISTTYP) STRD218
C STRD219
3000 IF (ISTTYP .EQ. 1) GO TO 3100 STRD220
C          IF (ISTTYP .EQ. 2) GO TO 3200 STRD221
C          IF (ISTTYP .EQ. 3) GO TO 3200 STRD222
CC          IF (ISTTYP .EQ. 4) GO TO STRD223
CC          IF (ISTTYP .EQ. 5) GO TO STRD224
C          GO TO 904 STRD225
C STRD226
C----- A) FOR 'NATURAL' RESERVOIR STRD227
C STRD228
3100 DDEPTH = DEPMAX/10.0 STRD229
C          IF (ISTCUT .EQ. 2 .OR. ISTOUT .EQ. 3) STRD230
*          DDEPTH = WEIRHT/3.0 STRD231
C          DUMCEP(1) = 0.0 STRD232
C          DEPTH = 0.0 STRD233
C STRD234
C          CALL INTERP(ADEPTH, AASURF, 11, DEPTH, AREA, KFLAG) STRD235
C STRD236
C          IF (KFLAG .NE. 0) GO TO 901 STRD237
C          AREA2 = AREA STRD238
C          DUMSTR(1) = 0.0 STRD239
C          DO 3150 I=2,11 STRD240

```

|                                                                 |         |
|-----------------------------------------------------------------|---------|
| AREA1 = AREA2                                                   | STRD241 |
| IF ((ISTOUT .EQ. 2 .OR. ISTOUT .EQ. 3) .AND.                    | STRD242 |
| * (I .EQ. 5)) DDEPTH = (DEPMAX - WEIRHT)/7.0                    | STRD243 |
| DUMDEP(I) = DUMDEP(I-1) + DDEPTH                                | STRD244 |
| DEPTH = DUMDEP(I)                                               | STRD245 |
| C                                                               | STRD246 |
| CALL INTERP(ACEPTH, AASURF, 11, DEPTH, AREA, KFLAG)             | STRD247 |
| C                                                               | STRD248 |
| IF (KFLAG .NE. 0) GO TO 901                                     | STRD249 |
| AREA2 = AREA                                                    | STRD250 |
| 3150 DUMSTR(I) = DUMSTR(I-1) + 0.5*DDEPTH*(AREA1+AREA2)         | STRD251 |
| WRITE(6,631)                                                    | STRD252 |
| 631 FORMAT(' ', 20X, 4('DEPTH(FT) STOR(CU.FT)', 4X))            | STRD253 |
| WRITE(6,632) (DUMDEP(I), DUMSTR(I), I=1,11)                     | STRD254 |
| 632 FORMAT(' ', 18X, F9.2, F11.0, 5X, F9.2, F11.0, 5X,          | STRD255 |
| * F9.2, F11.0, 5X, F9.2, F11.0)                                 | STRD256 |
| GO TO 3300                                                      | STRD257 |
| C                                                               | STRD258 |
| C----- B) FOR MAN-MADE RESERVOIR                                | STRD259 |
| C                                                               | STRD260 |
| 3200 DDEPTH = DEPMAX/10.0                                       | STRD261 |
| IF (ISTCUT .EQ. 2 .OR. ISTOUT .EQ. 3)                           | STRD262 |
| * DDEPTH = WEIRHT/3.0                                           | STRD263 |
| DUMDEP(I) = 0.0                                                 | STRD264 |
| DEPTH = 0.0                                                     | STRD265 |
| DUMSTR(I) = 0.0                                                 | STRD266 |
| DO 3250 I=2,11                                                  | STRD267 |
| IF ((ISTCUT .EQ. 2 .OR. ISTOUT .EQ. 3) .AND.                    | STRD268 |
| * (I .EQ. 5)) DDEPTH = (DEPMAX - WEIRHT)/7.0                    | STRD269 |
| DUMDEP(I) = DUMDEP(I-1) + DDEPTH                                | STRD270 |
| 3250 DUMSTR(I) = (BASEA + 0.5*BASEC*DUMDEP(I)*COTSLO)*DUMDEP(I) | STRD271 |
| WRITE(6,631)                                                    | STRD272 |
| WRITE(6,632) (DUMDEP(I), DUMSTR(I), I=1,11)                     | STRD273 |
| C                                                               | STRD274 |
| 3300 DO 3350 I=1,11                                             | STRD275 |
| BSTOR(I) = DUMSTR(I)                                            | STRD276 |
| 3350 BDEPTH(I) = DUMDEP(I)                                      | STRD277 |
| C                                                               | STRD278 |
| C CALL INTERP(BDEPTH,BSTOR,11,DEPMAX,STORMX,KFLAG)              | STRD279 |
| STORMX=BSTOR(11)                                                | STRD280 |
| C                                                               | STRD281 |
| C IF (KFLAG .NE. 0) GO TO 901                                   | STRD282 |
| C                                                               | STRD283 |
| C END OF DEPTH/STORAGE COMPUTATIONS                             | STRD284 |
| C                                                               | STRD285 |
| C----- BRANCH TO STORAGE MODE (ISTMOD) -----                    | STRD286 |
| C                                                               | STRD287 |
| IF (ISTMCC .EQ. 1) GO TO 4000                                   | STRD288 |
| CC IF (ISTMCD .EQ. 2) GO TO                                     | STRD289 |
| CC IF (ISTMOD .EQ. 3) GO TO                                     | STRD290 |
| CC IF (ISTMCC .EQ. 4) GO TO                                     | STRD291 |
| GO TO SC5                                                       | STRD292 |
| C                                                               | STRD293 |
| C----- COMPUTE AND PRINT ROUTING PARAMETERS -----               | STRD294 |
| C                                                               | STRD295 |
| C BRANCH TO OUTLET TYPE (ISTOUT)                                | STRD296 |
| C                                                               | STRD297 |
| 4000 IF (ISTCUT .EQ. 1) GO TO 4100                              | STRD298 |
| IF (ISTCUT .EQ. 2) GO TO 4200                                   | STRD299 |
| CC IF (ISTCUT .EQ. 3) GO TO                                     | STRD300 |



|         |                                                                    |         |
|---------|--------------------------------------------------------------------|---------|
| CC      | IF (ISTCUT .EQ. 4) GO TO                                           | STRD301 |
|         | IF (ISTCUT .EQ. 5) GO TO 8000                                      | STRD302 |
|         | IF (ISTCUT .EQ. 6) GO TO 4600                                      | STRD303 |
| CC      | IF (ISTCUT .EQ. 7) GO TO                                           | STRD304 |
| CC      | IF (ISTCUT .EQ. 8) GO TO                                           | STRD305 |
|         | GO TO 906                                                          | STRD306 |
| C       |                                                                    | STRD307 |
| C-----  | CCMPUTE AND PRINT ROUTING PARAMETERS                               | STRD308 |
| C       |                                                                    | STRD309 |
| C-----  | A) FOR ORIFICE OUTLET                                              | STRD310 |
| C       |                                                                    | STRD311 |
| 4100    | DT2 = 0.5*DT                                                       | STRD312 |
| DO 4150 | I=1,11                                                             | STRD313 |
|         | DEPTH = DUMDEP(I)                                                  | STRD314 |
|         | STOR = DUMSTR(I)                                                   | STRD315 |
|         | QOUT = CDAOUT*SQRT(64.4*DEPTH)                                     | STRD316 |
| C       |                                                                    | STRD317 |
|         | AO2DT2(I) = QOUT*DT2                                               | STRD318 |
| 4150    | ATERM(I) = QOUT*DT2 + STOR                                         | STRD319 |
|         | WRITE(6,641) DT, (ATERM(I), I=1,11),                               | STRD320 |
|         | *, (AO2DT2(I), I=1,11)                                             | STRD321 |
| 641     | FORMAT(' ', 20X, 'THE TWO SETS OF 11 STORAGE PARAMETERS, FOR DT =' | STRD322 |
| *       | , F10.3, ' SEC., ARE..', /, ' ', 11X, 'ATERM.. ',                  | STRD323 |
| *       | 11F10.0, /, ' ', 11X, 'AO2DT2..', 11F10.0)                         | STRD324 |
|         | GO TO 8000                                                         | STRD325 |
| C       |                                                                    | STRD326 |
| C-----  | B) FOR WEIR OUTLET                                                 | STRD327 |
| C       |                                                                    | STRD328 |
| 4200    | DT2 = 0.5*DT                                                       | STRD329 |
| DO 4250 | I=1,11                                                             | STRD330 |
|         | DEPTH = DUMDEP(I)                                                  | STRD331 |
|         | STOR = DUMSTR(I)                                                   | STRD332 |
|         | H = DEPTH - WEIRHT                                                 | STRD333 |
|         | QOUT = 0.0                                                         | STRD334 |
|         | IF (H .GT. 0.0) QOUT = WEIRL*3.33*H*SQRT(H)                        | STRD335 |
| C       |                                                                    | STRD336 |
|         | AO2DT2(I) = QOUT*DT2                                               | STRD337 |
| 4250    | ATERM(I) = QOUT*DT2 + STOR                                         | STRD338 |
|         | WRITE(6,641) DT, (ATERM(I), I=1,11),                               | STRD339 |
| *       | *, (AO2DT2(I), I=1,11)                                             | STRD340 |
|         | GO TO 8000                                                         | STRD341 |
| C       |                                                                    | STRD342 |
| C-----  | CHECK BUFFER VOLUME FOR PUMPED OUTFLOW                             | STRD343 |
| C       |                                                                    | STRD344 |
| 4600    | DSTRT = DSTART                                                     | STRD345 |
|         | CALL INTERP(DUMDEP, DUMSTR, 11, DSTRT, STORHI, KFLAG)              | STRD346 |
| C       |                                                                    | STRD347 |
|         | IF (KFLAG .NE. 0) GO TO 901                                        | STRD348 |
| C       |                                                                    | STRD349 |
|         | DSTP = DSTCP                                                       | STRD350 |
|         | CALL INTERP(DUMDEP, DUMSTR, 11, DSTP, STORLO, KFLAG)               | STRD351 |
| C       |                                                                    | STRD352 |
|         | IF (KFLAG .NE. 0) GO TO 901                                        | STRD353 |
|         | STORDV = STORHI - STORLO                                           | STRD354 |
|         | PUMPCV = QPUMP*DT                                                  | STRD355 |
|         | DVR = STORDV/PUMPCV                                                | STRD356 |
|         | IF (DVR .LT. 1.0) GO TO 4650                                       | STRD357 |
| C       |                                                                    | STRD358 |
|         | WRITE(6,646) DVR                                                   | STRD359 |
| 646     | FORMAT(' ', 20X, 'STORAGE BETWEEN PUMP START AND STOP LEVELS =',   | STRD360 |

```

*      F12.2, ' TIMES (QPUMP*DT)')
GO TO 8000
C
4650 WRITE(6,647) STORH1, STORL1, STORDV, PUMPDV
647 FORMAT('0',20X, 'AT LEVEL DSTART, STORAGE      =', F10.0, ' CU.FT',/,STRD365
*      ' ',20X, 'AT LEVEL DSTOP, STORAGE      =', F10.0, ' CU.FT',/,STRD366
*      ' ',20X, 'DIFFERENCE = BUFFER STORAGE =', F10.0, ' CU.FT',/,STRD367
*      '0',16X, 'CF. VOLUME PUMPED / TIME-STEP      =', F10.0, ' CU.FT',/,STRD368
*      '0',10X, 'A RELIABLE MODEL REQUIRES THE VOLUME PUMPED / TIME-',STRD369
*      'STEP TC BE LESS THAN THE BUFFER STORAGE', /, STRD370
*      ' ',10X, 'THEREFORE ONE OF THE FOLLOWING AMENDMENTS SHOULD ', STRD371
*      'PROBABLY BE MADE TO THE INPUT DATA -', /, STRD372
*      ' ',15X, 'A) REDUCE QPUMP RATE', /, STRD373
*      ' ',15X, 'B) REDUCE DSTOP LEVEL', /, STRD374
*      ' ',15X, 'C) INCREASE DSTART LEVEL', /, STRD375
*      ' ',15X, 'D) INCREASE RESERVOIR PLAN AREAS', /, STRD376
*      '0', 2X, '***** FOR THE ABOVE REASONS, THE FOLLOWING OUTPUT',STRD377
*      ' IS NOT NECESSARILY RELIABLE') STRD378
GO TO 8000
C
C----- READ RESERVOIR INITIAL CONDITIONS -----STRD381
C
C      STORO = STORAGE (CU.FT.) AT T = 0
C      QCUTO = OUTFLOW RATE (CFS) AT T = 0
C
8000 READ(5,551) STORO, QCUTO
551 FORMAT(2F10.2)
C
C----- READ STORAGE UNIT UNIT COSTS -----STRD388
C
C      READ(5,561) CPCUYD
561 FORMAT(F10.2)
C      WRITE(6,661) CPCUYD
661 FORMAT(' ', 20X, 'ASSUMED UNIT COST (EXCAVATION, LINING, ETC.) =',STRD393
*      F5.2, ' $/CU.YD.')
C
8888 CONTINUE
C
GO TO 9999
C
C----- ERROR MESSAGES -----STRD400
C
901 IF (KFLAG .EQ. 10) GO TO 902
WRITE(6,691)
691 FORMAT('0 *** TERMINATE - INPUT TO INTERP PROCEDURE IS LESS THAN LSTRD404
*LOWEST VALUE ON CURVE (IN SUBRT. STRDAT)')
STOP
C
902 WRITE(6,692)
692 FOKMAT('0 *** TERMINATE - INPUT TO INTERP PROCEDURE IS GREATER THASTRD409
*N LARGEST VALUE ON CURVE (IN SUBRT. STRDAT)')
STOP
C
903 WRITE(6,693) DEPMAX, ADEPTH(11)
693 FORMAT('0THE MAX. DEPTH, DEPMAX =',
* F7.2, ' FT, IS OUTSIDE DEPTH PARAMETER RANGE. LARGEST DEPTH PARASTRD415
*METER, ADEPTH(11) =', F7.2)
STOP
C
904 WRITE(6,694) ISTTYP
694 FORMAT('0 *** ISTTYP =', I3,

```

|                                           |         |
|-------------------------------------------|---------|
| * ', IS OF A TYPE NOT PRESENTLY MODELED') | STRD421 |
| STOP                                      | STRD422 |
| C                                         | STRD423 |
| 905 WRITE(6,695) ISTMOD                   | STRD424 |
| 695 FORMAT('O *** ISTMOD =', I3,          | STRD425 |
| * ', IS OF A TYPE NOT PRESENTLY MODELED') | STRD426 |
| STOP                                      | STRD427 |
| C                                         | STRD428 |
| 906 WRITE(6,696) ISTOUT                   | STRD429 |
| 696 FORMAT('O *** ISTOUT =', I3,          | STRD430 |
| * ', IS OF A TYPE NOT PRESENTLY MODELED') | STRD431 |
| STOP                                      | STRD432 |
| C                                         | STRD433 |
| 907 WRITE(6,697) ISTINF                   | STRD434 |
| 697 FORMAT('O *** ISTINF =', I3,          | STRD435 |
| * ', IS OF A TYPE NOT PRESENTLY MODELED') | STRD436 |
| STOP                                      | STRD437 |
| C                                         | STRD438 |
| C                                         | STRD439 |
| 9999 DT=DT/60.0                           | STRD440 |
| RETURN                                    | STRD441 |
| END                                       | STRD442 |
| C=====                                    | STRD443 |

|   |                                                                   |         |
|---|-------------------------------------------------------------------|---------|
|   | SUBROUTINE TREAT                                                  | TREA 1  |
|   | COMMON CCNVER,KHOUR,KPIN,L,KMCD,NFLAG,BIG,HEAD1,HEAD2,            | TREA 2  |
|   | * QDESYN,CQIF,WAOU(7),WAIN(7),QQOU(7),QQIN(7),WARM(7),QQR,        | TREA 3  |
|   | * BDIF,CGRM(7),BDOU(7),BDIN(7),BCOU(7),BCIN(7),BDRL,              | TREA 4  |
|   | * SSIF,SSIN(7),SSOU(7),SCOU(7),SCIN(7),SSKM(7),SSRL,COIF,CORL,    | TREA 5  |
|   | * ADEPTH(11),AASURF(11),ITREAT(7),ISTOR,IPRINT,ICOST,HREFD,       | TREA 6  |
|   | * MODS(2),ICHEM,ICL2,SCREEN,QQIFMX,DESF,IRANGE,KNTOF,TRIBA,SEDA,  | TREA 7  |
|   | * SQM,SREFFH,BREFFH,NUNITH,UAREAH,UPRAMA,ICHEMH,HM,VULDAF,ITABLE, | TREA 8  |
|   | * MODCST,TOTCST,RECIRC,OVRDAF,TSURFA,OVRSED,NSED,JM(7),WTRMT5,    | TREA 9  |
|   | * NSCRN,SCRCAP,SUAREA,FARFAB,NMS,AREAMS,VOLCON,VOLSED,ALUMUT,     | TREA 10 |
|   | * BDINT(7),SSINT(7),WAIN(7),BDOU(7),SSOU(7),WAOUT(7),             | TREA 11 |
|   | * WARM(7),BDRMT(7),SSRMT(7),CHEMUT(8),CLZUT(8),QMOD(20),WTRMT1,   | TREA 12 |
|   | * QQRMT(7),QQOUT(7),QCRMX(7),QQUMX(7),QCRMN(7),QQUMN(7),          | TREA 13 |
|   | * BCRMT(7),BCOUT(7),BCRMX(7),BCUMX(7),BCRMN(7),BCUMN(7),          | TREA 14 |
|   | * SCRMT(7),SCOUT(7),SCRMX(7),SCUMX(7),SCRMN(7),SCUMN(7),          | TREA 15 |
|   | * BDRM(7),BCRM(7),SCRM(7)                                         | TREA 16 |
|   | COMMON /TBLK/DT,NDT,KDT                                           | TREA 17 |
|   | COMMON /BLK1/NAME(4,21)                                           | TREA 18 |
|   | COMMON /STBK/QIN(150),BODIN(150),SUSIN(150),COLIN(150),           | TREA 19 |
|   | * QINST,CQUST,QINSTL,CQUSTL,STORL,QOUTO,STORO,                    | TREA 20 |
|   | * ISPRIN,IPOL,DEPMAX,CQMAX,DEPTH,                                 | TREA 21 |
|   | * ATERM(11),AU2DT2(11),BDEPTH(11),BSTOR(11),                      | TREA 22 |
|   | * DUMSTR(11),DUMDEP(11),                                          | TREA 23 |
|   | * VOLIN(150),VOLOUT(150),STOR,CUMIN,CUMOUT,                       | TREA 24 |
|   | * SBOD,SSS,SCOL,BODOUT,SUSOUT,COLOUT,                             | TREA 25 |
|   | * ISTMOD,ISTTYP,ISTOUT,                                           | TREA 26 |
|   | * QPUMP,DSTART,CSTOP,                                             | TREA 27 |
|   | * DTUN,STGRMX,DTPUMP,DTMURE,STORF,APLAN,                          | TREA 28 |
|   | * CLAND,CSTOR,CPS,CTOTAL,CPCUYD,CPACRE,                           | TREA 29 |
|   | * LP,JP,LPREV,LABEL,DETENT(150),FRAC(150)                         | TREA 30 |
|   | DIMENSION NM(11),MHOUR(11),MMIN(11),BDRD(11),SSRD(11),CQRD(11),   | TREA 31 |
|   | * BDCIF(11),BDCRL(11),SSCIF(11),SSCRL(11),CDCIF(11),              | TREA 32 |
|   | * CDCRL(11),QAV(11)                                               | TREA 33 |
| C |                                                                   | TREA 34 |
|   | IF (KDT .GT. 1) GO TO 10                                          | TREA 35 |
| C |                                                                   | TREA 36 |
| C | INITIALIZE                                                        | TREA 37 |
| C |                                                                   | TREA 38 |
|   | BIG = 10.0**12                                                    | TREA 39 |
|   | CORPT = 0.0                                                       | TREA 40 |
|   | J1 = JM(1)                                                        | TREA 41 |
|   | J3 = JM(3)                                                        | TREA 42 |
|   | J4 = JM(4)                                                        | TREA 43 |
|   | J5 = JM(5)                                                        | TREA 44 |
|   | J7 = JM(7)                                                        | TREA 45 |
|   | DO 5 K=1,7                                                        | TREA 46 |
|   | BDINT(K) = 0.0                                                    | TREA 47 |
|   | SSINT(K) = 0.0                                                    | TREA 48 |
|   | WAIN(K) = 0.0                                                     | TREA 49 |
|   | WARM(K) = 0.0                                                     | TREA 50 |
|   | BDRMT(K) = 0.0                                                    | TREA 51 |
|   | SSRMT(K) = 0.0                                                    | TREA 52 |
|   | BDOU(K) = 0.0                                                     | TREA 53 |
|   | SSOU(K) = 0.0                                                     | TREA 54 |
|   | WAOUT(K) = 0.0                                                    | TREA 55 |
|   | IF (IRANGE .EQ. 0) GO TO 5                                        | TREA 56 |
|   | QQUMX(K) = 0.0                                                    | TREA 57 |
|   | BCUMX(K) = 0.0                                                    | TREA 58 |
|   | SCUMX(K) = 0.0                                                    | TREA 59 |
|   | QCRM(K) = 0.0                                                     | TREA 60 |

```

BCRMMX(K)= 0.0
SCRMMX(K)= 0.0
QQOUT(K) = 0.0
BCOUT(K) = 0.0
SCOUT(K) = 0.0
QQRMT(K) = 0.0
BCRMT(K) = 0.0
SCRMT(K) = 0.0
QQOUMN(K)= BIG
BCOUMN(K)= BIG
SCOUMN(K)= BIG
QQRMMN(K)= BIG
BCRMMN(K)= BIG
SCRMMN(K)= BIG
CL2UT(K) = 0.0
5  CHEMUT(K)= 0.0
   CHEMUT(8)= 0.0
   CL2LT(8) = 0.0
   ALUMUT  = 0.0
   IA=0
   KNTGF = 0
   SLOAC = 0.0
   TCHEM = 0.0
   CHCUST= 0.0
   BDIFT = 0.0
   SSIFT = 0.0
   WAIFT = 0.0
   BDOFT = 0.0
   SSGFT = 0.0
   WAOFT = 0.0
   BDRLT = 0.0
   SSRLT = 0.0
   WARLT = 0.0
   WARST = 0.0
   BDRST = 0.0
   SSRST = 0.0
   COIFT = 0.0
   COOFT = 0.0
   COINT = 0.0
   COOUT = 0.0
   CORLT = 0.0

```

C

```

IF (IRANGE .EQ. 0) GO TO 7
QQIFMX = 0.0
BCIFMX = 0.0
SCIFMX = 0.0
CCIFMX = 0.0
QQCFMX = 0.0
BCCFMX = 0.0
SCCFMX = 0.0
CCCFMX = 0.0
QQINMX = 0.0
BCINMX = 0.0
SCINMX = 0.0
CCINMX = 0.0
QQRIMX = 0.0
BCRLMX = 0.0
SCRIMX = 0.0
CCRLMX = 0.0
CCOUMX = 0.0

```

```

TREA 61
TREA 62
TREA 63
TREA 64
TREA 65
TREA 66
TREA 67
TREA 68
TREA 69
TREA 70
TREA 71
TREA 72
TREA 73
TREA 74
TREA 75
TREA 76
TREA 77
TREA 78
TREA 79
TREA 80
TREA 81
TREA 82
TREA 83
TREA 84
TREA 85
TREA 86
TREA 87
TREA 88
TREA 89
TREA 90
TREA 91
TREA 92
TREA 93
TREA 94
TREA 95
TREA 96
TREA 97
TREA 98
TREA 99
TREA100
TREA101
TREA102
TREA103
TREA104
TREA105
TREA106
TREA107
TREA108
TREA109
TREA110
TREA111
TREA112
TREA113
TREA114
TREA115
TREA116
TREA117
TREA118
TREA119
TREA120

```

|                                                                      |         |
|----------------------------------------------------------------------|---------|
| QQIFMA = BIG                                                         | TREA121 |
| BCIFMA = BIG                                                         | TREA122 |
| SCIFMA = BIG                                                         | TREA123 |
| CCIFMA = BIG                                                         | TREA124 |
| QQOFMA = BIG                                                         | TREA125 |
| BCOFMA = BIG                                                         | TREA126 |
| SCOFMA = BIG                                                         | TREA127 |
| CCOFMA = BIG                                                         | TREA128 |
| QQINMA = BIG                                                         | TREA129 |
| BCINMA = BIG                                                         | TREA130 |
| SCINMA = BIG                                                         | TREA131 |
| CCINMA = BIG                                                         | TREA132 |
| QQRLMA = BIG                                                         | TREA133 |
| BCRLMA = BIG                                                         | TREA134 |
| SCRLMA = BIG                                                         | TREA135 |
| CCRLMA = BIG                                                         | TREA136 |
| CCOUMA = BIG                                                         | TREA137 |
| QQIFT = 0.0                                                          | TREA138 |
| BCIFT = 0.0                                                          | TREA139 |
| SCIFT = 0.0                                                          | TREA140 |
| CCIFT = 0.0                                                          | TREA141 |
| QQOFT = 0.0                                                          | TREA142 |
| BCOFT = 0.0                                                          | TREA143 |
| SCOFT = 0.0                                                          | TREA144 |
| CCOFT = 0.0                                                          | TREA145 |
| QQINT = 0.0                                                          | TREA146 |
| BCINT = 0.0                                                          | TREA147 |
| SCINT = 0.0                                                          | TREA148 |
| CCINT = 0.0                                                          | TREA149 |
| QQRLT = 0.0                                                          | TREA150 |
| BCRLT = 0.0                                                          | TREA151 |
| SCRLT = 0.0                                                          | TREA152 |
| CCRLT = 0.0                                                          | TREA153 |
| CCOUT = 0.0                                                          | TREA154 |
| C                                                                    |         |
| 7 RL = NDT - 1                                                       | TREA155 |
| ND = RL/10.0                                                         | TREA156 |
| NM(1) = 1                                                            | TREA157 |
| DO 8 M=2,11                                                          | TREA158 |
| MM = M - 1                                                           | TREA159 |
| 8 NM(M) = ND*MM + 1                                                  | TREA160 |
| M = 1                                                                | TREA161 |
| IF (IPRINT .EQ. 0) GO TO 10                                          | TREA162 |
| WRITE(6,457)                                                         | TREA163 |
| IF (ITREAT(4) .EQ. 43 .AND. IPRINT .EQ. 2) WRITE(6,458)              | TREA164 |
| WRITE (6,459) (NAME(I,J3), I=1,4), (NAME(I,J4), I=1,4),              | TREA165 |
| * (NAME(I,J7), I=1,4)                                                | TREA166 |
| 457 FORMAT('1PERFORMANCE PER TIME STEP', /,                          | TREA167 |
| * '0',10X,'NOTE: NO BOD OR SS ARE REMOVED IN LEVELS 2,',             | TREA168 |
| * '5, & 6, REGARDLESS OF THE OPTIONS SELECTED', /, ' ', 17X,         | TREA169 |
| * 'NO SS REMOVALS IN LEVEL 7 (CHLORINE CONTACT TANK)', /,            | TREA170 |
| * ' ', 17X, 'LEVEL 1 & 5 REMOVALS (AT BAR RACKS AND ',               | TREA171 |
| * 'EFFLUENT SCREENS) ARE REPORTED IN SUMMARY ONLY')                  | TREA172 |
| 458 FORMAT(' ', 17X, '***** IN HIGH RATE FILTER REMOVALS INDICATES', | TREA173 |
| * ' MATERIAL IN FILTER BED')                                         | TREA174 |
| 459 FORMAT('0 TIME', 16X, 'INFLOWS', 17X, 4A4, 3X, 4A4, 5X, 4A4,     | TREA175 |
| * 13X, 'OUTFLOWS', /,                                                | TREA176 |
| * ' ', 10X, '-----', 4X,                                             | TREA177 |
| * 3('-----'), '-----',                                               | TREA178 |
| * '-----', /, ' ', 11X, 'WATER BOD SS COLIFORMS', 4X,                | TREA179 |
|                                                                      | TREA180 |

|                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                  |                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                              |
|----------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|--------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| <pre> *      2(' TOTAL BOD SS'), ' TOTAL BOD COLIFORMS', *      ' TOTAL BOD SS COLIFORMS') IF (IPRINT.EQ. 1) WRITE(6,460) 460 FORMAT(' HR:MIN      CF      LB      LB      MPN', 4X, *      2('      CF      LB      LB', '      CF      LB      MPN', *      '      CF      LB      LB      MPN', //) IF (IPRINT.EQ. 2) WRITE(6,461) 461 FORMAT(' HR:MIN      CFS MG/L MG/L MPN/100ML', 4X, *      2('      CFS MG/L MG/L', '      CFS MG/L MPN/100ML', *      '      CFS MG/L MG/L MPN/100ML') C C      COMPUTE CLOCK TIME C 10  KMIN  = KMIN + DT 20 IF (KMIN .LT. 60) GO TO 30     KMIN  = KMIN - 60     K HOUR = K HOUR + 1     IF (K HOUR .GT. 23) K HOUR = K HOUR - 24     GO TO 20 30 CONTINUE C C      QQ..  = WATER FLOWRATE (CFS) C      BD..  = BOD FLOWRATE (LB/DT) C      SS..  = SS FLOWRATE (LB/DT) C      CD..  = COLIFORM FLOW RATE (MPN/DT) C      WA..  = WATER QUANTITY (CU.FT) C      BC..  = BOD CONCENTRATION (MG/L) C      SC..  = SS CONCENTRATION (MG/L) C      CC..  = COLIFORMS CONCENTRATION (MPN/100ML) C C      ..IF  = INFLOW (TO WHOLE MODEL, INCL OVERFLOW) C      ..OF  = OVERFLOW C      ..IN  = INFLOW (BY LEVEL) C      ..RM  = REMOVAL (BY LEVEL) FROM EFFLUENT (OUTFL) C      ..OU  = OUTFLOW (BY LEVEL) C      ..RL  = RELEASE (OUTFLOW FROM WHOLE MODEL, INCL OVE C C      ..RS  = REMOVAL BY SCREENS (SUBTOTALS FOR MODE 33 O C      ..RD  = REDUCTION (PERCENT, OF A QUANTITY) C C      ....T = TOTAL (CUMULATIVE) C      ....TT = OVERALL TOTAL (FOR ALL LEVELS) C      ....RF = REMOVAL FRACTION C C      ....MX = MAXIMUM (THROUGHOUT STORM) C      ....MN = MINIMUM (THROUGHOUT STORM) C C      NOTE: INPUT TO TREATMENT IS INPUT TO LEVEL 1 C C      MG/L  = 16050*(LB/CU.FT.) C C      QDESYN = DESIGN THROUGH FLOWRATE FOR WHOLE TREATMENT C C      CCMPUTE CVERFLOW C C      QQGF  = 0.0 C      BDDF  = 0.0 C      SSDF  = 0.0 C      CDDF  = 0.0 C      WADF  = 0.0 </pre> | <pre> TREA181 TREA182 TREA183 TREA184 TREA185 TREA186 TREA187 TREA188 TREA189 TREA190 TREA191 TREA192 TREA193 TREA194 TREA195 TREA196 TREA197 TREA198 TREA199 TREA200 TREA201 TREA202 TREA203 TREA204 TREA205 TREA206 TREA207 TREA208 TREA209 TREA210 TREA211 TREA212 TREA213 TREA214 TREA215 TREA216 TREA217 TREA218 TREA219 TREA220 TREA221 TREA222 TREA223 TREA224 TREA225 TREA226 TREA227 TREA228 TREA229 TREA230 TREA231 TREA232 TREA233 TREA234 TREA235 TREA236 TREA237 TREA238 TREA239 TREA240 </pre> |
|----------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|--------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|

|                                                                                        |         |
|----------------------------------------------------------------------------------------|---------|
| BCOF = 0.0                                                                             | TREA241 |
| SCOF = 0.0                                                                             | TREA242 |
| CCOF = 0.0                                                                             | TREA243 |
| IF (ITREAT(3) .EQ. 35) GC TO (40, 35), ISTOR                                           | TREA244 |
| FLOW FIRST THROUGH ASSOCIATED STORAGE                                                  | TREA245 |
| C                                                                                      | TREA246 |
| C                                                                                      | TREA247 |
| 35 IF(ISTOR.NE.02)GO TO 40                                                             | TREA248 |
| QIN(KDT) = QQIF                                                                        | TREA249 |
| BODIN(KDT) = BDIF                                                                      | TREA250 |
| SUSIN(KDT) = SSIF                                                                      | TREA251 |
| COLIN(KDT) = COIF                                                                      | TREA252 |
| C                                                                                      | TREA253 |
| C                                                                                      | TREA254 |
| ABOVE MAY NEED TO BE REDUCED BY NU-FLOOD FLOW SPLITTING TO PREVENT FLOODING IN STORAGE | TREA255 |
| IF (DEPTH .LT. LEPMAX) GO TO 39                                                        | TREA256 |
| IF (QQIF .LE. 0.0) GO TO 39                                                            | TREA257 |
| FR2ST = COMAX/QQIF                                                                     | TREA258 |
| QIN(KDT) = FR2ST*QQIF                                                                  | TREA259 |
| BODIN(KDT) = FR2ST*BDIF                                                                | TREA260 |
| SUSIN(KDT) = FR2ST*SSIF                                                                | TREA261 |
| COLIN(KDT) = FR2ST*COIF                                                                | TREA262 |
| QQOF = (1.0-FR2ST)*QQIF                                                                | TREA263 |
| BDOF = (1.0-FR2ST)*BDIF                                                                | TREA264 |
| SSOF = (1.0-FR2ST)*SSIF                                                                | TREA265 |
| CCOF = (1.0-FR2ST)*CCOF                                                                | TREA266 |
| 39 QINST = QIN(KDT)                                                                    | TREA267 |
| C                                                                                      | TREA268 |
| CALL STRAGE                                                                            | TREA269 |
| C                                                                                      | TREA270 |
| QQIF = CQUST                                                                           | TREA271 |
| BDIF = BCOCUT                                                                          | TREA272 |
| SSIF = SUSOUT                                                                          | TREA273 |
| COIF = CCLOUT                                                                          | TREA274 |
| C                                                                                      | TREA275 |
| NO FLOW THROUGH ASSOCIATED STORAGE                                                     | TREA276 |
| 40 IF (QQIF .GT. 0.0) GO TO 50                                                         | TREA277 |
| WAIF = 0.0                                                                             | TREA278 |
| BCIF = 0.0                                                                             | TREA279 |
| SCIF = 0.0                                                                             | TREA280 |
| COIF = 0.0                                                                             | TREA281 |
| CCIF = 0.0                                                                             | TREA282 |
| QQIN(1) = 0.0                                                                          | TREA283 |
| WAIN(1) = 0.0                                                                          | TREA284 |
| BODIN(1) = 0.0                                                                         | TREA285 |
| COIN = 0.0                                                                             | TREA286 |
| SSIN(1) = 0.0                                                                          | TREA287 |
| BCIN(1) = 0.0                                                                          | TREA288 |
| SCIN(1) = 0.0                                                                          | TREA289 |
| CCIN = 0.0                                                                             | TREA290 |
| GO TO 80                                                                               | TREA291 |
| 50 QQIN(1) = QDESYN                                                                    | TREA292 |
| IF (QQIF .LT. QDESYN) QQIN(1) = QQIF                                                   | TREA293 |
| QQOFR = CCIF - CQIN(1)                                                                 | TREA294 |
| QQOF = CCOF + QQOFR                                                                    | TREA295 |
| OFACT = QQOFR/QQIF                                                                     | TREA296 |
| C                                                                                      | TREA297 |
| ASSUMES BOD, SS & COLIFORMS ARE THOROUGHLY MIXED ON                                    | TREA298 |
| BDOF = OFACT*BDIF + BDOF                                                               | TREA299 |
| SSOF = OFACT*SSIF + SSOF                                                               | TREA300 |
| COOF = OFACT*COIF + CCOF                                                               |         |
| WDOF = QQOF*DT*60.0                                                                    |         |
| WDOFT = WDOFT + WDOF                                                                   |         |
| WAIF = QQIF*DT*60.0                                                                    |         |



|                                           |         |
|-------------------------------------------|---------|
| BCIF = 16050.0*BDIF/WAIF                  | TREA301 |
| SCIF = 16050.0*SSIF/WAIF                  | TREA302 |
| CCIF = COIF/(WAIF*28.32*10.0)             | TREA303 |
| IF (CQOF .LT. 0.001) GO TO 60             | TREA304 |
| BCOF = 16050.0*BDOF/WAOF                  | TREA305 |
| SCOF = 16050.0*SSOF/WAOF                  | TREA306 |
| CCOF = CCOF/(WAOF*28.32*10.0)             | TREA307 |
| KNTOF = KNTOF + 1                         | TREA308 |
| GO TO 70                                  | TREA309 |
| 60 BCOF = 0.0                             | TREA310 |
| SCOF = 0.0                                | TREA311 |
| CCOF = 0.0                                | TREA312 |
| 70 BDOFT = BDOFT + BDOF                   | TREA313 |
| SSOFT = SSOFT + SSOF                      | TREA314 |
| COOFT = COOFT + CCOF                      | TREA315 |
| BDIFT = BDIFT + BDIF                      | TREA316 |
| SSIFT = SSIFT + SSIF                      | TREA317 |
| WAIFT = WAIFT + WAIF                      | TREA318 |
| COIFT = COIFT + COIF                      | TREA319 |
| BDIN(1) = (1.0-CFACT)*BCIF                | TREA320 |
| SSIN(1) = (1.0-OFAC)*SSIF                 | TREA321 |
| COIN = (1.0-OFAC)*COIF                    | TREA322 |
| CGINT = CGINT + COIN                      | TREA323 |
| WAIN(1) = QQIN(1)*DT*60.0                 | TREA324 |
| BCIN(1) = BCIF                            | TREA325 |
| SCIN(1) = SCIF                            | TREA326 |
| CCIN = CCIF                               | TREA327 |
| C                                         | TREA328 |
| 80 IF (IRANGE .EQ. 0) GO TO 1000          | TREA329 |
| IF (QQIF .GT. QQIFMX) QQIFMX = QQIF       | TREA330 |
| IF (QQOF .GT. QQOFMX) QQOFMX = QQOF       | TREA331 |
| IF (QQIN(1) .GT. QQINMX) QQINMX = QQIN(1) | TREA332 |
| IF (BCIF .GT. BCIFMX) BCIFMX = BCIF       | TREA333 |
| IF (BCOF .GT. BCOFMX) BCOFMX = BCOF       | TREA334 |
| IF (BCIN(1) .GT. BCINMX) BCINMX = BCIN(1) | TREA335 |
| IF (SCIF .GT. SCIFMX) SCIFMX = SCIF       | TREA336 |
| IF (SCOF .GT. SCOFMX) SCOFMX = SCOF       | TREA337 |
| IF (SCIN(1) .GT. SCINMX) SCINMX = SCIN(1) | TREA338 |
| IF (CCIF .GT. CCIFMX) CCIFMX = CCIF       | TREA339 |
| IF (CCOF .GT. CCOFMX) CCOFMX = CCOF       | TREA340 |
| IF (CCIN .GT. CCINMX) CCINMX = CCIN       | TREA341 |
| IF (QQIF .LT. QQIFMN) QQIFMN = QQIF       | TREA342 |
| IF (QQOF .LT. QQOFMN) QQOFMN = QQOF       | TREA343 |
| IF (QQIN(1) .LT. QQINMN) QQINMN = QQIN(1) | TREA344 |
| IF (BCIF .LT. BCIFMN) BCIFMN = BCIF       | TREA345 |
| IF (BCOF .LT. BCOFMN) BCOFMN = BCOF       | TREA346 |
| IF (BCIN(1) .LT. BCINMN) BCINMN = BCIN(1) | TREA347 |
| IF (SCIF .LT. SCIFMN) SCIFMN = SCIF       | TREA348 |
| IF (SCOF .LT. SCOFMN) SCOFMN = SCOF       | TREA349 |
| IF (SCIN(1) .LT. SCINMN) SCINMN = SCIN(1) | TREA350 |
| IF (CCIF .LT. CCIFMN) CCIFMN = CCIF       | TREA351 |
| IF (CCOF .LT. CCOFMN) CCOFMN = CCOF       | TREA352 |
| IF (CCIN .LT. CCINMN) CCINMN = CCIN       | TREA353 |
| QQIFT = QQIFT + QQIF                      | TREA354 |
| QQOFT = QQOFT + QQOF                      | TREA355 |
| QQINT = QQINT + QQIN(1)                   | TREA356 |
| BCIFT = BCIFT + BCIF                      | TREA357 |
| BCOFT = BCOFT + BCOF                      | TREA358 |
| BCINT = BCINT + BCIN(1)                   | TREA359 |
| SCIFT = SCIFT + SCIF                      | TREA360 |

|                                                       |         |
|-------------------------------------------------------|---------|
| SCOFT = SCOFT + SCOF                                  | TREA361 |
| SCINT = SCINT + SCIN(1)                               | TREA362 |
| CCIFT = CCIFT + CCIF                                  | TREA363 |
| CCOFT = CCOFT + CCOF                                  | TREA364 |
| CCINT = CCINT + CCIN                                  | TREA365 |
| C                                                     | TREA366 |
| IF(ISTOR.NE.02)GO TO 1000                             | TREA367 |
| IF (QOUST .EQ. 0.0) GO TO 9998                        | TREA368 |
| C----- START REMOVALS. IN CASCADING LEVELS -----      | TREA369 |
| C                                                     | TREA370 |
| C----- LEVEL 1 BRANCH (FOR TREATMENT BY BAR RACKS)    | TREA371 |
| C                                                     | TREA372 |
| 1000 IF(QQIF.LE.0.0) GO TO 9998                       | TREA373 |
| K = ITREAT(1) - 10                                    | TREA374 |
| L = 1                                                 | TREA375 |
| BDINT(1) = BDINT(1) + BDIN(1)                         | TREA376 |
| SSINT(1) = SSINT(1) + SSIN(1)                         | TREA377 |
| WAIN(1) = WAIN(1) + WAIN(1)                           | TREA378 |
| GO TO (1100,1200), K                                  | TREA379 |
| C                                                     | TREA380 |
| C 11) FOR NO BAR RACKS                                | TREA381 |
| C                                                     | TREA382 |
| 1100 CALL BYPASS                                      | TREA383 |
| GO TO 1900                                            | TREA384 |
| C                                                     | TREA385 |
| C 12) FOR BAR RACKS                                   | TREA386 |
| C                                                     | TREA387 |
| 1200 WARM(1) = 6.0*QQIN(1)*0.646*DT/1440.0            | TREA388 |
| QQR(1) = WARM(1)/(DT*60.0)                            | TREA389 |
| C SCREENINGS (85% MOISTURE) AT 6 CU.FT/MG, & 50 LB/CT | TREA390 |
| SSRM(1) = WARM(1)*50.0*0.15                           | TREA391 |
| BDRM(1) = 0.05*SSRM(1)                                | TREA392 |
| QQCU(1) = QQIN(1) - QQR(1)                            | TREA393 |
| WACU(1) = WAIN(1) - WARM(1)                           | TREA394 |
| BDCU(1) = BDIN(1) - BDRM(1)                           | TREA395 |
| SSCU(1) = SSIN(1) - SSRM(1)                           | TREA396 |
| C                                                     | TREA397 |
| 1220 IF (QQCU(1) .EQ. 0.0) GO TO 1230                 | TREA398 |
| BCOU(1) = BDCU(1)*CCNVER/QQCU(1)                      | TREA399 |
| SCOU(1) = SSCU(1)*CCNVER/QQCU(1)                      | TREA400 |
| GO TO 1240                                            | TREA401 |
| 1230 BCOU(1) = 0.0                                    | TREA402 |
| SCOU(1) = 0.0                                         | TREA403 |
| 1240 IF (QQR(1) .EQ. 0.0) GO TO 1250                  | TREA404 |
| BCRM(1) = BDRM(1)*CCNVER/QQR(1)                       | TREA405 |
| SCRM(1) = SSRM(1)*CCNVER/QQR(1)                       | TREA406 |
| GO TO 1900                                            | TREA407 |
| 1250 BCRM(1) = 0.0                                    | TREA408 |
| SCRM(1) = 0.0                                         | TREA409 |
| C                                                     | TREA410 |
| 1900 CALL TRLINK                                      | TREA411 |
| C                                                     | TREA412 |
| C----- LEVEL 2 BRANCH (FOR INLET PUMPING)             | TREA413 |
| C                                                     | TREA414 |
| 2000 K = ITREAT(2) - 20                               | TREA415 |
| L = 2                                                 | TREA416 |
| BDINT(2) = BDINT(2) + BDIN(2)                         | TREA417 |
| SSINT(2) = SSINT(2) + SSIN(2)                         | TREA418 |
| WAIN(2) = WAIN(2) + WAIN(2)                           | TREA419 |
| GO TO (2100,2200), K                                  | TREA420 |

|   |                                                                       |         |
|---|-----------------------------------------------------------------------|---------|
| C |                                                                       | TREA421 |
| C | 21) FOR GRAVITY INLET (NO PUMPING, = BYPASS)                          | TREA422 |
| C |                                                                       | TREA423 |
| C | 2100 CALL BYPASS                                                      | TREA424 |
| C | GO TO 29C0                                                            | TREA425 |
| C |                                                                       | TREA426 |
| C | 22) FOR INLET PUMPING (STATION)                                       | TREA427 |
| C |                                                                       | TREA428 |
| C | 2200 CALL BYPASS                                                      | TREA429 |
| C |                                                                       | TREA430 |
| C | 2900 CALL TRLINK                                                      | TREA431 |
| C |                                                                       | TREA432 |
| C | ----- LEVEL 3 BRANCH (FOR PRIMARY TREATMENT)                          | TREA433 |
| C |                                                                       | TREA434 |
| C | 3000 K = ITREAT(3) - 30                                               | TREA435 |
| C | L = 3                                                                 | TREA436 |
| C | BDINT(3) = BDINT(3) + BDIN(3)                                         | TREA437 |
| C | SSINT(3) = SSINT(3) + SSIN(3)                                         | TREA438 |
| C | WAIN(3) = WAIN(3) + WAIN(3)                                           | TREA439 |
| C | GO TO (3100,3200,3400,3400,3500), K                                   | TREA440 |
| C |                                                                       | TREA441 |
| C | 31) FOR NO PRIMARY TREATMENT (BYPASS)                                 | TREA442 |
| C |                                                                       | TREA443 |
| C | 3100 CALL BYPASS                                                      | TREA444 |
| C | GO TO 390C                                                            | TREA445 |
| C |                                                                       | TREA446 |
| C | 32) & 33) FOR DISSOLVED AIR FLOTATION                                 | TREA447 |
| C | (WITH OR WITHOUT CHEMICALS)                                           | TREA448 |
| C |                                                                       | TREA449 |
| C | SELECT CONTINUATION THRU PROGRAM DEPENDING ON USE OR NON-USE OF       | TREA450 |
| C | CHEMICAL CCAGULANTS                                                   | TREA451 |
| C | 3200 OVFR = QQIN(3)*1000000.0*(1.0+0.01*RECIRC)/(1.547*ISJRFA)        | TREA452 |
| C | OVFR OVERFLOW RATE, GPD/SF                                            | TREA453 |
| C |                                                                       | TREA454 |
| C | *** COMPUTE REMOVALS IN FLOTATION CHAMBER ASSUMING NO FLOC. CHEMICALS | TREA455 |
| C |                                                                       | TREA456 |
| C | IF (OVFR .LT. 1000.0) OVFR = 1000.0                                   | TREA457 |
| C | BREFF = 0.59 + 0.05*BCIN(3)/100.0 - 0.36*(OVFR-1000.0)/7000.0         | TREA458 |
| C | SREFF = 0.656 + 0.06*SCIN(3)/190.0 - 0.40*(OVFR-1000.0)/7000.0        | TREA459 |
| C | IF (ICHEM .NE. 1) GO TO 3205                                          | TREA460 |
| C |                                                                       | TREA461 |
| C | *** COMPUTE REMOVALS IN FLOTATION CHAMBER ASSUMING FLOC CHEMICALS ARE | TREA462 |
| C |                                                                       | TREA463 |
| C | BREFF = BREFF + 0.02                                                  | TREA464 |
| C | SREFF = SREFF + (20000.0-OVFR)/100000.0                               | TREA465 |
| C | CHEMU=QQIN(3)*DT*60.0*7.48*100.0/1000000.                             | TREA466 |
| C | CHEMUT(3) = CHEMUT(3) + CHEMU                                         | TREA467 |
| C | 3205 IF (ICL2 .NE. 1) GO TO 3210                                      | TREA468 |
| C | BREFF = BREFF + 0.15                                                  | TREA469 |
| C | ADDED CL2 REDUCES BOD & COLIFORMS BUT DOES NOT AFFECT                 | TREA470 |
| C | IF (BCIN(3) .GT. 130.0) CL2U = QQIN(3)*15.0*8.34*DT/                  | TREA471 |
| C | (1.547*60.0*24.0)                                                     | TREA472 |
| C | * IF (BCIN(3) .LE. 130.0) CL2U = QQIN(3)*10.0*8.34*DT/                | TREA473 |
| C | (1.547*60.0*24.0)                                                     | TREA474 |
| C | * CL2UT(3) = CL2UT(3) + CL2U                                          | TREA475 |
| C | CALL KILL(VCLDAF,SCIF,SCIN(3),QQIN(3),CCIN,CCOU)                      | TREA476 |
| C | 3210 CONTINUE                                                         | TREA477 |
| C | IF (BREFF .GE. 0.60) BREFF = 0.60                                     | TREA478 |
| C | IF (SREFF .LE. 0.18) SREFF = 0.18                                     | TREA479 |
| C | IF (SREFF .GE. 0.82) SREFF = 0.82                                     | TREA480 |

|                                                                |         |
|----------------------------------------------------------------|---------|
| IF (SREFF .LE. 0.20) SREFF = 0.20                              | TREA481 |
| BDRM(3) = BREFF*BDRM(3)                                        | TREA482 |
| SSRM(3) = SREFF*SSIN(3)                                        | TREA483 |
| QCRM(3) = 0.015*QQIN(3)                                        | TREA484 |
| C QCRM(3) IS FLOATED SCUM VOLUME                               | TREA485 |
| GO TO 3220                                                     | TREA486 |
| C                                                              | TREA487 |
| 3203 QOUS = QQIN(3) - QQRS                                     | TREA488 |
| OVFRA = QOUS*1000000.0*(1.0+0.01*RECIRC)/(1.547*TSJRFA)        | TREA489 |
| IF (OVFRA .LT. 1000.0) OVFRA = 1000.0                          | TREA490 |
| C NC CHEMICALS USED                                            | TREA491 |
| BDOUS = BDRM(3) - BDRS                                         | TREA492 |
| BCOLS = 0.0                                                    | TREA493 |
| IF (QOLS .GT. 0.0) BCUS = BDOUS*CONVER/QOUS                    | TREA494 |
| SSOUS = SSIN(3) - SSRS                                         | TREA495 |
| SCCUS = 0.0                                                    | TREA496 |
| IF (QOUS .GT. 0.0) SCOUS = SSOUS*CONVER/QOUS                   | TREA497 |
| SREFF = 0.528 - 0.486*(OVFRA-1000.0)/7000.0 + 0.06*SCOUS/190.0 | TREA498 |
| BREFF = 0.475 + 0.05*BCOUS/100.0 - 0.405*(OVFRA-1000.0)/7000.0 | TREA499 |
| IF (ICL2 .NE. 1) GO TO 3215                                    | TREA500 |
| C CHEMICALS USED                                               | TREA501 |
| SREFF = SREFF + 1.37*(20000.0-OVFRA)/100000.0                  | TREA502 |
| BREFF = BREFF + 1.30*0.02                                      | TREA503 |
| CHEMU = QOUS*DT*60.0*7.48*100.0/1000000.0                      | TREA504 |
| C                                                              | TREA505 |
| C CHEMU IS CHEMICALS USED PER TIMESTEP BASED ON USE AT 12 MG/L | TREA506 |
| CHEMUT(3) = CHEMUT(3) + CHEMU                                  | TREA507 |
| 3215 IF (ICL2 .NE. 1) GO TO 3219                               | TREA508 |
| BREFF = BREFF + 0.15*1.30                                      | TREA509 |
| C CL2 REDUCES BOD & COLIFORMS, BUT NOT SS                      | TREA510 |
| IF (BCOUS .GT. 130.0) CL2U = QOUS*15.0*8.34*DT/                | TREA511 |
| * (1.547*60.0*24.0)                                            | TREA512 |
| IF (BCCUS .LE. 130.0) CL2U = QOUS*10.0*8.34*DT/                | TREA513 |
| * (1.547*60.0*24.0)                                            | TREA514 |
| CL2UT(3) = CL2UT(3) + CL2U                                     | TREA515 |
| CALL KILL(VCLDAF,SCIF,SCIN(3),QQIN(3),CCIN,CCOU)               | TREA516 |
| 3219 CONTINUE                                                  | TREA517 |
| IF (BREFF .GT. 0.48) BREFF = 0.48                              | TREA518 |
| IF (BREFF .LT. 0.15) BREFF = 0.15                              | TREA519 |
| IF (SREFF .GT. 0.75) SREFF = 0.75                              | TREA520 |
| IF (SREFF .LT. 0.15) SREFF = 0.15                              | TREA521 |
| BDRM(3) = BDRS + BREFF*BDOUS                                   | TREA522 |
| SSRM(3) = SSRS + SREFF*SSOUS                                   | TREA523 |
| QCRM(3) = QQRS + QOUS*0.010                                    | TREA524 |
| 3220 QQOU(3) = QQIN(3) - QCRM(3)                               | TREA525 |
| WAOU(3) = QQOU(3)*DT*60.0                                      | TREA526 |
| BDOU(3) = BDRM(3) - BDRM(3)                                    | TREA527 |
| SSOU(3) = SSIN(3) - SSRM(3)                                    | TREA528 |
| WARM(3) = WAIN(3) - WAOU(3)                                    | TREA529 |
| C                                                              | TREA530 |
| IF (QQCU(3) .EQ. 0.0) GO TO 3230                               | TREA531 |
| BCOU(3) = BDOU(3)*CONVER/QQCU(3)                               | TREA532 |
| SCCU(3) = SSOU(3)*CONVER/QQCU(3)                               | TREA533 |
| GO TO 3240                                                     | TREA534 |
| 3230 BCOU(3) = 0.0                                             | TREA535 |
| SCOU(3) = 0.0                                                  | TREA536 |
| 3240 IF (QCRM(3) .EQ. 0.0) GO TO 3250                          | TREA537 |
| BCRM(3) = BDRM(3)*CONVER/QCRM(3)                               | TREA538 |
| SSRM(3) = SSRM(3)*CONVER/QCRM(3)                               | TREA539 |
| GO TO 3900                                                     | TREA540 |

|      |                                                                        |         |
|------|------------------------------------------------------------------------|---------|
| 3250 | BCRM(3) = 0.0                                                          | TREA541 |
|      | SCRM(3) = 0.0                                                          | TREA542 |
|      | GO TO 3900                                                             | TREA543 |
| C    |                                                                        | TREA544 |
| C    | 33) & 34) FOR FINE SCREENS                                             | TREA545 |
| C    |                                                                        | TREA546 |
| 3400 | BDRM(3) = 0.2200*BDIN(3)                                               | TREA547 |
|      | SSRM(3) = 0.2700*SSIN(3)                                               | TREA548 |
|      | QCRM(3) = 0.0075*QQIN(3)                                               | TREA549 |
|      | WARM(3) = QCRM(3)*DT*60.0                                              | TREA550 |
|      | WARS = WARM(3)                                                         | TREA551 |
|      | BDRS = BDRM(3)                                                         | TREA552 |
|      | SSRS = SSRM(3)                                                         | TREA553 |
|      | QQRS = QCRM(3)                                                         | TREA554 |
|      | IF (ITREAT(3) .EQ. 33) GO TO 3440                                      | TREA555 |
|      | QQOU(3) = QQIN(3) - QCRM(3)                                            | TREA556 |
|      | WAOU(3) = QQOU(3)*DT*60.0                                              | TREA557 |
|      | BDOU(3) = BDIN(3) - BDRM(3)                                            | TREA558 |
|      | SSOU(3) = SSIN(3) - SSRM(3)                                            | TREA559 |
|      | IF (QQOU(3) .EQ. 0.0) GO TO 3410                                       | TREA560 |
|      | BCOU(3) = BDOU(3)*CCNVER/QQOU(3)                                       | TREA561 |
|      | SCOU(3) = SSOU(3)*CCNVER/QQOU(3)                                       | TREA562 |
|      | GO TO 3420                                                             | TREA563 |
| 3410 | BCOU(3) = 0.0                                                          | TREA564 |
|      | SCOU(3) = 0.0                                                          | TREA565 |
| C    | BCRM(3) AND SCRM(3) ARE THE CONCENTRATION OF BOD & SS IN THE SCREENING | TREA566 |
| 3420 | IF (QCRM(3) .EQ. 0.0) GO TO 3430                                       | TREA567 |
|      | BCRM(3) = BDRM(3)*CCNVER/QCRM(3)                                       | TREA568 |
|      | SCRM(3) = SSRM(3)*CCNVER/QCRM(3)                                       | TREA569 |
|      | GO TO 3440                                                             | TREA570 |
| 3430 | BCRM(3) = 0.0                                                          | TREA571 |
|      | SCRM(3) = 0.0                                                          | TREA572 |
| 3440 | SLOAD = SSIN(3)/(DT*SCREEN)                                            | TREA573 |
| C    | SLOAD = SOLIDS LOADING ON SCREENS (LB/MIN/SQ.FT)                       | TREA574 |
|      | IF (SLOAD .GT. 0.14) IA=1                                              | TREA575 |
|      | IF (ITREAT(3) .NE. 33) GO TO 3900                                      | TREA576 |
|      | WARST = WARST + WARS                                                   | TREA577 |
|      | BDRST = BDRST + BDRS                                                   | TREA578 |
|      | SSRST = SSRST + SSRS                                                   | TREA579 |
|      | GO TO 3203                                                             | TREA580 |
| C    |                                                                        | TREA581 |
| C    | 35) FOR SEDIMENTATION                                                  | TREA582 |
| C    |                                                                        | TREA583 |
| 3500 | CALL SEDIM                                                             | TREA584 |
| C    |                                                                        | TREA585 |
| 3900 | CALL TRLINK                                                            | TREA586 |
| C    |                                                                        | TREA587 |
| C    | ----- LEVEL 4 BRANCH (FOR SECONDARY TREATMENT)                         | TREA588 |
| C    |                                                                        | TREA589 |
| 4000 | K = ITREAT(4) - 40                                                     | TREA590 |
|      | L = 4                                                                  | TREA591 |
|      | BDINT(4) = BDINT(4) + BDIN(4)                                          | TREA592 |
|      | SSINT(4) = SSINT(4) + SSIN(4)                                          | TREA593 |
|      | WAIN(4) = WAIN(4) + WAIN(4)                                            | TREA594 |
|      | GO TO (4100,4200,4300), K                                              | TREA595 |
| C    |                                                                        | TREA596 |
| C    | 41) FOR NO SECONDARY TREATMENT (BYPASS)                                | TREA597 |
| C    |                                                                        | TREA598 |
| 4100 | CALL BYPASS                                                            | TREA599 |
|      | GO TO 4900                                                             | TREA600 |

|        |                                                      |         |
|--------|------------------------------------------------------|---------|
| C      |                                                      | TREA601 |
| C      |                                                      | TREA602 |
| C      | 42) FOR MICROSTRAINERS                               | TREA603 |
|        | 4200 FMS = SQRT(400.0/TRIBA)                         | TREA604 |
|        | IF (FMS .GT. 1.0) FMS = 1.0                          | TREA605 |
| C      |                                                      | TREA606 |
|        | IF (SCIN(4) .LE. 70.0) SCRM1 = SCIN(4)*SCIN(4)/140.0 | TREA607 |
|        | IF (SCIN(4) .GT. 70.0) SCRM1 = SCIN(4) - 35.0        | TREA608 |
|        | GPMSF = 10000.0*FMS/SCRM1                            | TREA609 |
|        | IF (GPMSF .GE. 40.0) GPMSF = 40.0                    | TREA610 |
|        | CFSTR = GPMSF*AREAMS*NMS/449.0                       | TREA611 |
|        | IF (CFSTR .GT. QQIN(4)) CFSTR = QQIN(4)              | TREA612 |
|        | CFSOF = QQIN(4) - CFSTR                              | TREA613 |
|        | QORM(4) = 0.10*NMS                                   | TREA614 |
|        | QQU(4) = QQIN(4) - QORM(4)                           | TREA615 |
|        | WARM(4) = QORM(4)*DT*60.0                            | TREA616 |
|        | WAUU(4) = QQU(4)*DT*60.0                             | TREA617 |
|        | CFSTR2 = CFSTR - QORM(4)                             | TREA618 |
|        | PSSTR = CFSTR2*(SCIN(4)-SCRM1)*0.646*8.35*DT/1440.0  | TREA619 |
|        | PSSOF = CFSOF*SCIN(4)*0.646*8.35*DT/1440.0           | TREA620 |
|        | SSOU(4) = PSSTR + PSSOF                              | TREA621 |
|        | SSRM(4) = SSIN(4) - SSOU(4)                          | TREA622 |
|        | SCOU(4) = SSOU(4)/(QQU(4)*0.646*8.35*DT/1440.0)      | TREA623 |
|        | SCRM(4) = SSRM(4)/(0.10*NMS*0.646*8.35*DT/1440.0)    | TREA624 |
| C      |                                                      | TREA625 |
|        | IF (BCIN(4) .GE. 27.0) BCRM1 = BCIN(4) - 10.0        | TREA626 |
|        | IF (BCIN(4) .LT. 27.0) BCRM1 = BCIN(4)*17.0/27.0     | TREA627 |
|        | PBDTR = CFSTR2*(BCIN(4)-BCRM1)*0.646*8.35*DT/1440.0  | TREA628 |
|        | PBDUF = CFSTR2*BCIN(4)*0.646*8.35*DT/1440.0          | TREA629 |
|        | BDUU(4) = PBDTR + PBDUF                              | TREA630 |
|        | BDRM(4) = BDIN(4) - BDUU(4)                          | TREA631 |
|        | BCUU(4) = BDUU(4)/(QQU(4)*0.646*8.35*DT/1440.0)      | TREA632 |
|        | BCRM(4) = BDRM(4)/(0.10*NMS*0.646*8.35*DT/1440.0)    | TREA633 |
|        | GO TO 4900                                           | TREA634 |
| C      |                                                      | TREA635 |
| C      | 43) FOR HIGH RATE FILTERS                            | TREA636 |
| C      |                                                      | TREA637 |
|        | 4300 CALL HIGHRF                                     | TREA638 |
| C      |                                                      | TREA639 |
|        | 4900 CALL TRLINK                                     | TREA640 |
| C      |                                                      | TREA641 |
| C----- | LEVEL 5 BRANCH FOR EFFLUENT SCREENS                  | TREA642 |
| C      |                                                      | TREA643 |
|        | 5000 K = ITREAT(5) - 50                              | TREA644 |
|        | L = 5                                                | TREA645 |
|        | BDINT(5) = BDINT(5) + BDIN(5)                        | TREA646 |
|        | SSINT(5) = SSINT(5) + SSIN(5)                        | TREA647 |
|        | WAIN(5) = WAIN(5) + WAIN(5)                          | TREA648 |
|        | GO TO (5100,5200), K                                 | TREA649 |
| C      |                                                      | TREA650 |
| C      | 51) FOR NO EFFLUENT SCREENS (BYPASS)                 | TREA651 |
| C      |                                                      | TREA652 |
|        | 5100 CALL BYPASS                                     | TREA653 |
|        | GO TO 5900                                           | TREA654 |
| C      |                                                      | TREA655 |
| C      | 52) FOR EFFLUENT SCREENS                             | TREA656 |
| C      |                                                      | TREA657 |
| C      | SCREENINGS VOLUME AT 0.05 CU.FT/M.G.                 | TREA658 |
| C      |                                                      | TREA659 |
|        | 5200 WARM(5) = 0.00000374*WAIN(5)                    | TREA660 |

|                                                        |         |
|--------------------------------------------------------|---------|
| SSRM(5) = WARM(5)*50.0*0.15                            | TREA661 |
| BDRM(5) = 0.05*SSRM(5)                                 | TREA662 |
| QCRM(5) = WARM(5)/(DT*60.0)                            | TREA663 |
| WAUU(5) = WAIN(5) - WARM(5)                            | TREA664 |
| BDOU(5) = BDIN(5) - BDRM(5)                            | TREA665 |
| SSOU(5) = SSIN(5) - SSRM(5)                            | TREA666 |
| QQOU(5) = QQIN(5) - QCRM(5)                            | TREA667 |
| 5220 IF (QQOU(5) .EQ. 0.0) GO TO 5230                  | TREA668 |
| BCOU(5) = BDOU(5)*CCNVER/QQOU(5)                       | TREA669 |
| SCOU(5) = SSOU(5)*CCNVER/QQOU(5)                       | TREA670 |
| GO TO 5240                                             | TREA671 |
| 5230 BCOU(5) = 0.0                                     | TREA672 |
| SCOU(5) = 0.0                                          | TREA673 |
| 5240 IF (QCRM(5) .EQ. 0.0) GO TO 5250                  | TREA674 |
| BCRM(5) = BDRM(5)*CCNVER/QCRM(5)                       | TREA675 |
| SCRM(5) = SSRM(5)*CCNVER/QCRM(5)                       | TREA676 |
| GO TO 5900                                             | TREA677 |
| 5250 BCRM(5) = 0.0                                     | TREA678 |
| SCRM(5) = 0.0                                          | TREA679 |
| C                                                      | TREA680 |
| C                                                      | TREA681 |
| 5900 CALL TRLINK                                       | TREA682 |
| C                                                      | TREA683 |
| C----- LEVEL 6 BRANCH (FOR OUTLET PUMPING)             | TREA684 |
| C                                                      | TREA685 |
| 6000 K = ITREAT(6) - 60                                | TREA686 |
| L = 6                                                  | TREA687 |
| BDINT(6) = BDINT(6) + BDIN(6)                          | TREA688 |
| SSINT(6) = SSINT(6) + SSIN(6)                          | TREA689 |
| WAIN(6) = WAIN(6) + WAIN(6)                            | TREA690 |
| GO TO (6100,6200), K                                   | TREA691 |
| C                                                      | TREA692 |
| C                                                      | TREA693 |
| C                                                      | TREA694 |
| 6100 CALL BYPASS                                       | TREA695 |
| GO TO 6900                                             | TREA696 |
| C                                                      | TREA697 |
| C                                                      | TREA698 |
| C                                                      | TREA699 |
| 6200 CALL BYPASS                                       | TREA700 |
| C                                                      | TREA701 |
| 6900 CALL TRLINK                                       | TREA702 |
| C                                                      | TREA703 |
| C----- LEVEL 7 BRANCH (FOR CHLORINE CONTACT TIME)      | TREA704 |
| C                                                      | TREA705 |
| 7000 K = ITREAT(7) - 70                                | TREA706 |
| L = 7                                                  | TREA707 |
| BDINT(7) = BDINT(7) + BDIN(7)                          | TREA708 |
| SSINT(7) = SSINT(7) + SSIN(7)                          | TREA709 |
| WAIN(7) = WAIN(7) + WAIN(7)                            | TREA710 |
| GO TO (7100,7200), K                                   | TREA711 |
| C                                                      | TREA712 |
| C                                                      | TREA713 |
| C                                                      | TREA714 |
| 7100 IF (ITREAT(3) .EQ. 32 .OR. ITREAT(3) .EQ. 33 .OR. | TREA715 |
| * ITREAT(3) .EQ. 35) GO TO 7110                        | TREA716 |
| GO TO 7120                                             | TREA717 |
| 7110 IF (ICL2 .GT. 0) GO TO 7150                       | TREA718 |
| C IF NO CHLORINATION ELSEWHERE, REDUCE COLIFORMS BY 55 | TREA719 |
| 7120 IF (SCIF .EQ. 0.0) GO TO 7130                     | TREA720 |

|                                                  |         |
|--------------------------------------------------|---------|
| CCOU = CCIN*SCIN(7)/SCIF                         | TREA721 |
| GO TO 7150                                       | TREA722 |
| 7130 CCOU = CCIN                                 | TREA723 |
| C                                                | TREA724 |
| 7150 CALL BYPASS                                 | TREA725 |
| GO TO 7900                                       | TREA726 |
| U                                                | TREA727 |
| C                                                | TREA728 |
| C                                                | TREA729 |
| 7200 CL2DEM = BCIN(7)/10.0                       | TREA730 |
| IF (CL2DEM .LT. 6.0) CL2DEM = 6.0                | TREA731 |
| IF (CL2DEM .GT. 25.0) CL2DEM = 25.0              | TREA732 |
| PCL2DM = CL2DEM*QQIN(7)*0.646*8.35               | TREA733 |
| CL2UC = PCL2DM*DT/1440.0                         | TREA734 |
| CL2UT(7) = CL2UT(7) + CL2UC                      | TREA735 |
| C                                                | TREA736 |
| BCREDU = 2.0*CL2DEM                              | TREA737 |
| IF (BCREDU .GT. BCIN(7)) BCREDU = 0.50*BCIN(7)   | TREA738 |
| BCOU(7) = BCIN(7) - BCREDU                       | TREA739 |
| QQRN(7) = 0.0                                    | TREA740 |
| QQOU(7) = QQIN(7)                                | TREA741 |
| WARM(7) = 0.0                                    | TREA742 |
| WAOU(7) = WAIN(7)                                | TREA743 |
| SSOU(7) = SSIN(7)                                | TREA744 |
| SSRM(7) = 0.0                                    | TREA745 |
| SCOU(7) = SCIN(7)                                | TREA746 |
| SCRM(7) = 0.0                                    | TREA747 |
| BDOU(7) = WAOU(7)*BCOU(7)/16050.0                | TREA748 |
| BDRN(7) = BDIN(7) - BDOU(7)                      | TREA749 |
| BCRM(7) = 0.0                                    | TREA750 |
| C                                                | TREA751 |
| CALL KILL(VOLCON,SCIF,SCIN(7),QQIN(7),CCIN,CCOU) | TREA752 |
| C                                                | TREA753 |
| 7900 CALL TRLINK                                 | TREA754 |
| C                                                | TREA755 |
| GO TO 8000                                       | TREA756 |
| 9998 QQOU(7)=0.0                                 | TREA757 |
| SCRM(7)=0.0                                      | TREA758 |
| BCRM(7)=0.0                                      | TREA759 |
| SCOU(7)=0.0                                      | TREA760 |
| BCOU(7)=0.0                                      | TREA761 |
| WAOU(7)=0.0                                      | TREA762 |
| QQRN(7)=0.0                                      | TREA763 |
| WARM(7)=0.0                                      | TREA764 |
| SSOU(7)=0.0                                      | TREA765 |
| SSRM(7)=0.0                                      | TREA766 |
| BDOU(7)=0.0                                      | TREA767 |
| BDRN(7)=0.0                                      | TREA768 |
| CCOU=0.0                                         | TREA769 |
| C-----                                           | TREA770 |
| C                                                | TREA771 |
| 8000 QQRL = CCCU(7) + QQOF                       | TREA772 |
| BDRL = BDOU(7) + BDOF                            | TREA773 |
| SSRL = SSOU(7) + SSOF                            | TREA774 |
| WARL = WAOU(7) + WAOF                            | TREA775 |
| COU = CCCU*DT*60.0*QQOU(7)*28.3*10.0             | TREA776 |
| CORL = CCU + COOF                                | TREA777 |
| IF (QQRL .LE. 0.0) GO TO 8100                    | TREA778 |
| BCRL = BDRL*CCNVER/CCRL                          | TREA779 |
| SCRL = SSRL*CCNVER/CCRL                          | TREA780 |



|                                         |                                                              |
|-----------------------------------------|--------------------------------------------------------------|
| CCRL = CORL/(DT*60.0*QQR*28.32*10.0)    | TREA781                                                      |
| C??                                     | CHLORINATE OVERFLOW ?? TREA782                               |
| GO TO 8200                              | TREA783                                                      |
| 8100 BCRL = 0.0                         | TREA784                                                      |
| SCRL = 0.0                              | TREA785                                                      |
| CCRL = 0.0                              | TREA786                                                      |
| C                                       | TREA787                                                      |
| 8200 WARLT = WARLT + WARL               | TREA788                                                      |
| BDRLT = BDRLT + BDRL                    | TREA789                                                      |
| SSRLT = SSRLT + SSRL                    | TREA790                                                      |
| CORLT = CCRLT + CORL                    | TREA791                                                      |
| C                                       | TREA792                                                      |
| CORM = COIF - CORL                      | TREA793                                                      |
| CORMT = CORMT + CORM                    | TREA794                                                      |
| COOUT = COOUT + COOU                    | TREA795                                                      |
| CCOUT = CCOUT + CCOU                    | TREA796                                                      |
| 8240 IF (IRANGE .EQ. 0) GO TO 8300      | TREA797                                                      |
| IF (QQR .GT. QQRMX) QQRMX = QQR         | TREA798                                                      |
| IF (BCRL .GT. BCRLMX) BCRLMX = BCRL     | TREA799                                                      |
| IF (SCRL .GT. SCRLMX) SCRLMX = SCRL     | TREA800                                                      |
| IF (CCRL .GT. CCRLMX) CCRLMX = CCRL     | TREA801                                                      |
| IF (QQR .LT. QQRMIN) QQRMIN = QQR       | TREA802                                                      |
| IF (BCRL .LT. BCRLMIN) BCRLMIN = BCRL   | TREA803                                                      |
| IF (SCRL .LT. SCRLMIN) SCRLMIN = SCRL   | TREA804                                                      |
| IF (CCRL .LT. CCRLMIN) CCRLMIN = CCRL   | TREA805                                                      |
| IF (CCOU .GT. CCOUTMX) CCOUTMX = CCOU   | TREA806                                                      |
| IF (CCOU .LT. CCOUTMIN) CCOUTMIN = CCOU | TREA807                                                      |
| QQRLT = QQRLT + QQR                     | TREA808                                                      |
| BCRLT = BCRLT + BCRL                    | TREA809                                                      |
| SCRLT = SCRLT + SCRL                    | TREA810                                                      |
| CCRLT = CCRLT + CCRL                    | TREA811                                                      |
| C                                       | TREA812                                                      |
| C                                       | COMPUTE POLLUTION REDUCTION AT 11 SELECTED TIMESTEPS TREA813 |
| C                                       | TREA814                                                      |
| 8300 IF (KDT .EQ. NM(M)) GO TO 8400     | TREA815                                                      |
| GO TO 9000                              | TREA816                                                      |
| 8400 MHDUR(M) = KHDUR                   | TREA817                                                      |
| MMIN(M) = KMIN                          | TREA818                                                      |
| C                                       | WATER TREA819                                                |
| QAV(M) = 0.5*(QQIF+QQR)                 | TREA820                                                      |
| C                                       | BCD TREA821                                                  |
| BDCIF(M) = BCIF                         | TREA822                                                      |
| BDCRL(M) = BCRL                         | TREA823                                                      |
| IF (BDIF .LE. 0.0) GO TO 8500           | TREA824                                                      |
| BDRD(M) = 100.0*(BDIF - BDRL)/BDIF      | TREA825                                                      |
| GO TO 8600                              | TREA826                                                      |
| 8500 BDRD(M) = 0.0                      | TREA827                                                      |
| C                                       | SUSP. SOLIDS TREA828                                         |
| 8600 SSCIF(M) = SCIF                    | TREA829                                                      |
| SSCRL(M) = SCRL                         | TREA830                                                      |
| IF (SSIF .LE. 0.0) GO TO 8700           | TREA831                                                      |
| SSRD(M) = 100.0*(SSIF - SSRL)/SSIF      | TREA832                                                      |
| GO TO 8800                              | TREA833                                                      |
| 8700 SSRD(M) = 0.0                      | TREA834                                                      |
| C                                       | CCIFORMS TREA835                                             |
| 8800 CQCIF(M) = CCIF                    | TREA836                                                      |
| CQCRL(M) = CCRL                         | TREA837                                                      |
| IF (COIF .LE. 0.0) GO TO 8900           | TREA838                                                      |
| CURD(M) = 100.0*(COIF-CORL)/COIF        | TREA839                                                      |
| GO TO 8999                              | TREA840                                                      |

|      |                                                                  |         |
|------|------------------------------------------------------------------|---------|
| 8900 | CORD(M) = 0.0                                                    | TREA841 |
| 8999 | IF (M .LT. 11) M = M + 1                                         | TREA842 |
| C    |                                                                  | TREA843 |
| C    | ----- PRINT SOLUTION FOR THIS TIME-STEP -----                    | TREA844 |
| C    |                                                                  | TREA845 |
|      | 9000 IF (IPRINT .EQ. 0) GO TO 9500                               | TREA846 |
|      | GO TO (9100,9200),IPRINT                                         | TREA847 |
| C    |                                                                  | TREA848 |
| C    | A) ON A QUANTITY BASIS (IPRINT = 1)                              | TREA849 |
| C    |                                                                  | TREA850 |
|      | 9100 WRITE(6,691) KFOUR,KMIN,WAIF,BDIF,SSIF,                     | TREA851 |
|      | * COIF,WAOU(3),BDOU(3),SSOU(3),                                  | TREA852 |
|      | * WAOU(4),BDOU(4),SSOU(4),                                       | TREA853 |
|      | * WAOU(7),BDOU(7),COOU,                                          | TREA854 |
|      | * WARL,BDRL,SSRL,CURL,                                           | TREA855 |
|      | * WAOF,BDOF,SSOF,COOF,                                           | TREA856 |
|      | * WARM(3),BDRM(3),SSRM(3),                                       | TREA857 |
|      | * WARM(4),BDRM(4),SSRM(4),                                       | TREA858 |
|      | * WARM(7),BDRM(7),CORM                                           | TREA859 |
|      | 691 FORMAT('0', I2, ':', I2, ' ARR', F7.0, 2F6.1, E10.2, ' OUT', | TREA860 |
|      | * 2(F7.0, 2F6.1), F7.0, F6.1, E10.2, F7.0, 2F6.1, E10.2, /,      | TREA861 |
|      | * ' ', 5X, ' OVF', F7.0, 2F6.1, E10.2, ' REM', 2(F7.0, 2F6.1),   | TREA862 |
|      | * F7.0, F6.1, E10.2)                                             | TREA863 |
|      | GO TO 9500                                                       | TREA864 |
| C    |                                                                  | TREA865 |
| C    | B) ON A CONCENTRATION BASIS (IPRINT = 2)                         | TREA866 |
| C    |                                                                  | TREA867 |
|      | 9200 WRITE(6,692) KFOUR,KMIN,QQIF,BCIF,SCIF,CCIF,                | TREA868 |
|      | * QCOU(3),BCOU(3),SCOU(3),                                       | TREA869 |
|      | * QCOU(4),BCOU(4),SCOU(4),                                       | TREA870 |
|      | * QCOU(7),BCOU(7),CCOU,                                          | TREA871 |
|      | * QQRL,BCRL,SCRL,CCRL,                                           | TREA872 |
|      | * QQOF,BCCF,SCOF,CCCF,                                           | TREA873 |
|      | * QQRM(3),BCRM(3),SCRM(3),                                       | TREA874 |
|      | * QQRM(4),BCRM(4),SCRM(4),                                       | TREA875 |
|      | * QQRM(7),BCRM(7)                                                | TREA876 |
|      | 692 FORMAT('0', I2, ':', I2, ' ARR', F7.2, 2F6.0, E10.2, ' OJT', | TREA877 |
|      | * 2(F7.2, 2F6.0), F7.2, F6.0, E10.2, F7.2, 2F6.0, E10.2, /,      | TREA878 |
|      | * ' ', 5X, ' OVF', F7.2, 2F6.0, E10.2, ' REM', 2(F7.2, 2F6.0),   | TREA879 |
|      | * F7.2, F6.0)                                                    | TREA880 |
|      | 9500 CONTINUE                                                    | TREA881 |
|      | IA=0                                                             | TREA882 |
|      | IF (KOT .LT. NOT) GO TO 9999                                     | TREA883 |
| C    |                                                                  | TREA884 |
| C    | ----- SUMMARIZE -----                                            | TREA885 |
| C    |                                                                  | TREA886 |
|      | IF (KNTOF .LT.1) KNTOF = 10**12                                  | TREA887 |
|      | WAIFT = WAIFT*7.48/1000000.0                                     | TREA888 |
|      | WAUFT = WAUFT*7.48/1000000.0                                     | TREA889 |
|      | WARMTT = 0.0                                                     | TREA890 |
|      | BDRMTT = 0.0                                                     | TREA891 |
|      | SSRMTT = 0.0                                                     | TREA892 |
|      | DO 9600 L=1,7                                                    | TREA893 |
|      | WAINT(L) = WAINT(L)*7.48/1000000.0                               | TREA894 |
|      | WARMT(L) = WARMT(L)*7.48/1000000.0                               | TREA895 |
|      | WARMTT = WARMTT + WARMT(L)                                       | TREA896 |
|      | BDRMTT = BDRMTT + BDRMT(L)                                       | TREA897 |
| 9600 | SSRMTT = SSRMTT + SSRMT(L)                                       | TREA898 |
|      | WARLT = WARLT*7.48/1000000.0                                     | TREA899 |
|      | WTRMT1 = WARMT(1)*1000000.0/7.48                                 | TREA900 |

```

WTRMT5 = WARM(5)*1000000.0/7.48
WAIFRF = 100.0*WARM(1)/WAIFT
BDIFRF = 100.0*BDRMT(1)/BDIFT
SSIFRF = 100.0*SSRMT(1)/SSIFT
COIFRF = 100.0*CORMT(1)/COIFT
WAINRF = 100.0*WARM(1)/WAIN(1)
BDINRF = 100.0*BDRMT(1)/BDINT(1)
SSINRF = 100.0*SSRMT(1)/SSINT(1)
COINRF = 100.0*CORMT(1)/COINT
CL2UTT = 0.0
CHEMTT = 0.0
DO 9700 L=1,8
CL2UTT = CL2UTT + CL2UT(L)
9700 CHEMTT = CHEMTT + CHEMT(L)
WRITE(6,697) WAIFT,BDIFT,SSIFT,COIFT,WAOFT,BDOFT,SSOFT,COOFT,
* WAINT(1),BDINT(1),SSINT(1),COINT,
* WARM(1),BDRMT(1),SSRMT(1),CORMT(1),WAKLT,BDRLT,SSRLT,CORLT,
* WARM(3),BDRMT(3),SSRMT(3),(NAME(I,J1), I=1,4),
697 FORMAT(' SUMMARY OF TREATMENT EFFECTIVENESS', //,
* '0', 20X, 'TOTALS', 15X,
* 'FLOW (M.G.) BOD (LB) SS (LB) COLIF (MPN)', /,
* ' ', 22X, 'INPUT', 15X, F10.3, 2F12.1, 1PE12.2, /,
* ' ', 22X, 'OVERFLOW (BYPASS) ', OPF10.3, 2F12.1, 1PE12.2, /,
* ' ', 22X, 'TREATED', 13X, OPF10.3, 2F12.1, 1PE12.2, /,
* ' ', 22X, 'REMOVED', 13X, OPF10.3, 2F12.1, 1PE12.2, /,
* ' ', 22X, 'RELEASED', 12X, OPF10.3, 2F12.1, 1PE12.2, //,
* 'C', 20X, 'REMOVALS', 14X,
* 'FLOW (M.G.) BOD (LB) SS (LB)', /,
* ' ', 22X, 'LEVEL 1', 11X, OPF12.3, 2F12.1, 18X, '= ', 4A4, /,
* ' ', 22X, 'LEVEL 3 (TOTAL) ', F12.3, 2F12.1, 18X, '= ', 4A4, /,
IF (ITREAT(3) .NE. 33) GO TO 9800
WARST = WARST*7.48/1000000.0
WRITE(6,695) WARST, BDRST, SSRST
695 FORMAT(' ', 22X, 'FINE SCREENS (33) (', F11.3, 2(')(', F10.1), ')',
* 17X, '= ( FINE SCREENS ) ')
9800 WRITE(6,696) WARM(4),BDRMT(4),SSRMT(4),(NAME(I,J4), I=1,4),
* WARM(5),BDRMT(5),SSRMT(5),(NAME(I,J5), I=1,4),
* WARM(7),BDRMT(7),SSRMT(7),(NAME(I,J7), I=1,4),
* WTRMT1,WTRMT5,
* WAIFRF,BDIFRF,SSIFRF,COIFRF,WAINRF,BDINRF,SSINRF,
* CCINRF,CL2UT(3),CHEMT(3),(NAME(I,J3), I=1,4),
* CL2UT(4),CHEMT(4),(NAME(I,J4), I=1,4),
* CL2UT(7),CHEMT(7),(NAME(I,J7), I=1,4),
* CL2UTT,CHEMTT
696 FORMAT(' ', 22X, 'LEVEL 4', 11X, F12.3, 2F12.1, 18X, '= ', 4A4, /,
* ' ', 22X, 'LEVEL 5', 11X, F12.3, 2F12.1, 18X, '= ', 4A4, /,
* ' ', 22X, 'LEVEL 7', 11X, F12.3, 2F12.1, 18X, '= ', 4A4, /,
* ' ', 24X, 'TRASH:', /,
* ' ', 22X, 'BAR RACKS', 9X, F12.3,
* ' CU.FT (AT 50 LB/CU.FT.)', /,
* ' ', 22X, 'EFFLUENT SCREENS', 2X, F12.3,
* ' CU.FT (AT 50 LB/CU.FT.)', //,
* '0', 20X, 'REMOVAL PERCENTAGES', 2X,
* ' FLOW (VOL) BOD (LB) SS (LB) COLIF (MPN)', /,
* ' ', 22X, 'OF OVERALL INPUTS', 1X, 4F12.2, /,
* ' ', 22X, 'OF TREATED FRACTIONS', F10.2, 3F12.2, //,
* '0', 20X, 'CONSUMPTIONS (LB)', 7X, 'CHLORINE', 4X,
* ' POLYMERS', /,
* ' ', 22X, 'LEVEL 3', 13X, F10.1, F12.1, 30X, '= ', 4A4, /,

```

```

*      ' ', 22X, 'LEVEL 4', 13X, F10.1, F12.1, 30X, '= ', 4A4, '/', TREA961
*      ' ', 22X, 'LEVEL 7', 13X, F10.1, F12.1, 30X, '= ', 4A4, '/', TREA962
*      ' ', 22X, 'TOTAL', 15X, F10.1, F12.1) TREA963
WRITE(6,693) (MHOUR(M),MMIN(M), M=1,11), (QAV(M), M=1,11), TREA964
*      (BDCIF(M), M=1,11), TREA965
*      (BDCRL(M), M=1,11), (BDRD(M), M=1,11), TREA966
*      (SSCIF(M), M=1,11), (SSCRL(M), M=1,11), TREA967
*      (SSRD(M), M=1,11), (COCIF(M), M=1,11), TREA968
*      (COCRL(M), M=1,11), (CORD(M), M=1,11) TREA969
693 FORMAT('0', 15X, 'REPRESENTATIVE VARIATION OF TREATMENT PERFORM', TREA970
*      'ANCE WITH TIME (OVERALL).', /, TREA971
*      '0', 11X, 'TIME', 15X, 11(4X, 12, ':', 12), /, TREA972
*      ' ', 11X, 'WATER', /, TREA973
*      ' ', 14X, 'AV. FLOW (CFS) ', 11F9.2, /, TREA974
*      ' ', 11X, 'BOD', /, TREA975
*      ' ', 14X, 'ARRIVING (MG/L) ', 11F9.2, /, TREA976
*      ' ', 14X, 'RELEASED (MG/L) ', 11F9.2, /, TREA977
*      ' ', 14X, '% REDUCTION (LB)', 11F9.2, /, TREA978
*      ' ', 11X, 'S. SOLIDS ', /, TREA979
*      ' ', 14X, 'ARRIVING (MG/L) ', 11F9.2, /, TREA980
*      ' ', 14X, 'RELEASED (MG/L) ', 11F9.2, /, TREA981
*      ' ', 14X, '% REDUCTION (LB)', 11F9.2, /, TREA982
*      ' ', 11X, 'COLIFORMS ', /, TREA983
*      ' ', 14X, 'ARR (MPN/100ML) ', 11F9.2, /, TREA984
*      ' ', 14X, 'REL (MPN/100ML) ', 11F9.2, /, TREA985
*      ' ', 14X, '% REDUCTION (LB)', 0P11F9.2) TREA986
IF (IRANGE .EQ. 0) GO TO 9999 TREA987
QQIFMX = QQIFMX/1.547 TREA988
QQQFMX = QQQFMX/1.547 TREA989
QQINMX = QQINMX/1.547 TREA990
QQRLMX = QQRLMX/1.547 TREA991
QQIFT = QQIFT/(1.547*NDT) TREA992
QQOFT = QQOFT/(1.547*KNTOF) TREA993
QQINT = QQINT/(1.547*NDT) TREA994
QQRLT = QQRLT/(1.547*NDT) TREA995
QQIFMN = QQIFMN/1.547 TREA996
QQQFMN = QQQFMN/1.547 TREA997
QQINMN = QQINMN/1.547 TREA998
QQRLMN = QQRLMN/1.547 TREA999
QQRMMX(3) = QQRMMX(3)/1.547 TREA001
QQRMMX(4) = QQRMMX(4)/1.547 TREA002
QQRMMX(7) = QQRMMX(7)/1.547 TREA003
QQOUMX(3) = QQOUMX(3)/1.547 TREA004
QQOUMX(4) = QQOUMX(4)/1.547 TREA005
QQOUMX(7) = QQOUMX(7)/1.547 TREA006
QQRMT(3) = QQRMT(3)/(1.547*NDT) TREA007
QQRMT(4) = QQRMT(4)/(1.547*NDT) TREA008
QQRMT(7) = QQRMT(7)/(1.547*NDT) TREA009
QQOUT(3) = QQOUT(3)/(1.547*NDT) TREA010
QQOUT(4) = QQOUT(4)/(1.547*NDT) TREA011
QQOUT(7) = QQOUT(7)/(1.547*NDT) TREA012
QQRMMN(3) = QQRMMN(3)/1.547 TREA013
QQRMMN(4) = QQRMMN(4)/1.547 TREA014
QQRMMN(7) = QQRMMN(7)/1.547 TREA015
QQOUMN(3) = QQOUMN(3)/1.547 TREA016
QQOUMN(4) = QQOUMN(4)/1.547 TREA017
QQOUMN(7) = QQOUMN(7)/1.547 TREA018
C BCIFT = BCIFT/NDT TREA019
BCUFI = BCUFI/KNTOF TREA020
TREA021

```

|                                                                   |         |
|-------------------------------------------------------------------|---------|
| BCINT = BCINI/NDT                                                 | TREA022 |
| BCRLT = BCRLT/NDT                                                 | TREA023 |
| SCIFT = SCIFT/NDT                                                 | TREA024 |
| SCOFT = SCOFT/KNTOF                                               | TREA025 |
| SCINT = SCINT/NDT                                                 | TREA026 |
| SCRLT = SCRLT/NDT                                                 | TREA027 |
| CCIFT = CCIFT/NDT                                                 | TREA028 |
| CCOFT = CCOFT/KNTOF                                               | TREA029 |
| CCINT = CCINT/NDT                                                 | TREA030 |
| CCRLT = CCRLT/NDT                                                 | TREA031 |
| BCRMT(3) = BCRMT(3)/NDT                                           | TREA032 |
| BCRMT(4) = BCRMT(4)/NDT                                           | TREA033 |
| BCRMT(7) = BCRMT(7)/NDT                                           | TREA034 |
| SCRMT(3) = SCRMT(3)/NDT                                           | TREA035 |
| SCRMT(4) = SCRMT(4)/NDT                                           | TREA036 |
| SCRMT(7) = SCRMT(7)/NDT                                           | TREA037 |
| BCCUT(3) = BCCUT(3)/NDT                                           | TREA038 |
| BCCUT(4) = BCCUT(4)/NDT                                           | TREA039 |
| BCCUT(7) = BCCUT(7)/NDT                                           | TREA040 |
| SCOUT(3) = SCOUT(3)/NDT                                           | TREA041 |
| SCOUT(4) = SCOUT(4)/NDT                                           | TREA042 |
| SCOUT(7) = SCOUT(7)/NDT                                           | TREA043 |
| CCOUT = CCOUT/NDT                                                 | TREA044 |
| C                                                                 | TREA045 |
| WRITE(6,698) QQIFMX,QQOFMX,QQINMX,QQRMMX(3),QQUUMX(3),QQRMMX(4),  | TREA046 |
| * QQOUMX(4),QQRMMX(7),QQOUMX(7),QQRMLX,                           | TREA047 |
| * QQIFT,QQOFT,QQINT,QQRMT(3),QQOUT(3),QQRMT(4),                   | TREA048 |
| * CCOUT(4),QQRMT(7),QQOUT(7),QQRLT,                               | TREA049 |
| * QCIFMN,CCOFMN,QQINMN,QQRMMN(3),QQOUMN(3),QQRMMN(4),             | TREA050 |
| * QQOUMN(4),QQRMMN(7),QQOUMN(7),QQRMLN,                           | TREA051 |
| * BCIFMX,BCOFMX,BCINMX,BCRMMX(3),BCOUMX(3),BCRMMX(4),             | TREA052 |
| * BCOUMX(4),BCRMMX(7),BCOUMX(7),BCRLMX,                           | TREA053 |
| * BCIFT,BCCFT,BCINT,BCRMT(3),BCOUT(3),BCRMT(4),                   | TREA054 |
| * BCOUT(4),BCRMT(7),BCOUT(7),BCRLT,                               | TREA055 |
| * BCIFMN,BCOFMN,BCINMN,BCRMMN(3),BCOUMN(3),BCRMMN(4),             | TREA056 |
| * BCOUMN(4),BCRMMN(7),BCOUMN(7),BCRLMN                            | TREA057 |
| WRITE(6,694) SCIFMX,SCOFMX,SCINMX,SCRMMX(3),SCOUMX(3),SCRMMX(4),  | TREA058 |
| * SCOUMX(4),SCRMMX(7),SCOUMX(7),SCRMLX,                           | TREA059 |
| * SCIFT,SCOFT,SCINT,SCRMT(3),SCOUT(3),SCRMT(4),                   | TREA060 |
| * SCOUT(4),SCRMT(7),SCOUT(7),SCRLT,                               | TREA061 |
| * SCIFMN,SCOFMN,SCINMN,SCRMMN(3),SCOUMN(3),SCRMMN(4),             | TREA062 |
| * SCOUMN(4),SCRMMN(7),SCOUMN(7),SCRMLN,                           | TREA063 |
| * CCIFMX,CCOFMX,CCINMX,CCOUMX,CCRLMX,                             | TREA064 |
| * CCIFT,CCOFT,CCINT,CCOUT,CCRLT                                   | TREA065 |
| IF (CCOFMN .LE. 0.0) WRITE(6,688) CCIFMN,CCOFMN,CCINMN,CCOUMN,    | TREA066 |
| * CCRLMN                                                          | TREA067 |
| 688 FORMAT(' MINIMUM ', 1PE12.2, 0PE12.2, 1PE12.2, 60X, 2E12.2)   | TREA068 |
| IF (CCOFMN .GT. 0.0) WRITE(6,689) CCIFMN,CCOFMN,CCINMN,CCOUMN,    | TREA069 |
| * CCRLMN                                                          | TREA070 |
| 689 FORMAT(' MINIMUM ', 3E12.2, 60X, 2E12.2)                      | TREA071 |
| 698 FORMAT('1', 15X, 'SUMMARY OF FLOWS - MAXIMA, AVERAGES, AND ', | TREA072 |
| * 'MINIMA', //,                                                   | TREA073 |
| * '0', 39X, 'TO', 15X, 'LEVEL 3', 17X, 'LEVEL 4', 17X, 'LEVEL 7', | TREA074 |
| * 8X, 'RECOMBINED', /,                                            | TREA075 |
| * ' ', 13X, 'ARRIVING OVERFLOW TREATMENT REMOVAL',                | TREA076 |
| * 5X, 'OUTFLOW REMOVAL OUTFLOW REMOVAL',                          | TREA077 |
| * 5X, 'OUTFLOW RELEASE', //,                                      | TREA078 |
| * '0', 5X, 'FLOW RATES (M.G.D.)', /,                              | TREA079 |
| * '0MAXIMUM ', 1CF12.3, /,                                        | TREA080 |
| * ' AVERAGE ', 1CF12.3, /,                                        | TREA081 |

|      |                                                              |         |
|------|--------------------------------------------------------------|---------|
| *    | ' MINIMUM ', 10F12.3, ///,                                   | TREA082 |
| *    | 'C', 5X, 'BOD CONCENTRATIONS (MG/L)', /,                     | TREA083 |
| *    | 'OMAXIMUM ', 10F12.1, /,                                     | TREA084 |
| *    | ' AVERAGE ', 10F12.1, /,                                     | TREA085 |
| *    | ' MINIMUM ', 10F12.1, ///                                    | TREA085 |
| 694  | FORMAT('C', 5X, 'SUSPENDED SOLIDS CONCENTRATIONS (MG/L)', /, | TREA087 |
| *    | 'CHAXIMUM ', 10F12.1, /,                                     | TREA088 |
| *    | ' AVERAGE ', 10F12.1, /,                                     | TREA089 |
| *    | ' MINIMUM ', 10F12.1, ///,                                   | TREA090 |
| *    | 'O', 5X, 'COLIFORM CONCENTRATIONS (MPN/100ML)', /,           | TREA091 |
| *    | 'CHAXIMUM ', 1P3E12.2, 60X, 2E12.2, /,                       | TREA092 |
| *    | ' AVERAGE ', 3E12.2, 60X, 2E12.2)                            | TREA093 |
|      | QQIFMX = QQIFMX*1.547                                        | TREA094 |
| C    |                                                              | TREA095 |
| 9999 | RETURN                                                       | TREA096 |
|      | END                                                          | TREA097 |
| C    | =====                                                        | TREA098 |

|                                                                   |        |    |
|-------------------------------------------------------------------|--------|----|
| SUBROUTINE BYPASS                                                 | BYPASS | 1  |
| COMMON CCONVER,KHCUR,KMIN,L,KMOD,NFLAG,BIG,HEAD1,HEAD2,           | BYPASS | 2  |
| * QDESYN,QQIF,WAOU(7),WAIN(7),QQOU(7),QQIN(7),WARM(7),QQRL,       | BYPASS | 3  |
| * BDIF,QQRM(7),BDOU(7),BDIN(7),BCOU(7),BCIN(7),BDRL,              | BYPASS | 4  |
| * SSIF,SSIN(7),SSOU(7),SCOU(7),SCIN(7),SSRM(7),SSRL,COIF,CORL,    | BYPASS | 5  |
| * ADEPTH(11),AASURF(11),ITREAT(7),ISTOR,IPRINT,ICOST,HREFD,       | BYPASS | 6  |
| * MODSIZ,ICHEM,ICL2,SCREEN,QQIFMX,DESF,IRANGE,KNTUF,TRIBA,SEDA,   | BYPASS | 7  |
| * SQM,SREFFH,BREFFH,NUNITH,UAREAH,OPRAMA,ICHEMH,HM,VOLDAF,ITABLE, | BYPASS | 8  |
| * MODCST,TOTCST,RECIRC,OVRDAF,TSURFA,OVRSED,NSED,JM(7),WTRMT5,    | BYPASS | 9  |
| * NSCRN,SCRCAP,SUAREA,FAREAB,NMS,AREAMS,VOLCON,VOLSED,ALJMT,      | BYPASS | 10 |
| * BDINT(7),SSINT(7),WAIN(7),BDOU(7),SSOU(7),WAOUT(7),             | BYPASS | 11 |
| * WARM(7),BDRMT(7),SSRMT(7),CHEMUT(8),CL2UT(8),QMOD(20),WTPMT1,   | BYPASS | 12 |
| * QQRMT(7),QQOUT(7),QQRMMX(7),QQOUMX(7),QQRMMN(7),QQOUMN(7),      | BYPASS | 13 |
| * BCRMT(7),BCOU(7),BCRMMX(7),BCOUMX(7),BCRMMN(7),BCOUMN(7),       | BYPASS | 14 |
| * SCRMT(7),SCOUT(7),SCRMMX(7),SCOUMX(7),SCRMMN(7),SCOUMN(7),      | BYPASS | 15 |
| * BDRM(7),BCRM(7),SCRM(7)                                         | BYPASS | 16 |
| C                                                                 | BYPASS | 17 |
| WAOU(L) = WAIN(L)                                                 | BYPASS | 18 |
| QQOU(L) = QQIN(L)                                                 | BYPASS | 19 |
| WARM(L) = 0.0                                                     | BYPASS | 20 |
| QQRM(L) = 0.0                                                     | BYPASS | 21 |
| BDOU(L) = BDIN(L)                                                 | BYPASS | 22 |
| BCOU(L) = BCIN(L)                                                 | BYPASS | 23 |
| BDRM(L) = 0.0                                                     | BYPASS | 24 |
| BCRM(L) = 0.0                                                     | BYPASS | 25 |
| SSOU(L) = SSIN(L)                                                 | BYPASS | 26 |
| SCOU(L) = SCIN(L)                                                 | BYPASS | 27 |
| SSRM(L) = 0.0                                                     | BYPASS | 28 |
| SCRM(L) = 0.0                                                     | BYPASS | 29 |
| C                                                                 | BYPASS | 30 |
| 9999 RETURN                                                       | BYPASS | 31 |
| END                                                               | BYPASS | 32 |
| C=====                                                            | BYPASS | 33 |

|                                                                   |         |
|-------------------------------------------------------------------|---------|
| SUBROUTINE TRLINK                                                 | TRLI 1  |
| COMMON CCNVER,KHOUR,KMIN,L,KMOD,NFLAG,BIG,HEAD1,HEAD2,            | TRLI 2  |
| * QDESYN,QQIF,WAOU(7),WAIN(7),QQOU(7),QQIN(7),WARM(7),QQRL,       | TRLI 3  |
| * BDIF,QCRM(7),BDOU(7),BDIN(7),BCOU(7),BCIN(7),BDRL,              | TRLI 4  |
| * SSIF,SSIN(7),SSOU(7),SCOU(7),SCIN(7),SSRM(7),SSRL,CUIF,CURL,    | TRLI 5  |
| * ADEPTH(11),AASURF(11),ITREAT(7),ISTOR,IPRINT,ICOST,HREFD,       | TRLI 6  |
| * MODSIZ,ICHEM,ICL2,SCREEN,QQIFMX,DESF,IRANGE,KNTOF,TRIBA,SEDA,   | TRLI 7  |
| * SQM,SREFFH,BREFFH,NUNITH,UAREAH,OPRAMA,ICHEMH,HM,VOLDAF,ITABLE, | TRLI 8  |
| * MODCST,TOTCST,RECIRC,OVRDAF,TSURFA,OVRSED,NSED,JM(7),WTRMT5,    | TRLI 9  |
| * NSCRN,SCRCAP,SUAREA,FAREAB,NMS,AREAMS,VOLCON,VULSED,ALJMUT,     | TRLI 10 |
| * BDINT(7),SSINT(7),WAIN(7),BDOUT(7),SSOUT(7),WAOUT(7),           | TRLI 11 |
| * WARM(7),BDRMT(7),SSRMT(7),CHEMUT(8),CLZUT(8),QMOD(20),WTRMT1,   | TRLI 12 |
| * QQRMT(7),QQOUT(7),QQRMMX(7),QQOUMX(7),QQRMMN(7),QQOUMN(7),      | TRLI 13 |
| * BCRMT(7),BCOUT(7),BCRMMX(7),BCOUMX(7),BCRMMN(7),BCOUMN(7),      | TRLI 14 |
| * SCRMT(7),SCOUT(7),SCRMMX(7),SCOUMX(7),SCRMMN(7),SCOUMN(7),      | TRLI 15 |
| * BDRM(7),BCRM(7),SCRM(7)                                         | TRLI 16 |
| C                                                                 | TRLI 17 |
| BDRMT(L) = BDRMT(L) + BDRM(L)                                     | TRLI 18 |
| SSRMT(L) = SSRMT(L) + SSRM(L)                                     | TRLI 19 |
| WARM(L) = WARM(L) + WARM(L)                                       | TRLI 20 |
| BDOUT(L) = BDOUT(L) + BDOU(L)                                     | TRLI 21 |
| SSOUT(L) = SSOUT(L) + SSOU(L)                                     | TRLI 22 |
| WAOUT(L) = WAOUT(L) + WAOU(L)                                     | TRLI 23 |
| IF (IRANGE.EQ. 0) GO TO 5000                                      | TRLI 24 |
| IF (QCRM(L).GT. QQRMMX(L)) QQRMMX(L) = QCRM(L)                    | TRLI 25 |
| IF (BCRM(L).GT. BCRMMX(L)) BCRMMX(L) = BCRM(L)                    | TRLI 26 |
| IF (SCRM(L).GT. SCRMMX(L)) SCRMMX(L) = SCRM(L)                    | TRLI 27 |
| IF (QQOU(L).GT. QQOUMX(L)) QQOUMX(L) = QQOU(L)                    | TRLI 29 |
| IF (BCOU(L).GT. BCOUMX(L)) BCOUMX(L) = BCOU(L)                    | TRLI 29 |
| IF (SCOU(L).GT. SCOUMX(L)) SCOUMX(L) = SCOU(L)                    | TRLI 30 |
| IF (QCRM(L).LT. QQRMMN(L)) QQRMMN(L) = QCRM(L)                    | TRLI 31 |
| IF (BCRM(L).LT. BCRMMN(L)) BCRMMN(L) = BCRM(L)                    | TRLI 32 |
| IF (SCRM(L).LT. SCRMMN(L)) SCRMMN(L) = SCRM(L)                    | TRLI 33 |
| IF (QQOU(L).LT. QQOUMN(L)) QQOUMN(L) = QQOU(L)                    | TRLI 34 |
| IF (BCOU(L).LT. BCOUMN(L)) BCOUMN(L) = BCOU(L)                    | TRLI 35 |
| IF (SCOU(L).LT. SCOUMN(L)) SCOUMN(L) = SCOU(L)                    | TRLI 35 |
| QQRMT(L) = QQRMT(L) + QCRM(L)                                     | TRLI 37 |
| BCRMT(L) = BCRMT(L) + BCRM(L)                                     | TRLI 38 |
| SCRMT(L) = SCRMT(L) + SCRM(L)                                     | TRLI 39 |
| QQOUT(L) = QQOUT(L) + QQOU(L)                                     | TRLI 40 |
| BCOUT(L) = BCOUT(L) + BCOU(L)                                     | TRLI 41 |
| SCOUT(L) = SCOUT(L) + SCOU(L)                                     | TRLI 42 |
| 5000 IF (L.EQ. 7) GO TO 9999                                      | TRLI 43 |
| C                                                                 | TRLI 44 |
| WAIN(L+1) = WAOU(L)                                               | TRLI 45 |
| QQIN(L+1) = QQOU(L)                                               | TRLI 46 |
| BDIN(L+1) = BDOU(L)                                               | TRLI 47 |
| BCIN(L+1) = BCOU(L)                                               | TRLI 48 |
| SSIN(L+1) = SSOU(L)                                               | TRLI 49 |
| SCIN(L+1) = SCOU(L)                                               | TRLI 50 |
| C                                                                 | TRLI 51 |
| 9999 RETURN                                                       | TRLI 52 |
| END                                                               | TRLI 53 |
| C=====                                                            | TRLI 54 |



|                                                                   |         |
|-------------------------------------------------------------------|---------|
| SUBROUTINE SEDI                                                   | SEDI 1  |
| COMMON CCNVER,KHOUR,KMIN,L,KMOD,NFLAG,BIG,HEAD1,HEAD2,            | SEDI 2  |
| * QDESYN,QQIF,WAOU(7),WAIN(7),QQUO(7),QQIN(7),WARM(7),QURL,       | SEDI 3  |
| * BDIF,QCRM(7),BDOU(7),BDIN(7),BCOU(7),BCIN(7),BDRL,              | SEDI 4  |
| * SSIF,SSIN(7),SSOU(7),SCOU(7),SCIN(7),SSRM(7),SSRL,COIF,CORL,    | SEDI 5  |
| * ADEPTH(11),AASURF(11),ITREAT(7),ISTOR,IPRINT,ICOST,HREF,        | SEDI 6  |
| * MDSIZ,ICHEM,ICL2,SCREEN,QQIFMX,DESF,IRANGE,KNTUF,TRIBA,SEDA,    | SEDI 7  |
| * SQM,SREFFH,BREFFH,NUNITH,UAREAH,OPRAMA,ICHEMH,HM,VOLDAF,ITABLE, | SEDI 8  |
| * MUDCST,TOTCST,RECIRC,OVRDAF,TSURFA,OVRSED,NSED,JM(7),WTRMT5,    | SEDI 9  |
| * NSCRN,SCRCAP,SUAREA,FARAB,NMS,AREAMS,VOLCON,VOLSED,ALJUT,       | SEDI 10 |
| * BDINT(7),SSINT(7),WAIN(7),BDOU(7),SSOUT(7),WAOUT(7),            | SEDI 11 |
| * WARM(7),BDRMT(7),SSRM(7),CHEMUT(8),CL2UT(8),QMOD(20),WTRMT1,    | SEDI 12 |
| * QCRM(7),QQUO(7),QCRM(7),QQUO(7),QCRM(7),QQUO(7),                | SEDI 13 |
| * BCRM(7),BDOU(7),BCRM(7),BDOU(7),BCRM(7),BDOU(7),                | SEDI 14 |
| * SCRM(7),SCOU(7),SCRM(7),SCOU(7),SCRM(7),SCOU(7),                | SEDI 15 |
| * BDRM(7),BCRM(7),SCRM(7)                                         | SEDI 16 |
| COMMON /TBLK/DT,NDT,KDT                                           | SEDI 17 |
| COMMON /STBK/QIN(150),BODIN(150),SUSIN(150),COLIN(150),           | SEDI 18 |
| * QINST,QCUST,QINSTL,QCUSTL,STORL,QUTO,STORO,                     | SEDI 19 |
| * ISPRIN,IPOL,DEPMAX,QCMAX,DEPTH,                                 | SEDI 20 |
| * ATERM(11),AO2DT2(11),BOPTH(11),BSTOR(11),                       | SEDI 21 |
| * DUMSTR(11),DUMDEP(11),                                          | SEDI 22 |
| * VOLIN(150),VOLOUT(150),STOR,CUMIN,CUMOUT,                       | SEDI 23 |
| * SBOD,SSS,SCGL,BDOU,SUSOUT,CULOUT,                               | SEDI 24 |
| * ISTMC,ISTYP,ISTOUT,                                             | SEDI 25 |
| * QPUMP,DSTART,DSTOP,                                             | SEDI 26 |
| * DTON,STORMX,DTPUMP,DTMURE,STORF,APLAN,                          | SEDI 27 |
| * CLAND,CSTOR,CPS,CTOTAL,CPCUYD,CPACRE,                           | SEDI 28 |
| * LP,JP,LPREV,LABEL,DETENT(150),FRAC(150)                         | SEDI 29 |
| C GO TO (3600,3700), ISTOR                                        | SEDI 30 |
| C SED TANKS TO BE INSTALLED (NO ASSOCIATED STORAGE)               | SEDI 31 |
| 3600 AREA = SEDA*NSED                                             | SEDI 32 |
| GO TO 3800                                                        | SEDI 33 |
| C SEDIMENTATION IN ASSOCIATED STORAGE UNIT                        | SEDI 34 |
| C 3700 CALL INTERP(ADEPTH,AASURF,11,DEPTH,AREA,KFLAG)             | SEDI 35 |
| C                                                                 | SEDI 36 |
| C 3700 CALL INTERP(ADEPTH,AASURF,11,DEPTH,AREA,KFLAG)             | SEDI 37 |
| C                                                                 | SEDI 38 |
| IF (KFLAG .EQ. -10) GO TO 901                                     | SEDI 39 |
| IF (KFLAG .EQ. 10) GO TO 902                                      | SEDI 40 |
| C VOLSED = ??                                                     | SEDI 41 |
| C                                                                 | SEDI 42 |
| 3800 OVFR = QQIN(3)*646000.0/AREA                                 | SEDI 43 |
| IF (OVFR .LE. 300.0) OVFR = 300.0                                 | SEDI 44 |
| SREFF = 0.656 + 0.06*SCIN(3)/190.0 - 0.4*(OVFR-300.0)/2000.0      | SEDI 45 |
| IF (SREFF .GE. 0.76) SREFF = 0.76                                 | SEDI 46 |
| IF (SREFF .LE. 0.30) SREFF = 0.30                                 | SEDI 47 |
| FACTOR = 0.646*8.35*DT/1440.0                                     | SEDI 48 |
| SSRM(3) = SREFF*SCIN(3)*QQIN(3)*FACTOR                            | SEDI 49 |
| BREFF = 0.55*SREFF                                                | SEDI 50 |
| IF (ICL2 .NE. 1) GO TO 3805                                       | SEDI 51 |
| BREFF = BREFF*1.15                                                | SEDI 52 |
| C? CALL KILL(VOLSED,SCIF,SCIN(3),QQIN(3),CCIN,CCOU)               | SEDI 53 |
| CC SOME OF ABOVE ARGUMENTS ARE NEEDED IN COMMON                   | SEDI 54 |
| 3805 BDRM(3) = BREFF*BCIN(3)*QQIN(3)*FACTOR                       | SEDI 55 |
| SSOU(3) = SSIN(3) - SSRM(3)                                       | SEDI 56 |
| BDOU(3) = BDIN(3) - BDRM(3)                                       | SEDI 57 |
| QCRM(3) = SSRM(3)*20.0/(8.35*DT*449.0)                            | SEDI 58 |
| GO TO (3810,3820), ISTOR                                          | SEDI 59 |
| 3810 IF (QCRM(3) .LT. 0.1) QCRM(3) = 0.1                          | SEDI 60 |

|                                                                        |         |
|------------------------------------------------------------------------|---------|
| WARM(3) = QCRM(3)*DT*60.0                                              | SEDI 61 |
| WAOU(3) = WAIN(3) - WARM(3)                                            | SEDI 62 |
| QQOU(3) = QQIN(3) - QCRM(3)                                            | SEDI 63 |
| GO TO 3850                                                             | SEDI 64 |
| 3820 IF (QCRM(3) .LT. 0.1 .AND. DEPTH .NE. 0.0) QCRM(3) = 0.1          | SEDI 65 |
| IF (DEPTH .EQ. 0.0) QCRM(3) = 0.0                                      | SEDI 66 |
| WARM(3) = QCRM(3)*DT*60.0                                              | SEDI 67 |
| WAOU(3) = WAIN(3) - WARM(3)                                            | SEDI 68 |
| QQOU(3) = QQIN(3) - QCRM(3)                                            | SEDI 69 |
| IF (WACU(3) .LT. 0.0) WACU(3) = 0.0                                    | SEDI 70 |
| IF (QQOU(3) .LT. 0.0) QQOU(3) = 0.0                                    | SEDI 71 |
| C MASS BALANCE INEXACT IN THIS CASE                                    | SEDI 72 |
| 3850 SCRM(3) = SSRM(3)/(QCRM(3)*FACTOR)                                | SEDI 73 |
| BCRM(3) = BDRM(3)/(QCRM(3)*FACTOR)                                     | SEDI 74 |
| SCOU(3) = SSOU(3)/(QQOU(3)*FACTOR)                                     | SEDI 75 |
| BCOU(3) = BDOU(3)/(QQOU(3)*FACTOR)                                     | SEDI 76 |
| C? CL2LT =                                                             | SEDI 77 |
| GO TO 9999                                                             | SEDI 78 |
| C                                                                      | SEDI 79 |
| C----- ERROR MESSAGES -----                                            | SEDI 80 |
| C                                                                      | SEDI 81 |
| 901 WRITE(6,691)                                                       | SEDI 82 |
| 691 FORMAT('0 *** TERMINATE - INPUT TO INTERP PROCEDURE IS LESS THAN L | SEDI 83 |
| *OWEST VALUE ON CURVE (IN SUBRT. SEDIM)')                              | SEDI 84 |
| STOP                                                                   | SEDI 85 |
| C                                                                      | SEDI 86 |
| 902 WRITE(6,692)                                                       | SEDI 87 |
| 692 FORMAT('0 *** TERMINATE - INPUT TO INTERP PROCEDURE IS GREATER THA | SEDI 88 |
| *N LARGEST VALUE ON CURVE (IN SUBRT. SEDIM)')                          | SEDI 89 |
| STOP                                                                   | SEDI 90 |
| C                                                                      | SEDI 91 |
| C                                                                      | SEDI 92 |
| 9999 RETURN                                                            | SEDI 93 |
| END                                                                    | SEDI 94 |
| C=====                                                                 | SEDI 95 |

|                   |                                                                 |         |
|-------------------|-----------------------------------------------------------------|---------|
| SUBROUTINE HIGHRF |                                                                 | HIGH 1  |
| COMMON            | CCNVER,KHCUR,KMIN,L,KMOD,NFLAG,BIG,HEAD1,HEAD2,                 | HIGH 2  |
| *                 | QDESYN,QCIF,WAQU(7),WAIN(7),QQU(7),QQIN(7),WARM(7),QURL,        | HIGH 3  |
| *                 | BDIF,CCRM(7),BDQU(7),BDIN(7),BCQU(7),BCIN(7),BDRL,              | HIGH 4  |
| *                 | SSIF,SSIN(7),SSQU(7),SCQU(7),SCIN(7),SSRM(7),SSRL,CQIF,CURL,    | HIGH 5  |
| *                 | ADEPTH(11),AASURF(11),ITREAT(7),ISTOR,IPRINT,ICOST,HKFD,        | HIGH 6  |
| *                 | MODSI2,ICHEM,ICL2,SCREEN,QQIFMX,DESF,IRANGE,KNTOF,TRIBA,SEDA,   | HIGH 7  |
| *                 | SQM,SREFFH,BREFFH,NUNITH,UAREAH,OPRAMA,ICHEMH,HM,VOLDAF,ITABLE, | HIGH 8  |
| *                 | MOUCST,TOTCST,RECIRC,OVRDAF,TSURFA,OVRSED,NSED,JM(7),WTRMT5,    | HIGH 9  |
| *                 | NSCRN,SCRCAP,SUAREA,FAREAB,AAMS,AREAMS,VOLCON,VOLSED,ALUMUT,    | HIGH 10 |
| *                 | BDINT(7),SSINT(7),WAINT(7),BDOUT(7),SSOUT(7),WAOUT(7),          | HIGH 11 |
| *                 | WARM(7),BCRMT(7),SSRMT(7),CHEMUT(8),CL2UT(8),QMOD(20),WTRMT1,   | HIGH 12 |
| *                 | QQRMT(7),QQOUT(7),QQRMMX(7),QQOUMX(7),QQRMMN(7),QQOUMN(7),      | HIGH 13 |
| *                 | BCRMT(7),BCOUT(7),BCRMMX(7),BCOUMX(7),BCRMMN(7),BCOUMN(7),      | HIGH 14 |
| *                 | SCRMT(7),SCOUT(7),SCRMMX(7),SCOUMX(7),SCRMMN(7),SCOUMN(7),      | HIGH 15 |
| *                 | BDRM(7),BCRM(7),SCRM(7)                                         | HIGH 16 |
| COMMON            | /TBLK/DT,NDT,KDT                                                | HIGH 17 |
| C                 |                                                                 | HIGH 18 |
| CC                | ACCORDING TO AJB, THIS SUBROUTINE                               | HIGH 19 |
| CC                | IS ONLY VALID FOR DT = 10 MIN.                                  | HIGH 20 |
| CC                | FIX UP FOR TIME GIVEN IN SUBRT. TRCHEK                          | HIGH 21 |
| CC                |                                                                 | HIGH 22 |
|                   | IF (KDT .GT. 1) GO TO 4300                                      | HIGH 23 |
|                   | ASSIGN 4310 TC LABEL                                            | HIGH 24 |
|                   | NDTBH = HKFD*NUNITH - 1.0                                       | HIGH 25 |
|                   | SBOB = 0.0                                                      | HIGH 26 |
|                   | S = 0.0                                                         | HIGH 27 |
| C                 | S = SOLIDS HELD IN FILTER (LB/SQ.FT)                            | HIGH 28 |
| C                 |                                                                 | HIGH 29 |
|                   | 4300 GO TO LABEL,(4310,4320,4330,4340,4350)                     | HIGH 30 |
| C                 | INITIALLY, BEFORE ANY BACKWASH IS NECESSARY                     | HIGH 31 |
|                   | 4310 FACTOR = S*20.0/SQM                                        | HIGH 32 |
|                   | IF (FACTOR .GT. 1.0) FACTOR = 1.0                               | HIGH 33 |
|                   | SREFF = 0.5*SREFFH*(1.0 + FACTOR)                               | HIGH 34 |
|                   | BREFF = 0.5*BREFFH*(1.0 + FACTOR)                               | HIGH 35 |
| C                 |                                                                 | HIGH 36 |
|                   | SCQU(4) = SCIN(4)*(1.0 - SREFF)                                 | HIGH 37 |
|                   | BCQU(4) = BCIN(4)*(1.0 - BREFF)                                 | HIGH 38 |
|                   | QQU(4) = QQIN(4)                                                | HIGH 39 |
|                   | OPRA = 449.0*QQIN(4)/(NUNITH*UAREAH)                            | HIGH 40 |
|                   | DS = SCIN(4)*SREFF*(OPRA/695.0)*8.35*(DT/1440.0)                | HIGH 41 |
|                   | S = S + DS                                                      | HIGH 42 |
|                   | HU = ((OPRA/OPRAMA)**1.18)*0.40*HM                              | HIGH 43 |
|                   | HCL = (OPRA/OPRAMA)*(S/SQM)*0.60*HM                             | HIGH 44 |
|                   | H = HU + HCL                                                    | HIGH 45 |
| C                 |                                                                 | HIGH 46 |
|                   | H1 = HM*C.90*(NUNITH - 1)/NUNITH                                | HIGH 47 |
|                   | IF (H .GE. H1) ASSIGN 4320 TO LABEL                             | HIGH 48 |
|                   | QCRM(4) = 0.0                                                   | HIGH 49 |
|                   | BCRM(4) = BIG                                                   | HIGH 50 |
|                   | SCRM(4) = BIG                                                   | HIGH 51 |
|                   | GO TO 4390                                                      | HIGH 52 |
| C                 | FIRST TIMESTEP OF BACKWASH                                      | HIGH 53 |
|                   | 4320 KDTBH = 0                                                  | HIGH 54 |
|                   | ASSIGN 4330 TC LABEL                                            | HIGH 55 |
|                   | S1 = 0.0                                                        | HIGH 56 |
|                   | OPRA = 449.0*QQIN(4)/(UAREAH*(NUNITH-1))                        | HIGH 57 |
|                   | DS = SCIN(4)*SREFF*(OPRA/695.0)*8.35*(DT/1440.0)                | HIGH 58 |
|                   | S = S + DS                                                      | HIGH 59 |
|                   | SCQU(4) = SCIN(4)*(1.0 - SREFF)                                 | HIGH 60 |

|      |                                                        |          |
|------|--------------------------------------------------------|----------|
|      | BCOU(4) = BCIN(4)*(1.0 - BREFF)                        | HIGH 61  |
| 4325 | QORM(4) = (UAREAH*15.0)/449.0                          | HIGH 62  |
|      | QQOU(4) = QQIN(4) - QORM(4)                            | HIGH 63  |
|      | SCRM(4) = UAREAH*S*1440.0/(10.0*QORM(4)*0.646*8.35)    | HIGH 64  |
|      | DBOD = BCIN(4)*BREFF*(OPRA/659.0)*8.35*(10.0/1440.0)   | HIGH 65  |
|      | SBOD = SBOD + DBOD                                     | HIGH 66  |
|      | BCRM(4) = UAREAH*SBOD*1440.0/(10.0*QORM(4)*0.646*8.35) | HIGH 67  |
|      | GO TO 4390                                             | HIGH 68  |
| C    | SUBSEQUENT TIMESTEPS OF BACKWASH                       |          |
| 4330 | KDTBW = KDTBW + 1                                      | HIGH 69  |
|      | IF (KDTBW .GE. NDTBW) ASSIGN 4340 TO LABEL             | HIGH 70  |
|      | SREFF2 = (0.5 + NUNITH - 1.0)*SREFF/NUNITH             | HIGH 71  |
|      | BREFF2 = (0.5 + NUNITH - 1.0)*BREFF/NUNITH             | HIGH 72  |
|      | SCOU(4) = SCIN(4)*(1.0 - SREFF2)                       | HIGH 73  |
|      | BCOU(4) = BCIN(4)*(1.0 - BREFF2)                       | HIGH 74  |
|      | DS = SCIN(4)*SREFF*(OPRA/695.0)*8.35*(10.0/1440.0)     | HIGH 75  |
|      | S = S + DS                                             | HIGH 76  |
|      | FACTR1 = S1*20.0/SQM                                   | HIGH 77  |
|      | IF (FACTR1 .GE. 1.0) FACTR1 = 1.0                      | HIGH 78  |
|      | SREFF1 = 0.5*SREFF*(1.0 + FACTR1)                      | HIGH 79  |
|      | DS1 = SCIN(4)*SREFF1*(OPRA/695.0)*8.35*(10.0/1440.0)   | HIGH 80  |
|      | S1 = S1 + DS1                                          | HIGH 81  |
|      | GO TO 4325                                             | HIGH 82  |
| C    | FIRST TIMESTEP AFTER BACKWASH COMPLETED                |          |
| 4340 | ASSIGN 4350 TO LABEL                                   | HIGH 83  |
|      | SREFF2 = (0.5 + NUNITH - 1.0)*SREFF/NUNITH             | HIGH 84  |
|      | BREFF2 = (0.5 + NUNITH - 1.0)*BREFF/NUNITH             | HIGH 85  |
|      | SCOU(4) = SCIN(4)*(1.0 - SREFF2)                       | HIGH 86  |
|      | BCOU(4) = BCIN(4)*(1.0 - BREFF2)                       | HIGH 87  |
|      | QORM(4) = 0.0                                          | HIGH 88  |
|      | QQOU(4) = QQIN(4)                                      | HIGH 89  |
|      | DS1 = SCIN(4)*SREFF1*(OPRA/695.0)*8.35*(10.0/1440.0)   | HIGH 90  |
|      | S1 = S1 + DS1                                          | HIGH 91  |
|      | SCRM(4) = BIG                                          | HIGH 92  |
|      | BCRM(4) = BIG                                          | HIGH 93  |
|      | GO TO 4390                                             | HIGH 94  |
| C    | SUBSEQUENT TIMESTEPS AFTER BACKWASH COMPLETION         |          |
| 4350 | FACTCR = 1.0                                           | HIGH 95  |
|      | SREFF = 0.5*SREFFH*(1.0 + FACTOR)                      | HIGH 96  |
|      | BREFF = 0.5*BREFFH*(1.0 + FACTOR)                      | HIGH 97  |
|      | SCOU(4) = SCIN(4)*(1.0 - SREFF)                        | HIGH 98  |
|      | BCOU(4) = BCIN(4)*(1.0 - BREFF)                        | HIGH 99  |
|      | QQOU(4) = QQIN(4)                                      | HIGH 100 |
|      | OPRA = 449.0*QQIN(4)/(NUNITH*UAREAH)                   | HIGH 101 |
|      | DS1 = SCIN(4)*SREFF1*(OPRA/695.0)*8.35*(10.0/1440.0)   | HIGH 102 |
|      | S1 = S1 + DS1                                          | HIGH 103 |
|      | QORM(4) = 0.0                                          | HIGH 104 |
|      | SCRM(4) = BIG                                          | HIGH 105 |
|      | BCRM(4) = BIG                                          | HIGH 106 |
|      | HO = ((OPRA/OPRAMA)**1.18)*0.40*HM                     | HIGH 107 |
|      | HCL = (OPRA/OPRAMA)*(S1/SQM)*0.60*HM                   | HIGH 108 |
|      | H = HO + HCL                                           | HIGH 109 |
|      | IF (H .LT. HM) GO TO 4390                              | HIGH 110 |
|      | SBOD = 0.0                                             | HIGH 111 |
|      | S = S1                                                 | HIGH 112 |
|      | ASSIGN 4320 TO LABEL                                   | HIGH 113 |
| C    | ALL TIMESTEPS                                          |          |
| 4390 | BDQU(4) = QQOU(4)*BCOU(4)*0.646*8.35*DT/1440.0         | HIGH 114 |
|      | SSOU(4) = QQOU(4)*SCOU(4)*0.646*8.35*DT/1440.0         | HIGH 115 |
|      |                                                        | HIGH 116 |
|      |                                                        | HIGH 117 |
|      |                                                        | HIGH 118 |
|      |                                                        | HIGH 119 |
|      |                                                        | HIGH 120 |

|   |                                              |         |
|---|----------------------------------------------|---------|
|   | BDRM(4) = BDIN(4) - BDOU(4)                  | HIGH121 |
|   | SSRM(4) = SSIN(4) - SSOU(4)                  | HIGH122 |
|   | WARM(4) = QCRM(4)*DT*60.0                    | HIGH123 |
|   | WAOU(4) = WAIN(4) - WARM(4)                  | HIGH124 |
| C | DOSE POLYMERS AT 4 MG/L AND ALUM AT 150 MG/L | HIGH125 |
|   | CHEMLH = QQIN(4)*0.646*8.35*4.0*DT/1440.0    | HIGH126 |
|   | ALUMUH = QQIN(4)*0.646*8.35*150.0*DT/1440.0  | HIGH127 |
|   | CHEMUT(4) = CHEMUT(4) + CHEMLH               | HIGH128 |
|   | ALUMUT = ALUMUT + ALUMUH                     | HIGH129 |
| C | RETURN                                       | HIGH130 |
|   | END                                          | HIGH131 |
| C | =====                                        | HIGH132 |
|   |                                              | HIGH133 |

|   |                                                          |         |
|---|----------------------------------------------------------|---------|
|   | SUBROUTINE KILL(CONVOL,SCIF,SCIN,QQIN,CCIN,CCOU)         | KILL 1  |
| C |                                                          | KILL 2  |
| C | GIVEN THE FIRST 5 ARGUMENTS,                             | KILL 3  |
| C | CCOMPUTES COLIFORM MPN IN OUTFLOW = CCOU                 | KILL 4  |
| C |                                                          | KILL 5  |
| C | COMPUTE FRACTION F DOSED (ASSUMING 15 MIN DETENT. TIKILL | KILL 6  |
| C |                                                          | KILL 7  |
|   | QKILL = CCNOL/(15.0*60.0)                                | KILL 8  |
|   | IF (QKILL .GT. QQIN) QKILL = QQIN                        | KILL 9  |
|   | F = QKILL/QQIN                                           | KILL 10 |
| C | COMPUTE MPN COLIFORMS AFTER REMIXING OF DOSED FRACTIKILL | KILL 11 |
|   | CCIN = CCIN*SCIN/SCIF                                    | KILL 12 |
|   | CCTR = CCIN*0.001                                        | KILL 13 |
|   | CCOU = F*CCTR + (1.0-F)*CCIN                             | KILL 14 |
| C |                                                          | KILL 15 |
|   | RETURN                                                   | KILL 16 |
|   | END                                                      | KILL 17 |
| C | =====                                                    | KILL 18 |

|                                                                       |         |
|-----------------------------------------------------------------------|---------|
| SUBROUTINE STRAGE                                                     | STRA 1  |
| COMMON /STBK/QIN(150),BODIN(150),SUSIN(150),COLIN(150),               | STRA 2  |
| * QINST,QCUST,QINSTL,QCUSTL,STURL,QOUTO,STORO,                        | STRA 3  |
| * ISPRIN,IFCL,DEPMAX,QCMAX,DEPTH,                                     | STRA 4  |
| * ATERM(11),AO2DT2(11),BDEPTH(11),BSTOR(11),                          | STRA 5  |
| * DUMSTR(11),DUMDEP(11),                                              | STRA 6  |
| * VOLIN(150),VOLOUT(150),STOR,CUMIN,CUMOUT,                           | STRA 7  |
| * SBOD,SSS,SCOL,BODOUT,SUSOUT,COLOUT,                                 | STRA 8  |
| * ISTMC,ISTTYP,ISTOUT,                                                | STRA 9  |
| * QPUMP,CSTART,DSTOP,                                                 | STRA 10 |
| * DTON,STORMX,DTPUMP,DTMORE,STORF,APLAN.                              | STRA 11 |
| * CLAND,CSTOR,CPS,CTOTAL,CPCUYD,CPACRE,                               | STRA 12 |
| * LP,JP,LPREV,LABEL,DETENT(150),FRAC(150)                             | STRA 13 |
| COMMON /TBLK/DT,NDT,KDT                                               | STRA 14 |
| C DT=DT*60.0                                                          | STRA 15 |
| IF (KDT.GT. 1) GO TO 1300                                             | STRA 16 |
| C IF (ISPRIN.GT. 0) GO TO 1100                                        | STRA 17 |
| GO TO 1200                                                            | STRA 18 |
| C 1100 WRITE(6,601) NDT                                               | STRA 19 |
| 601 FORMAT('1STORAGE SOLUTION FOR', I4, ' TIME-STEPS FOLLOWS',        | STRA 20 |
| * ' , CN A STEP-BY-STEP BASIS', //)                                   | STRA 21 |
| WRITE(6,602)                                                          | STRA 22 |
| 602 FORMAT('OU STP TIME INFLOW OUTFLOW STORAGE DEPTH', 2X, 'IN: BOD', | STRA 23 |
| * 5X, 'SS STOR: BOD', 7X, 'SS', 4X, 'BOD', 5X, 'SS OUT: BOD',         | STRA 24 |
| * 5X, 'SS', 4X, 'BOD', 5X, 'SS', 2X, ' J L', /,                       | STRA 25 |
| * ' N NC (MIN) (CFS) (CFS) (CU.FT) (FT.)', 5X,                        | STRA 26 |
| * '(LB) (LB)', 6X, '(LB)', 5X, '(LB) (MG/L) (MG/L)', 5X,              | STRA 27 |
| * '(LB) (LB) (MG/L) (MG/L)', 2X, ' P P', /)                           | STRA 28 |
| C 1200 STORMX = 0.0                                                   | STRA 29 |
| LPREV = 1                                                             | STRA 30 |
| SBOD = 0.0                                                            | STRA 31 |
| SSS = 0.0                                                             | STRA 32 |
| SCOL = 0.0                                                            | STRA 33 |
| STOR = STCR0                                                          | STRA 34 |
| DO 1230 I=1,11                                                        | STRA 35 |
| DUMSTR(I) = BSTOR(I)                                                  | STRA 36 |
| 1230 DUMDEP(I) = BDEPTH(I)                                            | STRA 37 |
| C CALL INTERP(DUMSTR,DUMDEP,11,STOR,DEPTH,KFLAG)                      | STRA 38 |
| C IF (KFLAG.EQ. -10) GO TO 901                                        | STRA 39 |
| IF (KFLAG.EQ. 10) GO TO 902                                           | STRA 40 |
| C DEPTH=L = DEPTH                                                     | STRA 41 |
| IF (ISPRIN.LT. 1) GO TO 1250                                          | STRA 42 |
| WRITE(6,603) QOUTO, STORO, DEPTH                                      | STRA 43 |
| 603 FORMAT(' 0 0.0 0.0', F7.1, F10.0, F6.2)                           | STRA 44 |
| 1250 CONTINUE                                                         | STRA 45 |
| WRITE(6,604)                                                          | STRA 46 |
| 604 FORMAT(' ')                                                       | STRA 47 |
| C-----                                                                | STRA 48 |
| 1300 DO 1350 I=1,11                                                   | STRA 49 |
| DUMSTR(I) = BSTOR(I)                                                  | STRA 50 |
| 1350 DUMDEP(I) = BDEPTH(I)                                            | STRA 51 |
| RKTSTP = KDT                                                          | STRA 52 |
| TIME2P = DT*RKTSTP/60.0                                               | STRA 53 |
|                                                                       | STRA 54 |
|                                                                       | STRA 55 |
|                                                                       | STRA 56 |
|                                                                       | STRA 57 |
|                                                                       | STRA 58 |
|                                                                       | STRA 59 |
|                                                                       | STRA 60 |

|                                                           |         |
|-----------------------------------------------------------|---------|
| BODOUT = 0.0                                              | STRA 61 |
| SUSOUT = 0.0                                              | STRA 62 |
| COLOUT = 0.0                                              | STRA 63 |
| C                                                         | STRA 64 |
| IF (ISTOUT .EQ. 5 .OR. ISTOUT .EQ. 6) GO TO 2000          | STRA 65 |
| GO TO 3000                                                | STRA 66 |
| C-----                                                    | STRA 67 |
| 2000 IF (KDT .GT. 1) GO TO 2100                           | STRA 68 |
| QINSTL = 0.0                                              | STRA 69 |
| QOUSTL = COUTO                                            | STRA 70 |
| STORL = STORO                                             | STRA 71 |
| CUMIN = 0.0                                               | STRA 72 |
| CUMOUT = 0.0                                              | STRA 73 |
| STOR = STORO                                              | STRA 74 |
| LP = 0                                                    | STRA 75 |
| JP = 0                                                    | STRA 76 |
| DTON = 0.0                                                | STRA 77 |
| C                                                         | STRA 78 |
| 2100 VOLIN(KET) = 0.5*DT*(QINSTL + QINST)                 | STRA 79 |
| VOLOUZ = DT*QOUSTL                                        | STRA 80 |
| STORZ = STORL + VOLIN(KDT) - VOLOUZ                       | STRA 81 |
| C                                                         | STRA 82 |
| CALL INTERP(DUMSTR,DUMDEP,11,STORZ,DEPTHZ,KFLAG)          | STRA 83 |
| C                                                         | STRA 84 |
| IF (KFLAG .EQ. -10) GO TO 901                             | STRA 85 |
| IF (KFLAG .EQ. 10) GO TO 902                              | STRA 86 |
| C                                                         | STRA 87 |
| C *** THE FOLLOWING STATEMENTS, ABOVE 2150, ARE TEMPORARY | STRA 88 |
| C                                                         | STRA 89 |
|                                                           | STRA 90 |
| IF (QINST .GT. QPUMP .AND. QOUSTL .EQ. 0.0)               | STRA 91 |
| * GO TO 2120                                              | STRA 92 |
| GO TO 2150                                                | STRA 93 |
| 2120 FON = (QINST-QPUMP)/(QINST-QINSTL)                   | STRA 94 |
| DSTART = DEPTHZ*(1.0-FON) + FON*DEPTHL                    | STRA 95 |
| DTON = DTON + FON                                         | STRA 96 |
| WRITE(6,605) DSTART                                       | STRA 97 |
| 605 FORMAT(' ', 60X, 'NEW DSTART =', F6.2, ' FT.')        | STRA 98 |
| C                                                         | STRA 99 |
| 2150 IF (QOUSTL .EQ. 0.0 .AND. DEPTHZ .GT. DSTART)        | STRA100 |
| * GO TO 2200                                              | STRA101 |
| IF (QOUSTL .GT. 0.0 .AND. DEPTHZ .LT. DSTOP)              | STRA102 |
| * GO TO 2300                                              | STRA103 |
| QOUST = QOUSTL                                            | STRA104 |
| VOLOUT(KDT) = VOLOUZ                                      | STRA105 |
| STOR = STORZ                                              | STRA106 |
| DEPTH = DEPTHZ                                            | STRA107 |
| GO TO 2500                                                | STRA108 |
| C                                                         | STRA109 |
| 2200 FON = (DEPTHZ - DSTART)/(DEPTHZ - DEPTHL)            | STRA110 |
| QOUST = QPUMP                                             | STRA111 |
| GO TO 2400                                                | STRA112 |
| C                                                         | STRA113 |
| 2300 FON = (DEPTHL - DSTOP)/(DEPTHL - DEPTHZ)             | STRA114 |
| QOUST = 0.0                                               | STRA115 |
| C                                                         | STRA116 |
| 2400 IF (FON .LT. 0.0) GO TO 8100                         | STRA117 |
| VOLOUT(KCT) = FON*DT*QPUMP                                | STRA118 |
| STOR = STORL + VOLIN(KDT)-VOLOUT(KDT)                     | STRA119 |
|                                                           | STRA120 |



|   |                                                  |         |
|---|--------------------------------------------------|---------|
| C | CALL INTERP(DUMSTR,DUMDEP,11,STOR,DEPTH,KFLAG)   | STRA121 |
| C | IF (KFLAG .EQ. -10) GO TO 901                    | STRA122 |
| C | IF (KFLAG .EQ. 10) GO TO 902                     | STRA123 |
| C |                                                  | STRA124 |
| C | 2500 CUMIN = CUMIN + VOLIN(KDT)                  | STRA125 |
| C | CUMOUT = CUMOUT + VOLOUT(KDT)                    | STRA126 |
| C | QINSTL = QINST                                   | STRA127 |
| C | QOUSTL = COUST                                   | STRA128 |
| C | DEPTHL = DEPTH                                   | STRA129 |
| C | STORL = STOR                                     | STRA130 |
| C | IF (KDT .EQ. NOT) STORF = STOR                   | STRA131 |
| C | IF (STOR .GT. STORMX) STORMX = STOR              | STRA132 |
| C | IF (QOUST .GT. 0.0) DTON = DTON + 1.0            | STRA133 |
| C | GO TO 4000                                       | STRA134 |
| C |                                                  | STRA135 |
| C |                                                  | STRA136 |
| C |                                                  | STRA137 |
| C |                                                  | STRA138 |
| C | 3000 CALL SRCUTE                                 | STRA139 |
| C |                                                  | STRA140 |
| C | CALL INTERP(DUMSTR,DUMDEP,11,STOR,DEPTH,KFLAG)   | STRA141 |
| C | IF (KFLAG .EQ. -10) GO TO 901                    | STRA142 |
| C | IF (KFLAG .EQ. 10) GO TO 902                     | STRA143 |
| C |                                                  | STRA144 |
| C |                                                  | STRA145 |
| C | COMPUTE SEDIMENT AND BOD OUTFLOW                 | STRA146 |
| C |                                                  | STRA147 |
| C |                                                  | STRA148 |
| C | DETERMINE PLUG FRACTIONS AND DETENTION TIMES     | STRA149 |
| C |                                                  | STRA150 |
| C | CUMIN = CUMULATIVE INFLOW (CU.FT.) SINCE T = 0   | STRA151 |
| C | CUMOUT = CUMULATIVE OUTFLOW (CU.FT.) SINCE T = 0 | STRA152 |
| C | SSIN = SS INFLOW (LB) IN THIS TIME-STEP          | STRA153 |
| C | BCDIN = BOD INFLOW (LB) IN THIS TIME-STEP        | STRA154 |
| C | SBOD = BOD (LB) IN RESERVOIR                     | STRA155 |
| C | SSS = SS (LB) IN RESERVOIR                       | STRA156 |
| C | BODOUT = BOD OUTFLOW (LB)                        | STRA157 |
| C | SSOUT = SS OUTFLOW (LB)                          | STRA158 |
| C | BODCOT = BOD OUTFLOW CONC. (MG/L)                | STRA159 |
| C | SSCOT = SS OUTFLOW CONC. (MG/L)                  | STRA160 |
| C | SBODC = BOD CONC. (MG/L) AVG. IN RESERVOIR       | STRA161 |
| C | SSSC = SS CONC. (MG/L) AVG. IN RESERVOIR         | STRA162 |
| C | MG/L = 16050*(LB/CU.FT.)                         | STRA163 |
| C |                                                  | STRA164 |
| C | 3100 IF(KDT .GT. 1) GO TO 3200                   | STRA165 |
| C | VOLIN(1) = 0.5*QINST*DT                          | STRA166 |
| C | VOLOUT(1) = 0.5*(QOUST+QOUSTL)*DT                | STRA167 |
| C | CUMIN = VOLIN(1)                                 | STRA168 |
| C | CUMOUT = VOLOUT(1)                               | STRA169 |
| C | SBOD = 0.0                                       | STRA170 |
| C | SSS = 0.0                                        | STRA171 |
| C | SCUL = 0.0                                       | STRA172 |
| C | LPREV = 1                                        | STRA173 |
| C | JP = 0                                           | STRA174 |
| C | LP = 0                                           | STRA175 |
| C | GO TO 4000                                       | STRA176 |
| C | 3200 VOLIN(KDT) = 0.5*(QINST+QINSTL)*DT          | STRA177 |
| C | VOLOUT(KDT) = 0.5*(QOUST+QOUSTL)*DT              | STRA178 |
| C | CUMIN = CUMIN + VOLIN(KDT)                       | STRA179 |
| C | CUMOUT = CUMOUT + VOLOUT(KDT)                    | STRA180 |

|                                           |        |
|-------------------------------------------|--------|
| C-----                                    | STR181 |
| 4000 IF(VOLOUT(KDT) .LE. 0.0) GO TO 4200  | STR182 |
| C                                         | STR183 |
| CALL PLUGS                                | STR184 |
| C                                         | STR185 |
| IF (LABEL .EQ. 5000) GO TO 4200           | STR186 |
| C                                         | STR187 |
| C                                         | STR188 |
| C                                         | STR189 |
| C                                         | STR190 |
| C                                         | STR191 |
| C                                         | STR192 |
| C                                         | STR193 |
| C                                         | STR194 |
| C                                         | STR195 |
| C                                         | STR196 |
| C                                         | STR197 |
| C                                         | STR198 |
| C                                         | STR199 |
| C                                         | STR200 |
| 4100 DO 4150 KP=JP,LP                     | STR201 |
| BODOUT = BODOUT + BODIN(KP)*FRAC(KP)      | STR202 |
| COLOUT = COLOUT + COLIN(KP)*FRAC(KP)      | STR203 |
| 4150 SUSOUT = SUSOUT + SUSIN(KP)*FRAC(KP) | STR204 |
| C                                         | STR205 |
| BODCCT = BODCCT*16050.0/VOLOUT(KDT)       | STR206 |
| SSCOUT = SUSOUT *16050.0/VOLCUT(KDT)      | STR207 |
| COLCOT = COLOUT/(VOLOUT(KDT)*283.2)       | STR208 |
| GO TO 4300                                | STR209 |
| C                                         | STR210 |
| C                                         | STR211 |
| C                                         | STR212 |
| C                                         | STR213 |
| C                                         | STR214 |
| C                                         | STR215 |
| C                                         | STR216 |
| C                                         | STR217 |
| 4180 BODCCT = SBODC                       | STR218 |
| SSCOUT = SSC                              | STR219 |
| COLCCT = SCOLC                            | STR220 |
| BODOUT = BODCCT*VOLCUT(KDT)/16050.0       | STR221 |
| SUSOUT = SSCOUT*VOLCUT(KDT)/16050.0       | STR222 |
| COLOUT = COLCCT*VOLCUT(KDT)*283.2         | STR223 |
| GO TO 4300                                | STR224 |
| C                                         | STR225 |
| 4200 BODCCT = 0.0                         | STR226 |
| SSCOUT = 0.0                              | STR227 |
| COLCOT = 0.0                              | STR228 |
| IF (LPREV .GT. 1) GO TO 4300              | STR229 |
| LP = 0                                    | STR230 |
| JP = C                                    | STR231 |
| C                                         | STR232 |
| 4300 SBOD = SBOD - BODOUT + BODIN(KDT)    | STR233 |
| SSS = SSS - SUSOUT + SUSIN(KDT)           | STR234 |
| SCOL = SCOL - COLOUT + COLIN(KDT)         | STR235 |
| IF (STOR .EQ. 0.0) GO TO 4350             | STR236 |
| SBODC = SBOD*16050.0/STOR                 | STR237 |
| SSSC = SSS *16050.0/STOR                  | STR238 |
| SCULC = SCOL/(STOR*283.2)                 | STR239 |
| GO TO 4400                                | STR240 |
| 4350 SBODC = 0.0                          |        |

|                                                                        |         |
|------------------------------------------------------------------------|---------|
| SSSC = 0.0                                                             | STRA241 |
| SCOLC = 0.0                                                            | STRA242 |
| C                                                                      | STRA243 |
| 4400 IF (ISPRIN .LT. 1) GO TO 5300                                     | STRA244 |
| C                                                                      | STRA245 |
| C PRINT SOLUTION FOR THIS TIME-STEP                                    | STRA246 |
| C                                                                      | STRA247 |
| 5000 WRITE(6,606) KDT,TIME2M,QINST,QOUST,                              | STRA248 |
| * STOR,DEPTH,BODIN(KDT),SUSIN(KDT),                                    | STRA249 |
| * SBOD,SSS,SBODC,SSSC,BUOOUT,SUSOUT,                                   | STRA250 |
| * BUOCOT,SSCOUT,JP,LP                                                  | STRA251 |
| 606 FORMAT(' ', I4, F6.1, 2F7.1, F10.0, F6.2, 2X, 2F7.1, F10.1,        | STRA252 |
| * F9.1, 2F7.1, F9.1, 3F7.1, 2I4)                                       | STRA253 |
| WRITE(6,666) COLIN(KDT),SCOL,SCOLC,COLOUT,COLOCOT                      | STRA254 |
| 666 FORMAT(' ', 40X, 'COLIN= ', 1PE9.2, 'SCOL(MPN)=', E9.2,            | STRA255 |
| * 'CONC=', E9.2, 'COLOUT=', E9.2, 'CONC=', E9.2)                       | STRA256 |
| GO TO 5200                                                             | STRA257 |
| 5100 WRITE(6,606) KDT,TIME2M,QINST,QOUST,                              | STRA258 |
| * STOR,DEPTH                                                           | STRA259 |
| 5200 WRITE(6,604)                                                      | STRA260 |
| 5300 IF (KDT .LT. NDT) GO TO 5400                                      | STRA261 |
| IF (ISPRIN .LT. 1) GO TO 5400                                          | STRA262 |
| WRITE(6,607) NDT, CUMIN, CUMCOT                                        | STRA263 |
| 607 FORMAT('0', /, ' ', 26X, 'FOR THESE', I4, ' TIME STEPS ', /,       | STRA264 |
| * ' ', 36X, 'CUMULATIVE INFLOW =', F12.0, ' CU.FT.', /,                | STRA265 |
| * ' ', 36X, 'CUMULATIVE OUTFLOW =', F12.0, ' CU.FT.')                  | STRA266 |
| 5400 GO TO 9999                                                        | STRA267 |
| C                                                                      | STRA268 |
| C----- ERROR MESSAGES -----                                            | STRA269 |
| C                                                                      | STRA270 |
| 8100 WRITE(6,692)                                                      | STRA271 |
| 692 FORMAT(' ', 14X, 'BUFFER VOLUME BETWEEN LEVELS DSTART AND DSTOP ', | STRA272 |
| * 'IS TOO SMALL,', /,                                                  | STRA273 |
| * ' ', 14X, 'RESULTING IN NEGATIVE VOLOUT (AND FON).')                 | STRA274 |
| GO TO 9999                                                             | STRA275 |
| C                                                                      | STRA276 |
| 901 IF (KFLAG .EQ. 10) GO TO 902                                       | STRA277 |
| WRITE(6,693)                                                           | STRA278 |
| 693 FORMAT('0 *** TERMINATE - INPUT TO INTERP PROCEDURE IS LESS THAN L | STRA279 |
| *OWEST VALUE ON CURVE (IN SUBRT. STORAG)')                             | STRA280 |
| STOP 901                                                               | STRA281 |
| C                                                                      | STRA282 |
| 902 WRITE(6,694)                                                       | STRA283 |
| 694 FORMAT('0 *** TERMINATE - INPUT TO INTERP PROCEDURE IS GREATER THA | STRA284 |
| *N LARGEST VALUE ON CURVE (IN SUBRT. STORAG)')                         | STRA285 |
| STOP 902                                                               | STRA286 |
| C                                                                      | STRA287 |
| 9999 DT=DT/60.0                                                        | STRA288 |
| RETURN                                                                 | STRA289 |
| END                                                                    | STRA290 |
| C=====                                                                 | STRA291 |

|      |                                                         |         |
|------|---------------------------------------------------------|---------|
|      | SUBROUTINE PLUGS                                        | PLUG 1  |
|      | COMMON /STBK/QIN(150),BODIN(150),SUSIN(150),COLIN(150), | PLUG 2  |
|      | * QINST,CCUST,CINSTL,QOUSTL,STORL,QOUTO,STORO,          | PLUG 3  |
|      | * ISPRIN,IPOL,DEPMAX,CCMAX,DEPTH,                       | PLUG 4  |
|      | * ATERM(11),AO2CT2(11),BDEPTH(11),BSTOR(11),            | PLUG 5  |
|      | * DUMSTR(11),DUMDEP(11),                                | PLUG 6  |
|      | * VOLIN(150),VOLOUT(150),STOR,CUMIN,CUMOUT,             | PLUG 7  |
|      | * SBOD,SSS,SCOL,BODOUT,SUSOUT,COLOUT,                   | PLUG 8  |
|      | * ISTMOD,ISTTYP,ISTOUT,                                 | PLUG 9  |
|      | * QPUMP,CSTART,CSTOP,                                   | PLUG 10 |
|      | * DTON,STORMX,DTPUMP,CTMORE,STORF,APLAN,                | PLUG 11 |
|      | * CLAND,CSTOR,CPS,CTOTAL,CPCUYD,CPACRE,                 | PLUG 12 |
|      | * LP,JP,LPREV,LABEL,DETENT(150),FRAC(150)               | PLUG 13 |
|      | COMMON /TBK/DT,NDT,KDT                                  | PLUG 14 |
| C    |                                                         | PLUG 15 |
| C    | GIVEN KDT, VOLIN, VOLOUT & STORE ARRAYS, KDT & LPREV    | PLUG 16 |
| C    | FINDS JP AND LP (NUMBERS OF FIRST AND LAST PART-PLUG)   | PLUG 17 |
| C    | CCOMPUTES DETENT AND FRAC ARRAYS (KP=JP,LP)             | PLUG 18 |
| C    | (SUBPROGRAM BY G.T.)                                    | PLUG 19 |
| C    |                                                         | PLUG 20 |
|      | DT=DT*60.0                                              | PLUG 21 |
|      | LABEL = 0                                               | PLUG 22 |
|      | LP = KDT                                                | PLUG 23 |
|      | JP = LPREV                                              | PLUG 24 |
|      | VIKK = VOLIN(KDT)                                       | PLUG 25 |
|      | VOKK = VOLOUT(KDT)                                      | PLUG 26 |
|      | SUM1 = 0.0                                              | PLUG 27 |
|      | SUM = 0.0                                               | PLUG 28 |
| 1000 | SUM = SUM + VOLIN(LP)                                   | PLUG 29 |
|      | SUM1 = SUM + VOKK - VIKK                                | PLUG 30 |
|      | BACK = SUM1 - STOR                                      | PLUG 31 |
|      | IF (BACK .GT. VOKK) GO TO 1100                          | PLUG 32 |
|      | GO TO 1200                                              | PLUG 33 |
| 1100 | BACK = VOKK                                             | PLUG 34 |
| 1200 | IF (BACK) 1300,1400,2000                                | PLUG 35 |
| 1300 | LP = LP - 1                                             | PLUG 36 |
|      | IF (LP-1) 5000,1000,1000                                | PLUG 37 |
| C    |                                                         | PLUG 38 |
| 1400 | WRITE(6,601)                                            | PLUG 39 |
|      | 601 FORMAT('C A PART-PLUG = 0. EXECUTION TERMINATED.')  | PLUG 40 |
|      | STOP 1400                                               | PLUG 41 |
| C    |                                                         | PLUG 42 |
| 2000 | IF (VIKK .LE. 0.0) GO TO 4000                           | PLUG 43 |
|      | IF (LP .EC. KDT) GO TO 3000                             | PLUG 44 |
| C    |                                                         | PLUG 45 |
| C    | GENERAL CASE                                            | PLUG 46 |
| C    | A NUMBER OF PLUGS AND PART-PLUGS LEAVE                  | PLUG 47 |
| C    |                                                         | PLUG 48 |
|      | IF (LP .EQ. 1) GO TO 2500                               | PLUG 49 |
|      | IF (JP .GT. 1) GO TO 2800                               | PLUG 50 |
|      | NN = JP + 1                                             | PLUG 51 |
|      | IF (NN .EC. LP) GO TO 2600                              | PLUG 52 |
|      | SOUT = 0.0                                              | PLUG 53 |
|      | M = LP - 1                                              | PLUG 54 |
|      | NN = JP + 1                                             | PLUG 55 |
|      | DO 2050 L=NN,M                                          | PLUG 56 |
| 2050 | SOUT = SOUT + VOLIN(L)                                  | PLUG 57 |
|      | SOUT = SOUT + BACK                                      | PLUG 58 |
|      | STOT = SOUT + VOLIN(JP)                                 | PLUG 59 |
|      | IF (STOT .GE. VOKK) GO TO 2400                          | PLUG 60 |

|                                 |         |
|---------------------------------|---------|
| FRAC(JP)= 1.0                   | PLUG 61 |
| 2100 RIJP = KDT - JP            | PLUG 62 |
| DETENT(JP) = RIJP*DT            | PLUG 63 |
| DO 2200 L=NN,M                  | PLUG 64 |
| FRAC(L)= 1.0                    | PLUG 65 |
| RIL = KDT - L                   | PLUG 66 |
| 2200 DETENT(L) = RIL*DT         | PLUG 67 |
| 2300 FRAC(LP)= BACK/VOLIN(LP)   | PLUG 68 |
| RILP = KDT - LP                 | PLUG 69 |
| DETENT(LP) = RILP*DT            | PLUG 70 |
| LPREV = LP                      | PLUG 71 |
| GO TO 9999                      | PLUG 72 |
| C                               | PLUG 73 |
| 2400 FRGNT = VOKK - SOUT        | PLUG 74 |
| FRAC(JP)= FRONT/VOLIN(JP)       | PLUG 75 |
| GO TO 2100                      | PLUG 76 |
| C                               | PLUG 77 |
| 2500 FRONT = VOKK - BACK        | PLUG 78 |
| GO TO 2300                      | PLUG 79 |
| C                               | PLUG 80 |
| 2600 SOUT = BACK + VOLIN(JP)    | PLUG 81 |
| IF(SOUT .GE. VOKK) GO TO 2700   | PLUG 82 |
| FRAC(JP)= 1.0                   | PLUG 83 |
| RIJP = KDT - JP                 | PLUG 84 |
| DETENT(JP) = RIJP*DT            | PLUG 85 |
| GO TO 2300                      | PLUG 86 |
| C                               | PLUG 87 |
| 2700 FRUNT = VOKK - BACK        | PLUG 88 |
| FRAC(JP)= FRONT/VOLIN(JP)       | PLUG 89 |
| RIJP = KDT - JP                 | PLUG 90 |
| DETENT(JP) = RIJP*DT            | PLUG 91 |
| GO TO 2300                      | PLUG 92 |
| C                               | PLUG 93 |
| 2800 IF (LP .EQ. JP) GO TO 4100 | PLUG 94 |
| NN = JP + 1                     | PLUG 95 |
| IF(NN .EC. LP) GO TO 2700       | PLUG 96 |
| SOUT = 0.0                      | PLUG 97 |
| M = LP - 1                      | PLUG 98 |
| DO 2850 L=NN,M                  | PLUG 99 |
| 2850 SOUT = SOUT + VOLIN(L)     | PLUG100 |
| SOUT = SOUT + BACK              | PLUG101 |
| GO TO 2400                      | PLUG102 |
| C                               | PLUG103 |
| C                               | PLUG104 |
| C                               | PLUG105 |
| C                               | PLUG106 |
| 3000 IF(LP .EC. 1) GO TO 3700   | PLUG107 |
| IF(JP .GT. 1) GO TO 3800        | PLUG108 |
| NN = JP + 1                     | PLUG109 |
| IF(NN .EC. LP) GO TO 3500       | PLUG110 |
| SOUT = 0.0                      | PLUG111 |
| M = LP - 1                      | PLUG112 |
| NN = JP + 1                     | PLUG113 |
| DO 3050 L=NN,M                  | PLUG114 |
| 3050 SOUT = SOUT + VOLIN(L)     | PLUG115 |
| SOUT = SOUT + BACK              | PLUG116 |
| STOT = SOUT + VOLIN(JP)         | PLUG117 |
| IF(STOT .GE. VOKK) GO TO 3400   | PLUG118 |
| FRAC(JP)= 1.0                   | PLUG119 |
| 3100 RIJP = KDT - JP            | PLUG120 |

```

      DETENT(JP) = RIJP*DT
      DO 3200 L=NN,M
        FRAC(L)= 1.0
        RIL    = KDT - L
3200    DETENT(L) = RIL*DT
3300    FRAC(LP)= BACK/VOLIN(LP)
        DETENT(LP) = DT*STOR/VIKK
        LPREV = LP
      GO TO 9959
C
3400    FRONT = VOKK - SOUT
        FRAC(JP)= FRONT/VOLIN(JP)
      GO TO 3100
C
3500    SOUT   = BACK + VOLIN(JP)
        RIJP   = KDT - JP
        DETENT(JP) = RIJP*DT
        IF(SOUT .GE. VOKK) GO TO 3600
        FRAC(JP)= 1.0
      GO TO 3300
C
3600    FRONT = VOKK - BACK
        FRAC(JP)= FRONT/VOLIN(JP)
      GO TO 3300
C
3700    FRONT = VOKK - BACK
        FRAC(LP)= BACK/VIKK
        DETENT(LP) = DT*STOR/VIKK
        LPREV = LP
      GO TO 9999
C
3800    IF (LP .EQ. JP) GO TO 4100
        NN    = JP + 1
        IF(NN .EQ. LP) GO TO 3600
        SOUT   = 0.0
        M     = LP - 1
      DO 3850 L=NN,M
3850    SOUT   = SOUT + VOLIN(L)
        SOUT   = SOUT + BACK
        FRONT  = VOKK - SOUT
        FRAC(JP)= FRONT/VOLIN(JP)
      GO TO 3100
C
C          SPECIAL END CONDITION
C          NO INFLOW, OUTFLOW FROM STORAGE
C
4000    IF (LP .EQ. JP) GO TO 4100
        NN    = JP + 1
        IF(NN .EQ. LP) GO TO 4200
        SOUT   = 0.0
        M     = LP - 1
      DO 4050 L=NN,M
4050    SOUT   = SOUT + VOLIN(L)
        SOUT   = SOUT + SUM - STOR
      GO TO 2400
C
4100    FRAC(LP)= BACK/VOLIN(LP)
        RILP   = KDT - LP
        DETENT(LP) = RILP*DT
        LPREV = LP

```

```

PLUG121
PLUG122
PLUG123
PLUG124
PLUG125
PLUG126
PLUG127
PLUG128
PLUG129
PLUG130
PLUG131
PLUG132
PLUG133
PLUG134
PLUG135
PLUG136
PLUG137
PLUG138
PLUG139
PLUG140
PLUG141
PLUG142
PLUG143
PLUG144
PLUG145
PLUG146
PLUG147
PLUG148
PLUG149
PLUG150
PLUG151
PLUG152
PLUG153
PLUG154
PLUG155
PLUG156
PLUG157
PLUG158
PLUG159
PLUG160
PLUG161
PLUG162
PLUG163
PLUG164
PLUG165
PLUG166
PLUG167
PLUG168
PLUG169
PLUG170
PLUG171
PLUG172
PLUG173
PLUG174
PLUG175
PLUG176
PLUG177
PLUG178
PLUG179
PLUG180

```

|        |                            |         |
|--------|----------------------------|---------|
|        | GO TO 9999                 | PLUG181 |
| C      |                            | PLUG182 |
| 4200   | FRONT = VOKK - BACK        | PLUG183 |
|        | FRAC(JP) = FRONT/VOLIN(JP) | PLUG184 |
|        | RILP = KDT - LP            | PLUG185 |
|        | DETENT(JP) = RILP*DT       | PLUG186 |
|        | GO TO 2300                 | PLUG187 |
| C      |                            | PLUG188 |
| 5000   | LABEL = 5000               | PLUG189 |
|        | GO TO 9999                 | PLUG190 |
| C      |                            | PLUG191 |
| 9999   | DT=DT/60.0                 | PLUG192 |
|        | RETURN                     | PLUG193 |
|        | END                        | PLUG194 |
| C===== |                            | PLUG195 |

|                                                             |         |
|-------------------------------------------------------------|---------|
| SUBROUTINE SROUTE                                           | SROJ 1  |
| COMMON /STBK/QIN(150),BODIN(150),SUSIN(150),COLIN(150),     | SROU 2  |
| * QINST,QCUST,QINSTL,CQUSTL,STORL,QOUTO,STORO,              | SROU 3  |
| * ISPRIN,IPOL,DEPMAX,QCMAX,DEPTH,                           | SROU 4  |
| * ATERM(11),AO2DT2(11),BDEPTH(11),BSTUR(11),                | SROU 5  |
| * DUMSTR(11),DUMDEP(11),                                    | SROJ 6  |
| * VOLIN(150),VOLOUT(150),STOR,CUMIN,CUMOUT,                 | SROJ 7  |
| * SBOD,SSS,SCOL,BODOUT,SUSOUT,COLOUT,                       | SROU 8  |
| * ISTMCD,ISTTYP,ISTOUT,                                     | SROJ 9  |
| * QPUMP,DSTART,DSTOP,                                       | SROU 10 |
| * DTON,STORMX,DTPUMP,DTMORE,STORF,APLAN,                    | SROJ 11 |
| * CLAND,CSTOR,CPS,CTOTAL,CPCUYD,CPACRE,                     | SROJ 12 |
| * LP,JP,LPREV,LABEL,DETENT(150),FRAC(150)                   | SROJ 13 |
| COMMON /TBLK/DT,MOT,KOT                                     | SROU 14 |
| DIMENSION DUMTRM(11),DUMAO2(11)                             | SROU 15 |
| C                                                           | SROJ 16 |
| DT=DT*60.0                                                  | SROJ 17 |
| C                                                           | SROJ 18 |
| C INPUTS ARE DT, QINST, KOT, STORO,                         | SROU 19 |
| C QOUTO, ATERM(11), AO2DT2(11)                              | SROU 20 |
| C OUTPUTS ARE QOUST, STORE(150)                             | SROJ 21 |
| C NOTE.. ALL FLOWS ARE IN CFS.                              | SROU 22 |
| C                                                           | SROJ 23 |
| IF (KOT .GT. 1) GO TO 1000                                  | SROU 24 |
| C                                                           | SROU 25 |
| C INITIALISE                                                | SROJ 26 |
| C                                                           | SROJ 27 |
| QINSTL = 0.0                                                | SROU 28 |
| QOUSTL = QOUTO                                              | SROU 29 |
| STORL = STORO                                               | SROU 30 |
| DT2 = DT/2.0                                                | SROJ 31 |
| C                                                           | SROU 32 |
| 1000 TERM = (QINSTL + CINST)*DT2 - (QOUSTL*DT2              | SROU 33 |
| * - STORL)                                                  | SROU 34 |
| IF (TERM .GE. 0.0) GO TO 2000                               | SROU 35 |
| WRITE(6,601) TERM, KOT                                      | SROU 36 |
| 601 FORMAT(' ', '** RESET TERM =', F12.5, ' TO ZERO ', I10) | SROJ 37 |
| TERM = C.0                                                  | SROU 38 |
| C                                                           | SROU 39 |
| C FIND 'O2DT2' CORRESPONDING TO 'TERM' ABOVE,               | SROU 40 |
| C BY LINEAR INTERPOLATION.                                  | SROU 41 |
| C                                                           | SROJ 42 |
| 2000 DO 2050 I=1,11                                         | SROU 43 |
| DUMTRM(I) = ATERM(I)                                        | SROU 44 |
| 2050 DUMAO2(I) = AO2DT2(I)                                  | SROU 45 |
| C                                                           | SROU 46 |
| CALL INTERP(DUMTRM,DUMAO2,11,TERM,O2DT2,KFLAG)              | SROU 47 |
| C                                                           | SROU 48 |
| IF (KFLAG .NE. 0) GO TO 901                                 | SROJ 49 |
| STOR = TERM - O2DT2                                         | SROU 50 |
| QOUST = O2DT2/DT2                                           | SROU 51 |
| CC WRITE(6,61)                                              | SROU 52 |
| C                                                           | SROU 53 |
| C INITIALISE FOR NEXT TIME-STEP                             | SROU 54 |
| C                                                           | SROU 55 |
| QINSTL = CINST                                              | SROU 56 |
| QOUSTL = QOUST                                              | SROJ 57 |
| STORL = STOR                                                | SROU 58 |
| GO TO 9999                                                  | SROU 59 |
| C                                                           | SROJ 60 |



| C----- ERROR MESSAGES |                                                                        | SROU 61 |
|-----------------------|------------------------------------------------------------------------|---------|
| C                     |                                                                        | SROU 62 |
|                       | 901 IF (KFLAG .EQ. 10) GO TO 902                                       | SROU 63 |
|                       | WRITE(6,691)                                                           | SROU 64 |
|                       | 691 FORMAT('O *** TERMINATE - INPUT TO INTERP PROCEDURE IS LESS THAN L | SROU 65 |
|                       | *OWEST VALUE ON CURVE (IN SUBRT. ROUTE)')                              | SROU 66 |
|                       | STOP                                                                   | SROU 67 |
| C                     |                                                                        | SROU 68 |
|                       | 902 WRITE(6,692)                                                       | SROU 69 |
|                       | 692 FORMAT('O *** TERMINATE - INPUT TO INTERP PROCEDURE IS GREATER THA | SROU 70 |
|                       | *N LARGEST VALUE ON CURVE (IN SUBRT. ROUTE)')                          | SROU 71 |
|                       | STOP                                                                   | SROU 72 |
| C                     |                                                                        | SROU 73 |
|                       | 9999 DT=DT/60.0                                                        | SROU 74 |
|                       | RETURN                                                                 | SROU 75 |
|                       | END                                                                    | SROU 76 |
| C=====                |                                                                        | SROU 77 |

|       |                                                                |         |
|-------|----------------------------------------------------------------|---------|
|       | SUBROUTINE SPRINT                                              | SPRI 1  |
| C     | COPIED BY EJJ FROM RUN.AND.TRAN OF 17 AUG 70                   | SPRI 2  |
|       | COMMON /TBLK/DT,NDT,KDT                                        | SPRI 3  |
|       | COMMON /BLK2/QQARR(150),BDARR(150)                             | SPRI 4  |
|       | COMMON /BLK3/JS,JNS,NPOLL,SSARR(150),COARR(150),POLL(160,5,5), | SPRI 5  |
|       | * QQ(160,5)                                                    | SPRI 6  |
|       | DIMENSION DUM(150)                                             | SPRI 7  |
| C     | PRINT INFLOWS                                                  | SPRI 8  |
|       | WRITE(6,921)                                                   | SPRI 9  |
|       | WRITE(6,911)(I,I=1,10)                                         | SPRI 10 |
|       | WRITE(6,912) JNS,(QQARR(KDT),KDT=1,NDT)                        | SPRI 11 |
|       | WRITE(6,922)                                                   | SPRI 12 |
|       | WRITE(6,911)(I,I=1,10)                                         | SPRI 13 |
|       | DO 217 K=1,NPOLL                                               | SPRI 14 |
|       | GO TO (30,33,36), K                                            | SPRI 15 |
| 30    | WRITE(6,960)                                                   | SPRI 16 |
|       | DO 31 KDT=1,NDT                                                | SPRI 17 |
| 31    | DUM(KDT) = BDARR(KDT)/DT                                       | SPRI 18 |
| 32    | WRITE(6,912) JNS,(DUM(KDT),KDT=1,NDT)                          | SPRI 19 |
|       | GO TO 217                                                      | SPRI 20 |
| 33    | WRITE(6,961)                                                   | SPRI 21 |
|       | DO 34 KCT=1,NDT                                                | SPRI 22 |
| 34    | DUM(KCT) = SSARR(KCT)/DT                                       | SPRI 23 |
|       | GO TO 32                                                       | SPRI 24 |
| 36    | WRITE(6,963)                                                   | SPRI 25 |
|       | DO 37 KCT=1,NDT                                                | SPRI 26 |
| 37    | DUM(KCT) = COARR(KCT)/DT                                       | SPRI 27 |
|       | WRITE(6,967) JNS,(DUM(KCT),KCT=1,NDT)                          | SPRI 28 |
| 217   | CONTINUE                                                       | SPRI 29 |
| C     | PRINT OUTFLOWS                                                 | SPRI 30 |
| 29    | WRITE(6,923)                                                   | SPRI 31 |
|       | WRITE(6,911)(I,I=1,10)                                         | SPRI 32 |
|       | WRITE(6,912) JNS,(QU(KDT,JS),KDT=1,NDT)                        | SPRI 33 |
| 300   | WRITE(6,924)                                                   | SPRI 34 |
|       | WRITE(6,911)(I,I=1,10)                                         | SPRI 35 |
|       | DO 219 K=1,NPOLL                                               | SPRI 36 |
|       | GO TO (40,42,46),K                                             | SPRI 37 |
| 40    | WRITE(6,960)                                                   | SPRI 38 |
|       | GO TO 161                                                      | SPRI 39 |
| 42    | WRITE(6,961)                                                   | SPRI 40 |
|       | GO TO 161                                                      | SPRI 41 |
| 46    | WRITE(6,963)                                                   | SPRI 42 |
| 161   | CONTINUE                                                       | SPRI 43 |
|       | IF(K.NE.3) WRITE(6,912) JNS,(POLL(KDT,K,JS),KDT=1,NDT)         | SPRI 44 |
|       | IF(K.EQ.3) WRITE(6,967) JNS,(POLL(KDT,K,JS),KDT=1,NDT)         | SPRI 45 |
| 219   | CONTINUE                                                       | SPRI 46 |
| C     | FIND TOTAL OUTPUT OF SS AND BOD IN POUNDS                      | SPRI 47 |
| CC    | TSSOUT=0.0                                                     | SPRI 48 |
| CC    | TBODOT=0.0                                                     | SPRI 49 |
| CC    | DO 1000 J=1,NDT                                                | SPRI 50 |
| CC    | TSSOUT=TSSOUT+OUT2(NNPE,J,2)*DT/60.                            | SPRI 51 |
| CC    | TBODUT=TBCOCT+OUT2(NNPE,J,1)*DT/60.                            | SPRI 52 |
| C1000 | CONTINUE                                                       | SPRI 53 |
|       | WRITE(6,924)                                                   | SPRI 54 |
|       | WRITE(6,911)(I,I=1,10)                                         | SPRI 55 |
|       | DO 236 K=1,NPOLL                                               | SPRI 56 |
|       | GO TO (50,52,56),K                                             | SPRI 57 |
| 50    | WRITE(6,966)                                                   | SPRI 58 |
|       | GO TO 234                                                      | SPRI 59 |
| 52    | WRITE(6,969)                                                   | SPRI 60 |

|                                                                          |         |
|--------------------------------------------------------------------------|---------|
| GO TO 234                                                                | SPRI 61 |
| 56 WRITE(6,966)                                                          | SPRI 62 |
| 234 CONTINUE                                                             | SPRI 63 |
| DO 235 KDT=1,NDT                                                         | SPRI 64 |
| IF(QO(KDT,JS).EQ.0.0) GO TO 237                                          | SPRI 65 |
| IF(K.NE.3) DUM(KDT) = POLL(KDT,K,JS)*267.5/QO(KDT,JS)                    | SPRI 66 |
| IF(K.EQ.3) DUM(KDT) = POLL(KDT,K,JS)/(QO(KDT,JS)*17040.0)                | SPRI 67 |
| GO TO 238                                                                | SPRI 68 |
| 237 DUM(KDT)=0.0                                                         | SPRI 69 |
| 238 CONTINUE                                                             | SPRI 70 |
| 235 CONTINUE                                                             | SPRI 71 |
| IF(K.NE.3) WRITE(6,912)JNS,(DUM(KDT),KDT=1,NDT)                          | SPRI 72 |
| IF(K.EQ.3) WRITE (6,967)JNS,(DUM(KDT),KDT=1,NDT)                         | SPRI 73 |
| 236 CONTINUE                                                             | SPRI 74 |
| CC WRITE(C6,965) JNS,TSSOUT,JNS,TBODOT                                   | SPRI 75 |
| CC965 FORMAT('1','0',5X,'TOTAL POUNDS OF SUSPENDED SOLIDS OUTPJT FROM EL | SPRI 76 |
| CC *EMENT ' ,I4,' =',F10.2//' ',5X,'TOTAL POUNDS OF FIVE-DAY BOD OUTPUTS | SPRI 77 |
| CC * FROM ELEMENT ' ,I4,' =',F10.2////)                                  | SPRI 78 |
| 311 RETURN                                                               | SPRI 79 |
| 911 FORMAT(' EXTERNAL'/' ELEMENT TIME STEP'/' NUMBER', 3X,               | SPRI 80 |
| * 10(I6,4X))                                                             | SPRI 81 |
| 912 FORMAT('0',I6,4X,10F10.3,'/ ' , (10X,10F10.3))                       | SPRI 82 |
| 921 FORMAT('1', /, ' ', 50X, 'INLET HYDROGRAPH - CFS')                   | SPRI 83 |
| 922 FORMAT('0', /, ' ', 52X, 'INLET POLLUTOGRAPHS')                      | SPRI 84 |
| 923 FORMAT('C', /, ' ', 45X, 'TREATED OUTFLOW HYDROGRAPH - CFS')         | SPRI 85 |
| 924 FORMAT('C', /, ' ', 46X, 'TREATED OUTFLOW POLLUTOGRAPHS')            | SPRI 86 |
| 960 FORMAT('0', 50X, '*** BOD IN LB/MIN ***')                            | SPRI 87 |
| 961 FORMAT('C', 44X, '*** SUSPENDED SOLIDS IN LB/MIN ***')               | SPRI 88 |
| 963 FORMAT('0', 46X, '*** COLIFORMS IN MPN/MIN ***')                     | SPRI 89 |
| 966 FORMAT('0', 45X, '***COLIFORMS IN MPN/100ML ***')                    | SPRI 90 |
| 967 FORMAT('0', I6, 4X, 1P10E10.2, /, ' ', (10X, 10E10.2))               | SPRI 91 |
| 968 FORMAT('C', 50X, '*** BOD IN MG/L ***')                              | SPRI 92 |
| 969 FORMAT('0', 44X, '*** SUSPENDED SOLIDS IN MG/L ***')                 | SPRI 93 |
| END                                                                      | SPRI 94 |
| C=====                                                                   | SPRI 95 |

|   |                                                                        |         |
|---|------------------------------------------------------------------------|---------|
|   | SUBROUTINE TRCOST                                                      | TRCD 1  |
|   | COMMON CCNVER,KHOUR,KMIN,L,KMCD,NFLAG,BIG,HEAD1,HEAD2,                 | TRCD 2  |
|   | * QDESYN,CQIF,WAOU(7),WAIN(7),QQOU(7),QQIN(7),WARM(7),QORL,            | TRCD 3  |
|   | * BDIR,CCRM(7),BDU(7),BDIN(7),BCU(7),BCIN(7),BDRL,                     | TRCD 4  |
|   | * SSIF,SSIN(7),SSOU(7),SCOU(7),SCIN(7),SSRM(7),SSRL,COIF,CORL,         | TRCD 5  |
|   | * ADEPTH(11),AASURF(11),ITREAT(7),ISTOR,IPRINT,ICOST,HRFD,             | TRCD 6  |
|   | * MODS12,ICHEM,ICL2,SCREEN,QQIFMX,DESF,IRANGE,KNTOF,TRIBA,SEDA,        | TRCD 7  |
|   | * SQM,SREFFH,BREFFH,NUNITH,UAREAH,UPRAMA,ICHEMH,HM,VOLDAF,ITABLE,      | TRCD 8  |
|   | * MUDCST,TCTCST,RECIRC,OVRODAF,TSURFA,OVRSED,NSED,JM(7),WTRMT5,        | TRCD 9  |
|   | * NSCRN,SCRCAP,SUAREA,FAREAB,NMS,AREAMS,VOLCON,VOLSED,ALUMUT,          | TRCD 10 |
|   | * BDIRT(7),SSINT(7),WAIN(7),BDOUT(7),SSOUT(7),WAOUT(7),                | TRCD 11 |
|   | * WARM(7),BDRMT(7),SSRMT(7),CHEMUT(8),CL2UT(8),QMOD(20),WTRMT1,        | TRCD 12 |
|   | * QQRMT(7),QQOUT(7),QQRMMX(7),QQOUMX(7),QQRMMN(7),QQOUMN(7),           | TRCD 13 |
|   | * BCRMT(7),BCOUT(7),BCRMMX(7),BCOUMX(7),BCRMMN(7),BCOUMN(7),           | TRCD 14 |
|   | * SCRMT(7),SCOUT(7),SCRMMX(7),SCOUMX(7),SCRMMN(7),SCOUMN(7),           | TRCD 15 |
|   | * BDRM(7),BCRM(7),SCRM(7)                                              | TRCD 16 |
|   | COMMON /BLK1/NAME(4,21)                                                | TRCD 17 |
|   | COMMON /STBK/QIN(150),BDIN(150),SUSIN(150),COLIN(150),                 | TRCD 18 |
|   | * QINST,CCST,QINSTL,COUSTL,STORL,QOUTO,STORO,                          | TRCD 19 |
|   | * ISPRI,IPOL,DEPMAX,QCMAX,DEPTH,                                       | TRCD 20 |
|   | * ATERM(11),AQ2DT2(11),BDEPTH(11),BSTOR(11),                           | TRCD 21 |
|   | * DUMSTR(11),DUMDEP(11),                                               | TRCD 22 |
|   | * VOLIN(150),VOLOUT(150),STOR,CUMIN,CUMOUT,                            | TRCD 23 |
|   | * SBOD,SSS,SCOL,BODOUT,SUSOUT,COLOUT,                                  | TRCD 24 |
|   | * ISTMCD,ISTTYP,ISTOUT,                                                | TRCD 25 |
|   | * QPUMP,DSTART,DSTOP,                                                  | TRCD 26 |
|   | * DTOR,STORMX,DTPUMP,DTMORE,STORF,APLAN,                               | TRCD 27 |
|   | * CLANC,CSTOR,CPS,CTOTAL,CPCUYD,CPCRE,                                 | TRCD 28 |
|   | * LP,JP,LPREV,LABEL,DETENT(150),FRAC(150)                              | TRCD 29 |
|   | DIMENSION ENR(15),NYEAR(7),ACJ(7),CAPCST(7),ALAND(7),CLACST(7),        | TRCD 30 |
|   | * ANACST(7),ALACST(7),BASICH(7),CHCOST(7),OTHCST(7),                   | TRCD 31 |
|   | * IENR(15),CL2CST(7)                                                   | TRCD 32 |
| C | IF (ICCST .LT. 1) GO TO 9999                                           | TRCD 33 |
| C | WRITE(6,601)                                                           | TRCD 34 |
|   | 601 FORMAT('SUMMARY OF TREATMENT COSTS' ,/)                            | TRCD 35 |
| C |                                                                        | TRCD 36 |
| C | READ AND WRITE GENERAL COST PARAMETERS                                 | TRCD 37 |
| C |                                                                        | TRCD 38 |
|   | REAC (5,501) RATEPC, NYRS, MODYR, SITEF                                | TRCD 39 |
|   | 501 FORMAT(F10.2, 2I10, F10.4)                                         | TRCD 40 |
|   | IF (MODYR .LT. 1970) GO TO 9100                                        | TRCD 41 |
|   | IF (MODYR .GT. 1980) GO TO 9200                                        | TRCD 42 |
| C | DEFAULT VALUES                                                         | TRCD 43 |
|   | IF (RATEPC .EQ. 0.0) RATEPC = 7.00                                     | TRCD 44 |
|   | IF (NYRS .EQ. 0) NYRS = 25                                             | TRCD 45 |
|   | IF (SITEF .EQ. 0.0) SITEF = 1.00                                       | TRCD 46 |
|   | CRC = (1.0 + 0.01*RATEPC)**NYRS                                        | TRCD 47 |
|   | CRF = 0.01*RATEPC*CRC/(CRC - 1.0)                                      | TRCD 48 |
|   | KENR= MODYR - 1969                                                     | TRCD 49 |
|   | READ(5,502) (IENR(I), I=1,11)                                          | TRCD 50 |
|   | 502 FORMAT(8I10)                                                       | TRCD 51 |
|   | WRITE(6,603)                                                           | TRCD 52 |
|   | 603 FORMAT('C', 15X, 'ASSUMED FUTURE ENGINEERING NEWS RECORD INDICES', | TRCD 53 |
|   | * /, ' ', 15X, 'CONSTRUCTION - 20 CITY AVERAGE', /,                    | TRCD 54 |
|   | * 'C', 23X, 'YEAR', 5X, 'ENR INDEX', /)                                | TRCD 55 |
|   | DO 700 I=1,11                                                          | TRCD 56 |
|   | ENR(I) = IENR(I)                                                       | TRCD 57 |
|   | KYEAR = 1969 + I                                                       | TRCD 58 |
|   |                                                                        | TRCD 59 |
|   |                                                                        | TRCD 60 |

|                                                        |         |
|--------------------------------------------------------|---------|
| WRITE(6,605) KYEAR, IENR(1)                            | TRCO 61 |
| 605 FORMAT(' ', 21X, I6, 5X, I6)                       | TRCO 62 |
| 700 CONTINUE                                           | TRCO 63 |
| C                                                      | TRCO 64 |
| C                                                      | TRCO 65 |
| C                                                      | TRCO 66 |
| READ(5,504) UCLAND,UCPOWR,UCCL2,UCPOLY,UCALUM          | TRCO 67 |
| 504 FORMAT(F10.0,F10.5,3F10.2)                         | TRCO 68 |
| C                                                      | TRCO 69 |
| C                                                      | TRCO 70 |
| C                                                      | TRCO 71 |
| C                                                      | TRCO 72 |
| C                                                      | TRCO 73 |
| C                                                      | TRCO 74 |
| C                                                      | TRCO 75 |
| C                                                      | TRCO 76 |
| C                                                      | TRCO 77 |
| C                                                      | TRCO 78 |
| C                                                      | TRCO 79 |
| C                                                      | TRCO 80 |
| C                                                      | TRCO 81 |
| C                                                      | TRCO 82 |
| IF (UCLAND.EQ. 0.0) UCLAND = 20000.0                   | TRCO 83 |
| IF (UCPOWR.EQ. 0.0) UCPOWR = 0.02                      | TRCO 84 |
| IF (UCCL2.EQ. 0.0) UCCL2 = 0.20                        | TRCO 85 |
| IF (UCPOLY.EQ. 0.0) UCPOLY = 1.25                      | TRCO 86 |
| IF (UCALUM.EQ. 0.0) UCALUM = 0.03                      | TRCO 87 |
| WRITE(6,602) RATEPC, NYRS, CRF, MODYR, SITEF           | TRCO 88 |
| 602 FORMAT('0', 15X, 'COST PARAMETERS . .', /,         | TRCO 89 |
| * ' ', 20X, 'INTEREST RATE =', F10.2, ' PERCENT', /,   | TRCO 90 |
| * ' ', 20X, 'AMORTIZATION PERIOD =', I10, ' YEARS', /, | TRCO 91 |
| * ' ', 20X, 'CAP. RECOVERY FACTOR =', F10.4, /,        | TRCO 92 |
| * ' ', 20X, 'YEAR OF SIMULATION =', I10, /,            | TRCO 93 |
| * ' ', 20X, 'SITE LOCATION FACTOR =', F10.4)           | TRCO 94 |
| C                                                      | TRCO 95 |
| WRITE(6,604) UCLAND,UCPOWR,UCCL2,UCPOLY,UCALUM         | TRCO 96 |
| 604 FORMAT('0', 15X, 'UNIT COSTS . .', /,              | TRCO 97 |
| * ' ', 20X, 'LAND =', F10.2, ' \$/ACRE', /,            | TRCO 98 |
| * ' ', 20X, 'POWER =', F11.3, ' \$/KWH', /,            | TRCO 99 |
| * ' ', 20X, 'CHLORINE =', F11.3, ' \$/LB', /,          | TRCO100 |
| * ' ', 20X, 'POLYMERS =', F11.3, ' \$/LB', /,          | TRCO101 |
| * ' ', 20X, 'ALUM =', F10.2, ' \$/LB')                 | TRCO102 |
| C                                                      | TRCO103 |
| C                                                      | TRCO104 |
| C                                                      | TRCO105 |
| DO 900 L=1,7                                           | TRCO106 |
| CAPCST(L) = 0.0                                        | TRCO107 |
| ALAND(L) = 0.0                                         | TRCO108 |
| CLACST(L) = 0.0                                        | TRCO109 |
| ANNCST(L) = 0.0                                        | TRCO110 |
| ALACST(L) = 0.0                                        | TRCO111 |
| BASICM(L) = 0.0                                        | TRCO112 |
| CL2CST(L) = 0.0                                        | TRCO113 |
| CHCOST(L) = 0.0                                        | TRCO114 |
| 900 QTHCST(L) = 0.0                                    | TRCO115 |
| C                                                      | TRCO116 |
| C                                                      | TRCO117 |
| C                                                      | TRCO118 |
| C                                                      | TRCO119 |
| C                                                      | TRCO120 |

CAPCST = CAPITAL COST OF INSTALLED EQUIPMENT, \$  
ALAND = AREA OF LAND REQUIRED FOR THIS EQUIP, ACRES  
CLACST = CAPITAL COST OF LAND REQUIRED, \$  
ANNCST = AMORTIZED COST OF INSTALLED EQUIPMENT, \$/YR

|        |                                                                   |         |
|--------|-------------------------------------------------------------------|---------|
| C      | ALACST = AMORTIZED COST OF LAND REQUIRED, \$/YEAR                 | TRCJ121 |
| C      | BASICM = COST OF MINIMUM MAINTENANCE (NO STORMS), \$              | TRCJ122 |
| C      | CL2CST = COST OF CHLORINE, \$/STORM                               | TRCJ123 |
| C      | CHCOST = COST OF POLYMERS AND ALUM, \$/STORM                      | TRCJ124 |
| C      | OTHCST = ALL OTHER STORM COSTS, \$/STORM                          | TRCJ125 |
| C      |                                                                   | TRCJ126 |
|        | Q = QMGD(KMOD)                                                    | TRCJ127 |
|        | QMGD = Q                                                          | TRCJ128 |
|        | IF (Q .LT. 12.0) Q = 12.0                                         | TRCJ129 |
| C      |                                                                   | TRCJ130 |
| C----- | LEVEL 1 BRANCH (FOR TREATMENT BY BAR RACKS)                       | TRCJ131 |
| C      |                                                                   | TRCJ132 |
|        | 1000 K = ITREAT(1) - 10                                           | TRCJ133 |
|        | L = 1                                                             | TRCJ134 |
|        | GO TO (2000,1200), K                                              | TRCJ135 |
| C      |                                                                   | TRCJ136 |
| C      | 12) FOR BAR RACKS                                                 | TRCJ137 |
| C      |                                                                   | TRCJ138 |
|        | 1200 IF (SCRCAP .LT. 120.0) SCRCAP = 120.0                        | TRCJ139 |
|        | CAPCST(1) = NSCRN*(11000.0 + 1000.0*(SCRCAP-120.0)/240.0)         | TRCJ140 |
|        | * ENR(KENR)/1000.0                                                | TRCJ141 |
| C      | BASED ON JACKSONVILLE, 1959 (ENR = 790), ADJ TO ENR               | TRCJ142 |
|        | IF (QMGD .LE. 100.0) CAPCST(1) = CAPCST(1) +                      | TRCJ143 |
|        | * 1000.0*10.0*QMGD**0.625                                         | TRCJ144 |
|        | * ENR(KENR)/1034.0                                                | TRCJ145 |
|        | IF (QMGD .GT. 100.0) CAPCST(1) = CAPCST(1) +                      | TRCJ146 |
|        | * 1000.0*(QMGD/100.0)*178.0                                       | TRCJ147 |
|        | * ENR(KENR)/1034.0                                                | TRCJ148 |
|        | CAPCST(1) = CAPCST(1)*SITEF                                       | TRCJ149 |
| C      | BASED ON 20 CITIES AV, ASSUMED 1966 (ENR = 1034)                  | TRCJ150 |
|        | ALAND(1) = (2700.0 + 3.0*QDESYN)/43560.0                          | TRCJ151 |
|        | CLACST(1) = ALAND(1)*UCLAND*ENR(KENR)/ENR(1)                      | TRCJ152 |
|        | ANNCST(1) = CAPCST(1)*CRF                                         | TRCJ153 |
|        | ALACST(1) = CLACST(1)*0.01*RATEPC                                 | TRCJ154 |
|        | BASICM(1) = CAPCST(1)*0.01                                        | TRCJ155 |
|        | CL2CST(1) = 0.0                                                   | TRCJ156 |
|        | CHCOST(1) = 0.0                                                   | TRCJ157 |
|        | OTHCST(1) = 20.0 + 0.07*WTRMT1*ENR(KENR)/250.0                    | TRCJ158 |
| C      | BASED ON WORCESTER, 1940 (ENR = 250)                              | TRCJ159 |
| C      |                                                                   | TRCJ160 |
| C----- | LEVEL 2 BRANCH (FOR INLET PUMPING)                                | TRCJ161 |
| C      |                                                                   | TRCJ162 |
|        | 2000 K = ITREAT(2) - 20                                           | TRCJ163 |
|        | L = 2                                                             | TRCJ164 |
|        | GO TO (3000,2200), K                                              | TRCJ165 |
| C      |                                                                   | TRCJ166 |
| C      | 22) FOR INLET PUMPING (STATION)                                   | TRCJ167 |
| C      |                                                                   | TRCJ168 |
|        | 2200 CAPCST(2) = 25000.0*(QMGD**0.58)*SITEF*ENR(KENR)/1314.0      | TRCJ169 |
|        | IF (QMGD .GT. 20.0)                                               | TRCJ170 |
|        | * CAPCST(2) = 16000.0*(QMGD**0.73)*SITEF*ENR(KENR)/1314.0         | TRCJ171 |
|        | IF (QMGD .GT. 100.0)                                              | TRCJ172 |
|        | * CAPCST(2) = QMGD*(16000.0/(100.0**0.27))*SITEF*ENR(KENR)/1314.0 | TRCJ173 |
|        | H = HEAD1                                                         | TRCJ174 |
|        | IF (H .LT. 10.0) H = 10.0                                         | TRCJ175 |
|        | CAPCST(2) = CAPCST(2) + (H-10.0)*(0.0017 +                        | TRCJ176 |
|        | * (ALOG10(QMGD)-0.778)*0.00095/1.824)                             | TRCJ177 |
|        | ALAND(2) = QMGD*(975.0-QMGD)*6.0/(475.0*43560.0)                  | TRCJ178 |
|        | CLACST(2) = ALAND(2)*UCLAND*ENR(KENR)/ENR(1)                      | TRCJ179 |
|        | ANNCST(2) = CAPCST(2)*CRF                                         | TRCJ180 |

|                                                             |         |
|-------------------------------------------------------------|---------|
| ALACST(2) = CLACST(2)*0.01*RATEPC                           | TRC0181 |
| BASICM(2) = CAPCST(2)*0.02                                  | TRC0182 |
| CL2CST(2) = 0.0                                             | TRC0183 |
| CHCOST(2) = 0.0                                             | TRC0184 |
| OTHCSST(2) = 15.00 + UCPOWR*62.4*QDESYN*H/(550.0*1.341*0.7) | TRC0185 |
| C                                                           | TRC0186 |
| C----- LEVEL 3 BRANCH (FOR PRIMARY TREATMENT)               | TRC0187 |
| C                                                           | TRC0188 |
| 3000 K = ITREAT(3) - 30                                     | TRC0189 |
| L = 3                                                       | TRC0190 |
| GO TO (4000,3200,3200,3400,3500), K                         | TRC0191 |
| C                                                           | TRC0192 |
| C 32) & 33) FOR DISSOLVED AIR FLOATATION                    | TRC0193 |
| C                                                           | TRC0194 |
| C BASED ON BAKER ST, SAN FRANCISCO, ADJUSTED                | TRC0195 |
| 3200 IF (QMGD .LE. 15.0) CAPCST(3) = 1.35*QMGD*             | TRC0196 |
| * EXP(2.3026/(0.2075 + 0.0114*ALOG10(QMGD)))                | TRC0197 |
| IF (QMGD .GT. 15.0 .AND. QMGD .LE. 100.0)                   | TRC0198 |
| * CAPCST(3) = 1.35*QMGD*EXP(2.3026*(4.6032 -                | TRC0199 |
| * 0.0559*ALOG10(QMGD)))                                     | TRC0200 |
| IF (QMGD .GT. 100.0) CAPCST(3) = 28000.0*1.35*QMGD          | TRC0201 |
| C BASED ON 20 CITIES AV, 1967 (ENR = 1098)                  | TRC0202 |
| C CAPCST(3) = CAPCST(3)*SITEF*ENR(KENR)/1098.0              | TRC0203 |
| C ADD FOR MECH. SCRAPERS IF NOT PRECEDED BY FINE SCRE       | TRC0204 |
| C IF (ITREAT(3) .EQ. 33) CAPCST(3) = CAPCST(3) + 500.0*QMGD | TRC0205 |
| C ADD FOR CHLORINATOR (GAS FEED ASS'D) IF CHLORINE ADD      | TRC0206 |
| C IF (ICL2 .GT. 0) CAPCST(3) = CAPCST(3) + 500.0*QMGD       | TRC0207 |
| C ADD FOR DOSING EQUIP IF POLYMERS ADDED                    | TRC0208 |
| C?? IF (ICHEM .GT. 0)                                       | TRC0209 |
| ALAND(3) = 10000.0*QMGD/(25.0*43560.0)                      | TRC0210 |
| RQLAND = 2.0*TSURFA/43560.0                                 | TRC0211 |
| IF (ALAND(3) .LT. RQLAND) ALAND(3) = RQLAND                 | TRC0212 |
| CLACST(3) = ALAND(3)*UCLAND*ENR(KENR)/ENR(1)                | TRC0213 |
| ANNCST(3) = CAPCST(3)*CRF                                   | TRC0214 |
| ALACST(3) = CLACST(3)*0.01*RATEPC                           | TRC0215 |
| BASICM(3) = CAPCST(3)*0.02                                  | TRC0216 |
| CL2CST(3) = CL2UT(3)*UCCL2                                  | TRC0217 |
| CHCOST(3) = CHEMUT(3)*UCPOLY                                | TRC0218 |
| OTHCSST(3) = WAIINT(3)*1000.0*0.004 + 15.00                 | TRC0219 |
| IF (ITREAT(3) .EQ. 33) GO TO 3400                           | TRC0220 |
| GO TO 4000                                                  | TRC0221 |
| C                                                           | TRC0222 |
| C 33) & 34) FOR FINE SCREENS                                | TRC0223 |
| C                                                           | TRC0224 |
| C                                                           | TRC0225 |
| 3400 CAPSC = QMGD*12000.0*SITEF*ENR(KENR)/1314.0            | TRC0226 |
| CAPCST(3) = CAPCST(3) + CAPSC                               | TRC0227 |
| ALSC = (Q/6.0)*3000.0/(43560.0*SQR(TQ/6.0))                 | TRC0228 |
| ALAND(3) = ALAND(3) + ALSC                                  | TRC0229 |
| CLASC = ALSC*UCLAND                                         | TRC0230 |
| CLACST(3) = ALAND(3)*UCLAND*ENR(KENR)/ENR(1)                | TRC0231 |
| ANNCST(3) = CAPCST(3)*CRF                                   | TRC0232 |
| ANNSC = CAPSC*CRF                                           | TRC0233 |
| ALACST(3) = CLACST(3)*0.01*RATEPC                           | TRC0234 |
| ALASC = CLASC*0.01*RATEPC                                   | TRC0235 |
| BMSC = CAPSC*0.02                                           | TRC0236 |
| BASICM(3) = BASICM(3) + BMSC                                | TRC0237 |
| CL2CST(3) = 0.0 + CL2CST(3)                                 | TRC0238 |
| CHCOST(3) = 0.0 + CHCOST(3)                                 | TRC0239 |
| OTHSC = WAIINT(3)*1000.0*0.006 + 15.00                      | TRC0240 |
| OTHCSST(3) = OTHCSST(3) + OTHSC                             |         |

|                                                                   |         |
|-------------------------------------------------------------------|---------|
| GO TO 4000                                                        | TRCJ241 |
| C                                                                 | TRC0242 |
| C                                                                 | TRC0243 |
| C                                                                 | TRC0244 |
| 3500 GO TO (3600,3700), ISTOP                                     | TRC0245 |
| C                                                                 | TRCJ246 |
| IN NEW SED TANKS                                                  | TRC0247 |
| 3600 CAPCST(3) = 43000.0*(QMGD*700.0/OVRSED)**0.91                | TRC0248 |
| IF (QMGD .GT. 100.0) CAPCST(3) = 430.0*QMGD*(70000./OVRSED)**0.91 | TRC0249 |
| CAPCST(3) = CAPCST(3)*SITEF*ENR(KENR)/1000.0                      | TRC0250 |
| C                                                                 | TRC0251 |
| BASED ON 20 CITIES AV, 1967 (ENR = 1098), ADJUSTED                | TRC0252 |
| ALAND(3) = Q*1000000.0*2.0/(OVRSED*43560.0)                       | TRC0253 |
| OTHCST(3) = WAINT(3)*1000.0*0.004 + 15.00                         | TRC0254 |
| GO TO 3800                                                        | TRC0255 |
| C                                                                 | TRC0256 |
| IN STORAGE                                                        | TRC0257 |
| 3700 CAPCST(3) = CPCUYD*(STORMX/27.0)*SITEF*ENR(KENR)/1314.0      | TRC0258 |
| ALAND(3) = APLAN/43560.0                                          | TRC0259 |
| OTHCST(3) = WAINT(3)*1000.0*0.005                                 | TRC0260 |
| C                                                                 | TRC0261 |
| IN EITHER                                                         | TRC0262 |
| 3800 CLACST(3) = ALAND(3)*UCLAND*ENR(KENR)/ENR(1)                 | TRC0263 |
| ANNCST(3) = CAPCST(3)*CRF                                         | TRC0264 |
| ALACST(3) = CLACST(3)*0.01*RATEPC                                 | TRC0265 |
| BASICM(3) = CAPCST(3)*0.01                                        | TRC0266 |
| CHCOST(3) = 0.0                                                   | TRC0267 |
| CL2CST(3) = CL2UT(3)*UCCL2                                        | TRC0268 |
| C                                                                 | TRC0269 |
| C-----                                                            | TRC0270 |
| LEVEL 4 BRANCH (FOR SECONDARY TREATMENT)                          | TRC0271 |
| C                                                                 | TRC0272 |
| 4000 K = ITREAT(4) - 40                                           | TRC0273 |
| L = 4                                                             | TRC0274 |
| GO TO (5000,4200,4300), K                                         | TRC0275 |
| C                                                                 | TRC0276 |
| C                                                                 | TRC0277 |
| C                                                                 | TRC0278 |
| 4200 CAPCST(4) = 20000.0                                          | TRC0279 |
| IF (QMGD .LT. 25.0) CAPCST(4) = 30000.0                           | TRC0280 |
| CAPCST(4) = CAPCST(4)*QMGD*SITEF*ENR(KENR)/1314.0                 | TRC0281 |
| C                                                                 | TRC0282 |
| BASED ON 20 CITIES AV, 1970 (ENR = ENR(1))                        | TRC0283 |
| ALAND(4) = 0.50*Q*3000.0/(43560.0*SQR(Q/12.0))                    | TRC0284 |
| IF (QMGD .LT. 25.0) ALAND(4) = ALAND(4)*1.333                     | TRC0285 |
| CLACST(4) = ALAND(4)*UCLAND*ENR(KENR)/ENR(1)                      | TRC0286 |
| ANNCST(4) = CAPCST(4)*CRF                                         | TRC0287 |
| ALACST(4) = CLACST(4)*0.01*RATEPC                                 | TRC0288 |
| BASICM(4) = CAPCST(4)*0.02                                        | TRC0289 |
| CHCOST(4) = 0.0                                                   | TRC0290 |
| CL2CST(4) = 0.0                                                   | TRC0291 |
| C                                                                 | TRC0292 |
| BASED ON HANOVER PARK                                             | TRC0293 |
| OTHCST(4) = WAINT(4)*1000.0*0.006 + 15.00                         | TRC0294 |
| GO TO 5000                                                        | TRC0295 |
| C                                                                 | TRC0296 |
| C                                                                 | TRC0297 |
| C                                                                 | TRC0298 |
| 4300 CAPCST(4) = 0.6*90000.0*QMGD**0.67                           | TRC0299 |
| CAPCST(4) = CAPCST(4)*SITEF*ENR(KENR)/1058.0                      | TRC0300 |
| C                                                                 |         |
| BASED ON 20 CITIES AV, 1967 (ENR = 1058)                          |         |
| ALAND(4) = Q*1000000.0*2.5/(1440.0*20.0*43560.0)                  |         |
| CLACST(4) = ALAND(4)*UCLAND*ENR(KENR)/ENR(1)                      |         |
| ANNCST(4) = CAPCST(4)*CRF                                         |         |
| ALACST(4) = CLACST(4)*0.01*RATEPC                                 |         |
| BASICM(4) = CAPCST(4)*0.02                                        |         |
| CHCOST(4) = CHEMUT(4)*UCPOLY                                      |         |



|                                                                           |          |
|---------------------------------------------------------------------------|----------|
| CL2CST(4) = 0.0                                                           | TRC0301  |
| OTH CST(4) = WAI NT(4)*1000.0*0.010 + 15.00                               | TRC0302  |
| C-----                                                                    | TRC0303  |
| LEVEL 5 BRANCH (FOR EFFLUENT SCREENS)                                     | TRC0304  |
| C                                                                         | TRC0305  |
| 5000 K = ITREAT(5) - 50                                                   | TRC0306  |
| L = 5                                                                     | TRC0307  |
| GO TO (6000,5200), K                                                      | TRC0308  |
| C                                                                         | TRC0309  |
| 52) FOR EFFLUENT SCREENS                                                  | TRC0310  |
| C                                                                         | TRC0311  |
| 5200 IF (QMGD .GE. 100.0) GO TO 5210                                      | TRC0312  |
| CAPCST(5) = QMGD*(5000.0/QMGD**0.7)*SITEF*ENR(KENR)/1314.0                | TRC0313T |
| INSTALLATION BASED ON 20 CITIES AV, 1970                                  | TRC0314  |
| C      CAPCST(5) = CAPCST(5) + 0.7*1000.0*10.0*(QMGD**0.625)*             | TRC0315  |
| *                    SITEF*ENR(KENR)/1034.0                               | TRC0315  |
| C                    CHANNEL WKS BASED ON 20 CITIES AV, ASSUMED 1966 (ENR | TRC0317  |
| GO TO 5220                                                                | TRC0318  |
| 5210 CAPCST(5) = QMGD*200.0*SITEF*ENR(KENR)/1314.0                        | TRC0319  |
| C      INSTALLATION BASED ON 20 CITIES AV, 1970                           | TRC0320  |
| CAPCST(5) = CAPCST(5) + 0.7*1000.0*(QMGD/100.0)*178.0*                    | TRC0321  |
| *                    SITEF*ENR(KENR)/1034.0                               | TRC0322  |
| C                    CHANNEL WKS BASED ON 20 CITIES AV, ASSUMED 1966 (ENR | TRC0323  |
| 5220 ALAND(5) = (2700.0 + 3.0*QDESYN)/43560.0                             | TRC0324  |
| CLACST(5) = ALAND(5)*UCLAND*ENR(KENR)/ENR(1)                              | TRC0325  |
| ANNCST(5) = CAPCST(5)*CRF                                                 | TRC0326  |
| ALACST(5) = CLACST(5)*0.01*RATEPC                                         | TRC0327  |
| BASICM(5) = CAPCST(5)*0.015                                               | TRC0328  |
| CHCOST(5) = 0.0                                                           | TRC0329  |
| CL2CST(5) = 0.0                                                           | TRC0330  |
| OTH CST(5) = 15.0 + 0.30*WTRMT5*1000.0/250.0                              | TRC0331  |
| C                    BASED ON WORCESTER, 1940 (ENR = 250)                 | TRC0332  |
| GO TO 6000                                                                | TRC0333  |
| C-----                                                                    | TRC0334  |
| LEVEL 6 BRANCH (FOR OUTLET PUMPING)                                       | TRC0335  |
| C                                                                         | TRC0336  |
| 6000 K = ITREAT(6) - 60                                                   | TRC0337  |
| L = 6                                                                     | TRC0338  |
| GO TO (7000,6200), K                                                      | TRC0339  |
| C                                                                         | TRC0340  |
| 62) FOR OUTLET PUMPING (STATION)                                          | TRC0341  |
| C                                                                         | TRC0342  |
| 6200 CAPCST(6) = 25000.0*(QMGD**0.58)*SITEF*ENR(KENR)/1314.0              | TRC0343  |
| IF (QMGD .GT. 20.0)                                                       | TRC0344  |
| * CAPCST(6) = 16000.0*(QMGD**0.73)*SITEF*ENR(KENR)/1314.0                 | TRC0345  |
| IF (QMGD .GT. 100.0)                                                      | TRC0346  |
| * CAPCST(6) = QMGD*(16000.0/(100.0**0.27))*SITEF*ENR(KENR)/1314.0         | TRC0347  |
| H = HEAD2                                                                 | TRC0348  |
| IF (H .LT. 10.0) H = 10.0                                                 | TRC0349  |
| CAPCST(6) = CAPCST(6) + (H-10.0)*(0.0017 +                                | TRC0350  |
| *                    (ALOG10(QMGD)-0.778)*0.00095/1.824)                  | TRC0351  |
| ALAND(6) = QMGD*(975.0-QMGD)*6.0/(475.0*43560.0)                          | TRC0352  |
| CLACST(6) = ALAND(6)*UCLAND*ENR(KENR)/ENR(1)                              | TRC0353  |
| ANNCST(6) = CAPCST(6)*CRF                                                 | TRC0354  |
| ALACST(6) = CLACST(6)*0.01*RATEPC                                         | TRC0355  |
| BASICM(6) = CAPCST(6)*0.02                                                | TRC0356  |
| CL2CST(6) = 0.0                                                           | TRC0357  |
| CHCOST(6) = 0.0                                                           | TRC0358  |
| OTH CST(6) = 15.00 + UCPOWER*62.4*QDESYN*H/(550.0*1.341*0.8)              | TRC0359  |
| C                                                                         | TRC0360  |

|        |                                                                 |          |
|--------|-----------------------------------------------------------------|----------|
| C----- | LEVEL 7 BRANCH (FOR CHLORINE CONTACT TIME)                      | TRC0361  |
| C      |                                                                 | TRC0362  |
| 7000   | K = ITREAT(7) - 70                                              | TRC0363  |
|        | L = 7                                                           | TRC0364  |
|        | GO TO (8000,7200), K                                            | TRC0365  |
| C      |                                                                 | TRC0366  |
| C      | 72) FOR CHLORINE CONTACT TANK                                   | TRC0367  |
| C      |                                                                 | TRC0368  |
| 7200   | CAPCST(7) = 18350.0*QMGD**0.628*SITEF*ENR(KENR)/1000.0          | TRC0369T |
| C      | ABOVE COSTS INCLUDE CHLORINATOR                                 | TRC0370T |
| C      | BASED ON AVERAGE R/T SMITH'S FIG. 16 AND M&E COST               | TRC0371T |
| C      | CURVE BY RICH (ENR=1000)                                        | TRC0372T |
|        | ALAND(7) = 1.15*VOLCON/(10.0*43560.0)                           | TRC0374  |
|        | CLACST(7) = ALAND(7)*UCLAND*ENR(KENR)/ENR(1)                    | TRC0375  |
|        | ANNCST(7) = CAPCST(7)*CRF                                       | TRC0376  |
|        | ALACST(7) = CLACST(7)*0.01*RATEPC                               | TRC0377  |
|        | BASICM(7) = CAPCST(7)*0.02                                      | TRC0378  |
|        | CL2CST(7) = CL2UT(7)*UCCL2                                      | TRC0379  |
|        | CHCOST(7) = 0.0                                                 | TRC0380  |
|        | OTHCST(7) = 15.00                                               | TRC0381  |
| C      |                                                                 | TRC0382  |
| C----- | PREPARE AND PRINT COST SUMMARY                                  | TRC0383  |
| C      |                                                                 | TRC0384  |
| 8000   | CPCSTT = 0.0                                                    | TRC0385  |
|        | ALANCT = 0.0                                                    | TRC0386  |
|        | CLCSTT = 0.0                                                    | TRC0387  |
|        | ANCSTT = 0.0                                                    | TRC0388  |
|        | ALCSTT = 0.0                                                    | TRC0389  |
|        | BSICMT = 0.0                                                    | TRC0390  |
|        | C2CSTT = 0.0                                                    | TRC0391  |
|        | CHCSTT = 0.0                                                    | TRC0392  |
|        | OTCSTT = 0.0                                                    | TRC0393  |
|        | DO 8100 L=1,7                                                   | TRC0394  |
|        | CPCSTT = CPCSTT + CAPCST(L)                                     | TRC0395  |
|        | ALANCT = ALANCT + ALAND(L)                                      | TRC0396  |
|        | CLCSTT = CLCSTT + CLACST(L)                                     | TRC0397  |
|        | ANCSTT = ANCSTT + ANNCST(L)                                     | TRC0398  |
|        | ALCSTT = ALCSTT + ALACST(L)                                     | TRC0399  |
|        | BSICMT = BSICMT + BASICM(L)                                     | TRC0400  |
|        | C2CSTT = C2CSTT + CL2CST(L)                                     | TRC0401  |
|        | CHCSTT = CHCSTT + CHCOST(L)                                     | TRC0402  |
| 8100   | OTCSTT = OTCSTT + OTHCST(L)                                     | TRC0403  |
|        | CAPTGT = CPCSTT + CLCSTT                                        | TRC0404  |
|        | ANNTOT = ANCSTT + ALCSTT + BSICMT                               | TRC0405  |
|        | STMTOT = C2CSTT + CHCSTT + CTCSTT                               | TRC0406  |
|        | WRITE(6,681)                                                    | TRC0407  |
|        | 681 FORMAT('C', 39X, 'CAPITAL COSTS', 13X, 'ANNUAL COSTS', 16X, | TRC0408  |
|        | * 'STORM EVENT COSTS')                                          | TRC0409  |
|        | WRITE(6,682)                                                    | TRC0410  |
|        | 682 FORMAT(' ', 26X, '-----',                                   | TRC0411  |
|        | * '-----')                                                      | TRC0412  |
|        | WRITE(6,683)                                                    | TRC0413  |
|        | 683 FORMAT(' ', 11X, 'TREATMENT', 8X, 'LEVEL', 6X, 'INSTAL',    | TRC0414  |
|        | * 'LAND INSTAL LAND MIN MAINT CHLORINE',                        | TRC0415  |
|        | * 'CHEM OTHER')                                                 | TRC0416  |
|        | WRITE(6,684)                                                    | TRC0417  |
|        | 684 FORMAT(' ', 24X, 9('-----'))                                | TRC0418  |
|        | DU 8200 L=1,7                                                   | TRC0419  |
|        | J = JN(L)                                                       | TRC0420  |
| 8200   | WRITE(6,685) (NAME(I,J), I=1,4), L, CAPCST(L), CLACST(L),       | TRC0421  |

|                                                                       |         |
|-----------------------------------------------------------------------|---------|
| * ANNCST(L), ALACST(L), BASICM(L), CL2CST(L),                         | TRC0422 |
| * CHCOST(L), OTHCST(L)                                                | TRC0423 |
| 685 FORMAT(' ', 8X, 4A4, 5X, I2, F14.0, 7F11.0)                       | TRC0424 |
| IF (ITREAT(3) .EQ. 33) WRITE(6,686) CAPSC, CLASC, ANNSC, ALASC,       | TRC0425 |
| * BMSC, OTHSC                                                         | TRC0426 |
| 686 FORMAT(' ', 8X, '( FINE SCREENS )', 5X, '(3A) (' , F9.0,          | TRC0427 |
| * 4(')(' , F9.0), ')(', 2(7X, '0.)(', F9.0, '))' )                    | TRC0428 |
| WRITE(6,684)                                                          | TRC0429 |
| WRITE(6,687) CFCSTT, CLCSTT, ANCSTT, ALCSTT, BSICMT, C2CSTT,          | TRC0430 |
| * CHCSTT, UTCSTT                                                      | TRC0431 |
| 687 FORMAT('0', 26X, 'SUBTOTAL', 8(' \$', F9.0))                      | TRC0432 |
| WRITE(6,682)                                                          | TRC0433 |
| WRITE(6,688) CAPTOT, ANNTOT, STMTOT                                   | TRC0434 |
| 688 FORMAT('0', 26X, 'TOTAL', 9X, '\$', F10.0, 16X, '\$', F10.0, 22X, | TRC0435 |
| * '\$', F10.0)                                                        | TRC0436 |
| WRITE(6,682)                                                          | TRC0437 |
| C                                                                     | TRC0438 |
| CAPTOT = CAPTOT/TRIBA                                                 | TRC0439 |
| ANNTOT = ANNTOT/TRIEA                                                 | TRC0440 |
| STMTOT = STMTOT/TRIBA                                                 | TRC0441 |
| WRITE(6,680) CAPTOT, ANNTOT, STMTOT                                   | TRC0442 |
| 680 FORMAT('0', 26X, 'TOTAL PER', /,                                  | TRC0443 |
| * ' ', 26X, 'TRIB ACRE', 5X, '\$', F10.0, 16X, '\$', F10.0,           | TRC0444 |
| * 22X, '\$', F10.0)                                                   | TRC0445 |
| WRITE(6,682)                                                          | TRC0446 |
| WRITE(6,689) ALANDT                                                   | TRC0447 |
| 689 FORMAT('0', /, '0', 26X, 'TOTAL LAND REQUIREMENT =',              | TRC0448 |
| * F10.2, ' ACRES.')                                                   | TRC0449 |
| GO TO 9999                                                            | TRC0450 |
| C                                                                     | TRC0451 |
| C----- ERROR MESSAGES -----                                           | TRC0452 |
| C                                                                     | TRC0453 |
| 9000 WRITE(6,690) L, ITREAT(L)                                        | TRC0454 |
| 690 FORMAT('0*** ITREAT(' , I1, ') = ' , I2,                          | TRC0455 |
| * ' IS OF A TYPE NOT PRESENTLY MODELED. EXECUTION STOPPED')           | TRC0456 |
| STOP                                                                  | TRC0457 |
| C                                                                     | TRC0458 |
| 9100 WRITE(6,691) MODYR                                               | TRC0459 |
| 691 FORMAT('0*** YEAR OF MODEL =', 16, ' IS BEFORE 1970',             | TRC0460 |
| * ', WHICH IS NOT ACCEPTABLE TO THIS PROGRAM.')                       | TRC0461 |
| STOP                                                                  | TRC0462 |
| C                                                                     | TRC0463 |
| 9200 WRITE(6,692) MODYR                                               | TRC0464 |
| 692 FORMAT('0*** YEAR OF MODEL =', 16, ' IS AFTER 1980',              | TRC0465 |
| * ', WHICH IS NOT ACCEPTABLE TO THIS PROGRAM.')                       | TRC0466 |
| STOP                                                                  | TRC0467 |
| C                                                                     | TRC0468 |
| C                                                                     | TRC0469 |
| 9999 RETURN                                                           | TRC0470 |
| END                                                                   | TRC0471 |
| C=====                                                                | TRC0472 |

|                                                            |         |
|------------------------------------------------------------|---------|
| SUBROUTINE INTERP(X, Y, L, XE, YE, K)                      | INTE 1  |
| DIMENSION X(L),Y(L)                                        | INTE 2  |
| C                                                          | INTE 3  |
| C                                                          | INTE 4  |
| C                                                          | INTE 5  |
| C                                                          | INTE 6  |
| C                                                          | INTE 7  |
| C                                                          | INTE 8  |
| C                                                          | INTE 9  |
| C                                                          | INTE 10 |
| 1000 K = -10                                               | INTE 11 |
| GO TO 9999                                                 | INTE 12 |
| C                                                          | INTE 13 |
| C                                                          | INTE 14 |
| 2000 DO 2050 J=2,L                                         | INTE 15 |
| IF (XE-X(J) ) 4000,3000,2050                               | INTE 16 |
| 2050 CONTINUE                                              | INTE 17 |
| C                                                          | INTE 18 |
| C                                                          | INTE 19 |
| C                                                          | INTE 20 |
| C                                                          | INTE 21 |
| C                                                          | INTE 22 |
| C                                                          | INTE 23 |
| 3000 YE = Y(J)                                             | INTE 24 |
| GO TO 5000                                                 | INTE 25 |
| C                                                          | INTE 26 |
| C                                                          | INTE 27 |
| C                                                          | INTE 28 |
| C                                                          | INTE 29 |
| 4000 YE = Y(J-1) + (Y(J)-Y(J-1))*(XE-X(J-1))/(X(J)-X(J-1)) | INTE 30 |
| 5000 K = 0                                                 | INTE 31 |
| C                                                          | INTE 32 |
| 9999 RETURN                                                | INTE 33 |
| END                                                        | INTE 34 |

```

C=====BLOC 1
      BLOCK DATA                                BLOC 2
      COMMON /BLK1/NAME(4,21)                   BLOC 3
      DATA NAME /                                BLOC 4
* 4H STO, 4HRAGE, 4H ROU, 4HTED , 4HNO S, 4HEP. , 4HSTOR, 4HAGE ,BLOC 5
* 4H B, 4HAR R, 4HACKS, 4H , 4HNO I, 4HNLET, 4H PUM, 4HPING,BLOC 6
* 4H INL, 4HET P, 4HUMPI, 4HNG , 4H BYP, 4HASS , 4HLEVE, 4HL 3 ,BLOC 7
* 4HDISS, 4H AIR, 4H FLO, 4HAT'N, 4HFINE, 4H SCR, 4H + D, 4H.A.F,BLOC 8
* 4H FI, 4HNE S, 4HCREE, 4HNS , 4H SED, 4HIMEN, 4HTATI, 4HON ,BLOC 9
* 4H BYP, 4HASS , 4HLEVE, 4HL 4 , 4H MIC, 4HROST, 4HRAIN, 4HERS ,BLOC 10
* 4HHIGH, 4HRATE, 4H FIL, 4HTERS, 4HNO E, 4HFFL., 4H SCR, 4HEENS,BLOC 11
* 4HEFFL, 4FUENT, 4H SCR, 4HEENS, 4HNO O, 4HUTLE, 4HT PU, 4HMPS ,BLOC 12
* 4H OUT, 4HLEFT , 4HPUMP, 4HTNG . 4HNO C, 4HONTA, 4HCT T, 4HANK ,BLOC 13
* 4H CG, 4HNTAC, 4HT TA, 4HNK , 4H , 4HSTOR, 4HAGE , 4H /BLOC 14
      END                                         BLOC 15
C=====BLOC 16

```

## Section 5

### RECEIVING WATER BLOCK

|                   | <u>Page</u> |
|-------------------|-------------|
| Subroutine INDATA | 203         |
| Subroutine INQUAL | 209         |
| Subroutine LOOPQL | 215         |
| Subroutine OUTPUT | 220         |
| Subroutine PRTOUT | 222         |
| Subroutine QPRINT | 225         |
| Subroutine RECEIV | 227         |
| Subroutine SWFLOW | 228         |
| Subroutine SWQUAL | 240         |
| Subroutine TIDCF  | 243         |
| Subroutine TRIAN  | 247         |

|   |                                                                   |         |
|---|-------------------------------------------------------------------|---------|
|   | SUBROUTINE INDATA                                                 | INDA 1  |
| C |                                                                   | INDA 2  |
| C |                                                                   | INDA 3  |
| C | INPUT DATA                                                        | INDA 4  |
| C | HYDRODYNAMICS PROGRAM                                             | INDA 5  |
| C | SPECIFICATION STATEMENTS                                          | INDA 6  |
| C |                                                                   | INDA 7  |
| C |                                                                   | INDA 8  |
| C | CONTROL                                                           | INDA 9  |
| C |                                                                   | INDA 10 |
|   | COMMON /CENTR/ N5,N6,N20,N21, NTCYC,NQCYC,NHCYC, NT,NQSWRT        | INDA 11 |
|   | 1, DELTQ,DELT,TZERO, ISWCH(10)                                    | INDA 12 |
| C |                                                                   | INDA 13 |
| C | GENERAL                                                           | INDA 14 |
| C |                                                                   | INDA 15 |
|   | COMMON ALPHA(30), NJ,NC, ICYC,KCYC,NCYC, WIND,WDIR,EVAP           | INDA 16 |
|   | 1, PRECP(50),NEXIT                                                | INDA 17 |
| C |                                                                   | INDA 18 |
| C | JUNCTIONS                                                         | INDA 19 |
| C |                                                                   | INDA 20 |
|   | COMMON H(100),HN(100),HT(100),HBAR(100),HAVE(100)                 | INDA 21 |
|   | 1, NCHAN(100,8),IPOINT(100,8),AS(100),VOL(100),X(100),Y(100)      | INDA 22 |
|   | 2, DEP(100),COF(100),QIN(100),QOU(100),QINST(100)                 | INDA 23 |
|   | 3, QINBAR(100),COUBAR(100)                                        | INDA 24 |
| C |                                                                   | INDA 25 |
| C | CHANNELS                                                          | INDA 26 |
| C |                                                                   | INDA 27 |
|   | COMMON LEN(225),NJUNC(225,2),B(225),R(225),A(225),AT(225),AK(225) | INDA 28 |
|   | 1, Q(225),QBAR(225),CAVE(225), V(225),VT(225),VBAR(225)           | INDA 29 |
|   | 2, FWIND(225),NUMCH(225),NTEMP(8)                                 | INDA 30 |
|   | 3,NCLOS(225)                                                      | INDA 31 |
| C |                                                                   | INDA 32 |
| C | PRINTOUT AND PLOTTING                                             | INDA 33 |
| C |                                                                   | INDA 34 |
|   | COMMON NPRT,IPRT, NHPRT,JPRT(50),PRTH(30,50)                      | INDA 35 |
|   | 1, NQPRT,CPRT(50),PRTV(30,50),PRTQ(30,50), IDUM(12),ICOL(10)      | INDA 36 |
|   | 2, LTIME, NPLT,NPDEL,JPLT(50),HPLT(50)                            | INDA 37 |
|   |                                                                   | INDA 38 |
| C |                                                                   | INDA 39 |
| C | STAGE-TIME COEFFICIENTS                                           | INDA 40 |
| C |                                                                   | INDA 41 |
|   | COMMON YY(50) ,TT(50) ,AA(10),XX(10),SXX(10,10),SXY(10)           | INDA 42 |
|   | 1,A1,A2,A3,A4,A5,A6,A7,PERIOD,JGW                                 | INDA 43 |
|   |                                                                   | INDA 44 |
| C | STORMWATER                                                        | INDA 45 |
| C |                                                                   | INDA 46 |
|   | COMMON TITLE(30),NJSW,QE(20,2),JSW(20)                            | INDA 47 |
|   | 2, RAIN(100),INTIME(100),INRAIN,JBOUND(20),JJBOUND                | INDA 48 |
|   |                                                                   | INDA 49 |
| C | TAPES                                                             | INDA 50 |
| C |                                                                   | INDA 51 |
|   | COMMON /TAPES/ INCNT,IOUTCT,JIN(10),JOUT(10),NSCRAT(5)            | INDA 52 |
| C |                                                                   | INDA 53 |
| C | TYPE DESIGNATIONS                                                 | INDA 54 |
| C |                                                                   | INDA 55 |
|   | DATA ASTERK,BLANK /4H****,4H                                      | INDA 56 |
|   | INTEGER CPRT                                                      | INDA 57 |
|   | REAL LEN,INTIME                                                   | INDA 58 |
| C |                                                                   | INDA 59 |
| C | OPTION SWITCH, ISWCH(1)                                           | INDA 60 |
| C |                                                                   |         |

|   |                                                                        |         |
|---|------------------------------------------------------------------------|---------|
| C | ISWCH(1)                                                               | INDA 61 |
| C | IF 1, WILL CALL TIDAL                                                  | INDA 62 |
| C | COEFFICIENTS PROGRAM                                                   | INDA 63 |
| C | ISWCH(2)                                                               | INDA 64 |
| C | IF 1, SUPPRESSES CHANNEL AND                                           | INDA 65 |
| C | NODAL INFORMATION PRINT                                                | INDA 66 |
| C |                                                                        | INDA 67 |
| C | STEP ONE                                                               | INDA 68 |
| C | INITIALIZATION                                                         | INDA 69 |
| C |                                                                        | INDA 70 |
|   | N5=5                                                                   | INDA 71 |
|   | N6=6                                                                   | INDA 72 |
| C |                                                                        | INDA 73 |
|   | N20 ASSIGNED IN RECEIV                                                 | INDA 74 |
|   |                                                                        | INDA 75 |
| C |                                                                        | INDA 76 |
| C | STEP TWO                                                               | INDA 77 |
| C | TITLES, GENERAL CONTROL DATA,                                          | INDA 78 |
| C | AND JUNCTION AND CHANNEL INFOR-                                        | INDA 79 |
| C | MATION                                                                 | INDA 80 |
| C |                                                                        | INDA 81 |
| C | READ TYPE A CARDS                                                      | INDA 82 |
| C | (FIRST TWO CARDS CONTAIN HEAD-                                         | INDA 83 |
| C | INGS FOR HYDRODYNAMICS, SECOND                                         | INDA 84 |
| C | TWO CARDS CONTAIN HEADINGS                                             | INDA 85 |
| C | FUR IDENTIFICATION OF STORMWATER                                       | INDA 86 |
| C | INFORMATION)                                                           | INDA 87 |
| C |                                                                        | INDA 88 |
|   | READ(N5,100) ALPHA                                                     | INDA 89 |
|   | READ(N5,100) TITLE                                                     | INDA 90 |
|   | 100 FORMAT(15A4)                                                       | INDA 91 |
|   | WRITE(N6,102) ALPHA                                                    | INDA 92 |
|   | 102 FORMAT(1H115A4,20X,31HWATER RESOURCES ENGINEERS, INC./1H 15A4,20X, | INDA 93 |
|   | 124HWALNUT CREEK, CALIFORNIA/1H 80X,29HRECEIVING WATER HYDRODYNAMI     | INDA 94 |
|   | 2C5//)                                                                 | INDA 95 |
| C |                                                                        | INDA 96 |
| C |                                                                        | INDA 97 |
| C | READ TYPE B CARDS                                                      | INDA 98 |
| C | SWITCH INFORMATION                                                     | INDA 99 |
|   | READ (N5,104) (ISWCH(1),I=1,10)                                        | INDA100 |
|   | 104 FORMAT (10I5)                                                      | INDA101 |
| C |                                                                        | INDA102 |
| C |                                                                        | INDA103 |
| C | READ TYPE C CARDS                                                      | INDA104 |
| C | CONTROL INFORMATION                                                    | INDA105 |
|   | READ (N5,106) NTCYC,PERIOD,QINT,DELT,TZERO,NHPRT,NQPRT,NPLT,EVAP       | INDA106 |
|   | 1,WIND,WDIR,NQSWRT,AJSW,INRAIN,JGW                                     | INDA107 |
|   | 106 FORMAT (15,4F5.0,3I5,3F5.0,4I5)                                    | INDA108 |
|   | IPERID = PERIOD + 0.1                                                  | INDA109 |
|   | IQINT=QINT*3600.+0.1                                                   | INDA110 |
|   | IDELT = DELT + 0.1                                                     | INDA111 |
|   | NQCYC=(IPERID *3600)/IQINT                                             | INDA112 |
|   | NHCYC = IQINT/IDELT                                                    | INDA113 |
|   | NINT = (IPERID*3600)/IDELT                                             | INDA114 |
|   | NPDEL = (NINT+50)/100                                                  | INDA115 |
| C |                                                                        | INDA116 |
| C |                                                                        | INDA117 |
| C | READ TYPE D CARDS                                                      | INDA118 |
| C | PRECIPITATION IS READ AT THIS                                          | INDA119 |
| C | POINT, RATE IS INCHES PER HOUR,                                        | INDA120 |
| C | TIME IS READ IN MINUTES FROM                                           |         |



|     |                                                                     |                |         |
|-----|---------------------------------------------------------------------|----------------|---------|
| C   |                                                                     | START OF STORM | INDA121 |
| C   |                                                                     |                | INDA122 |
|     | DO 210 A=1,100                                                      |                | INDA123 |
|     | RAIN(N)=0.0                                                         |                | INDA124 |
|     | INTIME(N)=0.0                                                       |                | INDA125 |
| 210 | CONTINUE                                                            |                | INDA126 |
|     | IF (INRAIN.EQ.0) GO TO 215                                          |                | INDA127 |
|     | READ(N5,11C)(RAIN(I),INTIME(I),I=1,INRAIN)                          |                | INDA128 |
| 110 | FORMAT (8F10.0)                                                     |                | INDA129 |
| 215 | CONTINUE                                                            |                | INDA130 |
|     | DELTQ=DELT*FLOAT(NHCYC)                                             |                | INDA131 |
|     | WRITE(N6,112) NTCYC                                                 |                | INDA132 |
| 112 | FORMAT (15HO DAYS SIMULATED,I4)                                     |                | INDA133 |
|     | WRITE(N6,114) NCCYC                                                 |                | INDA134 |
| 114 | FORMAT (29HOWATER QUALITY CYCLES PER DAY,I4)                        |                | INDA135 |
|     | WRITE (N6,116) NHCYC                                                |                | INDA136 |
| 116 | FORMAT (43HOINTEGRATION CYCLES PER WATER QUALITY CYCLE,I4)          |                | INDA137 |
|     | WRITE (6,118) DELT                                                  |                | INDA138 |
| 118 | FORMAT (30HOLENGTH OF INTEGRATION STEP IS,F6.0,8H SECONDS)          |                | INDA139 |
|     | WRITE (6,120) TZERO                                                 |                | INDA140 |
| 120 | FORMAT (13HOINITIAL TIME,F6.2,6H HOURS)                             |                | INDA141 |
|     | WRITE(N6,122)EVAP                                                   |                | INDA142 |
| 122 | FORMAT(18HOEVAPORATION RATE,F5.1,17H INCHES PER MONTH)              |                | INDA143 |
|     | WRITE(N6,124)WIND,WDIR                                              |                | INDA144 |
| 124 | FORMAT(15HOWIND VELOCITY,F5.0,22H MPH WIND DIRECTION,F5.0,19H DE    | INDA145        |         |
|     | IGREES FROM NORTH)                                                  | INDA146        |         |
|     | IF (ISCH(1).NE.1) GO TO 216                                         | INDA147T       |         |
|     | WRITE (N6,126)                                                      | INDA148T       |         |
| 126 | FORMAT (16HOESTURIAL SYSTEM)                                        | INDA148A       |         |
|     | GO TO 218                                                           | INDA149B       |         |
| 216 | CONTINUE                                                            | INDA146C       |         |
|     | WRITE (N6,127)                                                      | INDA148D       |         |
| 127 | FORMAT (19HOSTREAM/LAKE SYSTEM)                                     | INDA148E       |         |
| 218 | CONTINUE                                                            | INDA148F       |         |
|     | WRITE (N6,128) NQSWRT                                               | INDA149        |         |
| 128 | FORMAT (26HOWRITE CYCLE STARTS AT THE,I4,11H TIME CYCLE//)          | INDA150        |         |
|     | IF (INRAIN.LE.0) GO TO 225                                          | INDA151        |         |
|     | WRITE (N6,130)                                                      | INDA152        |         |
| 130 | FORMAT(75HRAIN IN INCHES PER HOUR, AND TIME IN MINUTES, MEASURED    | INDA153        |         |
|     | 1FROM START OF STORM/)                                              | INDA154        |         |
|     | WRITE (N6,131)                                                      | INDA155        |         |
| 131 | FORMAT (15X,8H IN./HR.,2X,8H MINUTES,4X,8H IN./HR.,2X,8H MINUTES,   | INDA156        |         |
|     | 14X,8H IN./HR.,2X,8H MINUTES,4X,8H IN./HR.,2X,8H MINUTES,4X,8H IN./ | INDA157        |         |
|     | 2HR.,2X,8H MINUTES/)                                                | INDA158        |         |
|     | DO 220 I = 1,100,5                                                  | INDA159        |         |
|     | L = MINC(I + 4,100)                                                 | INDA160        |         |
|     | WRITE (N6,132) I, L, (RAIN(J),INTIME(J), J=I,L)                     | INDA161        |         |
| 132 | FORMAT (14,' TO '13, 10F11.3)                                       | INDA162        |         |
| 220 | CONTINUE                                                            | INDA163        |         |
|     | DO 222 I=1,100                                                      | INDA164        |         |
| 222 | INTIME(I)=INTIME(I)*60.                                             | INDA165        |         |
|     | GO TO 230                                                           | INDA166        |         |
| 225 | CONTINUE                                                            | INDA167        |         |
|     | WRITE (N6,133)                                                      | INDA168        |         |
| 133 | FORMAT (23HONO PRECIPITATION INPUT)                                 | INDA169        |         |
| 230 | CONTINUE                                                            | INDA170        |         |
| C   |                                                                     | INDA171        |         |
| C   |                                                                     | INDA172        |         |
| C   | READ TYPE E CARDS                                                   | INDA173        |         |
| C   | JUNCTION NUMBERS FOR DETAILED                                       | INDA174        |         |
|     | PRINTOUT                                                            |                |         |

|     |                                                                  |         |
|-----|------------------------------------------------------------------|---------|
| C   | READ(N5,134)(JPRT(I),I=1,NHPRT)                                  | INDA175 |
| 134 | FORMAT(8I10)                                                     | INDA176 |
|     | WRITE(N6,136)NHPRT,(JPRT(I),I=1,NHPRT)                           | INDA177 |
| 136 | FORMAT (32HOPRINTED OUTPUT AT THE FOLLOWING,I3,10H JUNCTIONS, // | INDA178 |
|     | 1 (10X,16I6))                                                    | INDA179 |
| C   |                                                                  | INDA180 |
| C   |                                                                  | INDA181 |
| C   | READ TYPE F CARDS                                                | INDA182 |
| C   | CHANNEL NUMBERS FOR DETAILED                                     | INDA183 |
| C   | PRINTOUT                                                         | INDA184 |
| C   |                                                                  | INDA185 |
|     | READ(N5,134)(CPRT(I),I=1,NQPRT)                                  | INDA186 |
|     | WRITE(N6,138)NCPRT,(CPRT(I),I=1,NQPRT)                           | INDA187 |
| 138 | FORMAT(//15X,21HAND FOR THE FOLLOWINGI3,9H CHANNELS//(10X,8I10)) | INDA188 |
| C   |                                                                  | INDA189 |
| C   | READ TYPE G CARDS                                                | INDA190 |
| C   | READ THE JUNCTION NUMBERS IF                                     | INDA191 |
| C   | PLOTS ARE REQUESTED, OTHERWISE                                   | INDA192 |
| C   | SKIP THIS READ                                                   | INDA193 |
| C   |                                                                  | INDA194 |
|     | IF (NPLT.NE.0) READ(N5,134) (JPLT(N),N=1,NPLT)                   | INDA195 |
| C   |                                                                  | INDA196 |
| C   | TIDAL OPTION AT THIS POINT                                       | INDA197 |
| C   |                                                                  | INDA198 |
|     | IF (ISWCH(1).NE.1) GO TO 560                                     | INDA199 |
|     | READ (N5,140) KO,NI,MAXIT,NCHTID                                 | INDA200 |
| 140 | FORMAT (4I5)                                                     | INDA201 |
|     | READ (N5,142) (TT(I),YY(I),I=1,NI)                               | INDA202 |
| 142 | FORMAT (8F10.0)                                                  | INDA203 |
|     | CALL TIDCF(KO,NI,MAXIT,NCHTID)                                   | INDA204 |
|     | GO TO 580                                                        | INDA205 |
| 560 | CONTINUE                                                         | INDA206 |
|     | READ (N5,142) A1,A2,A3                                           |         |
|     | WRITE (N6,144) A1,A2,A3                                          |         |
| 144 | FORMAT (3F10.1)                                                  |         |
| 580 | CONTINUE                                                         | INDA209 |
|     | NJ=0                                                             | INDA210 |
| C   |                                                                  | INDA211 |
| C   | READ CARDS FOR                                                   | INDA212 |
| C   | NODAL INFORMATION                                                | INDA213 |
| C   |                                                                  | INDA214 |
|     | DO 620 I=1,100                                                   | INDA215 |
|     | READ(N5,166) J,HEAD,SURF,CF1,CF2,DT,CF,X1,Y1                     | INDA216 |
| 166 | FORMAT(I5,F5.0,-6PF10.0,0P2F5.0,2F10.0,20X,-3P2F5.0)             | INDA217 |
|     | IF(J.GT.100)GO TO 640                                            | INDA218 |
|     | IF(J.GT.NJ)NJ=J                                                  | INDA219 |
|     | H(J)=HEAD                                                        | INDA220 |
|     | AS(J)=SURF                                                       | INDA221 |
|     | QIN(J)=QF1                                                       | INDA222 |
|     | QINST(J)=CF1                                                     | INDA223 |
|     | QOU(J)=QF2                                                       | INDA224 |
|     | X(J)=X1                                                          | INDA225 |
|     | Y(J)=Y1                                                          | INDA226 |
|     | DEP(J)=DT                                                        | INDA227 |
|     | COF(J)=CF                                                        | INDA228 |
| 620 | CONTINUE                                                         | INDA229 |
| 640 | CONTINUE                                                         | INDA230 |
|     | NC=0                                                             | INDA231 |
| C   |                                                                  | INDA232 |
| C   | READ CARDS FOR                                                   | INDA233 |

|     |                                                                                 |                     |         |
|-----|---------------------------------------------------------------------------------|---------------------|---------|
| C   |                                                                                 | CHANNEL INFORMATION | INDA234 |
| C   |                                                                                 |                     | INDA235 |
|     | DO 660 I=1,225                                                                  |                     | INDA236 |
|     | READ(N5,172)N,(NTEMP(K),K=1,4),ALEN,WIDTH,RAD,COEF,VEL                          |                     | INDA237 |
| 172 | FORMAT(5I5,5F10.0)                                                              |                     | INDA238 |
|     | IF (N.GT.225) GO TO 670                                                         |                     | INDA239 |
|     | IF (NTEMP(3).NE.0) GO TO 655                                                    |                     | INDA240 |
|     | NC=NC+1                                                                         |                     | INDA241 |
|     | N=NC                                                                            |                     | INDA242 |
|     | LEN(N)=ALEN                                                                     |                     | INDA243 |
|     | B(N)=WIDTH                                                                      |                     | INDA244 |
|     | A(N)=RAD*WIDTH                                                                  |                     | INDA245 |
|     | R(N)=RAD                                                                        |                     | INDA246 |
|     | AK(N)=CCEF                                                                      |                     | INDA247 |
|     | V(N)=VEL                                                                        |                     | INDA248 |
|     | NJUNC(N,1)=MINO(NTEMP(1),NTEMP(2))                                              |                     | INDA249 |
|     | NJUNC(N,2)=MAXO(NTEMP(1),NTEMP(2))                                              |                     | INDA250 |
|     | K=NJUNC(N,1)                                                                    |                     | INDA251 |
|     | DO 643 J=1,8                                                                    |                     | INDA252 |
|     | IF(IPOINT(K,J).EQ.NJUNC(N,2))GO TO 648                                          |                     | INDA253 |
|     | IF(IPOINT(K,J).EQ.0) GO TO 646                                                  |                     | INDA254 |
| 643 | CONTINUE                                                                        |                     | INDA255 |
| 646 | IPOINT(K,J)=NJUNC(N,2)                                                          |                     | INDA256 |
|     | NCHAN(K,J)=NC                                                                   |                     | INDA257 |
|     | GO TO 660                                                                       |                     | INDA258 |
| 648 | NC=NC-1                                                                         |                     | INDA259 |
|     | M=NCHAN(K,J)                                                                    |                     | INDA260 |
|     | LEN(M)=ALEN                                                                     |                     | INDA261 |
|     | B(M)=B(N)+WIDTH                                                                 |                     | INDA262 |
|     | R(M)=RAD                                                                        |                     | INDA263 |
|     | A(M)=R(M)*B(M)                                                                  |                     | INDA264 |
|     | AK(M)=CCEF                                                                      |                     | INDA265 |
|     | V(M)=VEL                                                                        |                     | INDA266 |
|     | GO TO 660                                                                       |                     | INDA267 |
| 655 | CALL TRIAN(NTEMP(1),NTEMP(2),NTEMP(3),NTEMP(4))                                 |                     | INDA268 |
| 660 | CONTINUE                                                                        |                     | INDA269 |
| 670 | CONTINUE                                                                        |                     | INDA270 |
|     | IF (ISWCH(2).EQ.1) GO TO 674                                                    |                     | INDA271 |
|     | WRITE(N6,170)                                                                   |                     | INDA272 |
| 170 | FORMAT(108H1CHANNEL      LENGTH      WIDTH      AREA      MANNING      VELOCITY |                     | INDA273 |
|     | 1Y    HYD RADIUS      JUNCTIONS AT ENDS      MAX INT /      67H                 |                     | INDA274 |
|     | 2 NUMBER      (FT)      (FT)      (SQ FT)      COEF.      (FPS)      (FT)       |                     | INDA275 |
|     | 3 /)                                                                            |                     | INDA276 |
| 674 | CONTINUE                                                                        |                     | INDA277 |
|     | DO 695 N=1,NC                                                                   |                     | INDA278 |
|     | IF (AK(N).LE.0.0) AK(N)=0.018                                                   |                     | INDA279 |
|     | IF (B(N).GT.0.) GO TO 683                                                       |                     | INDA280 |
|     | K=NJUNC(N,1)                                                                    |                     | INDA281 |
|     | NJUNC(N,1)=0                                                                    |                     | INDA282 |
|     | IDEL=0                                                                          |                     | INDA283 |
|     | DO 682 J=1,8                                                                    |                     | INDA284 |
|     | IF(IPOINT(K,J).EQ.0) GO TO 682                                                  |                     | INDA285 |
|     | IF(IPOINT(K,J).NE.NJUNC(N,2)) GO TO 681                                         |                     | INDA286 |
|     | WRITE(N6,168) N,K,NJUNC(N,2)                                                    |                     | INDA287 |
| 168 | FORMAT (8H CHANNEL,14,8H JOINING,14,4H AND,14,38H DELETED DUE TO Z              |                     | INDA288 |
|     | 1ERO OR NEGATIVE WIDTH)                                                         |                     | INDA289 |
|     | NCHAN(K,J)=0                                                                    |                     | INDA290 |
|     | IPOINT(K,J)=0                                                                   |                     | INDA291 |
|     | NJUNC(N,2)=0                                                                    |                     | INDA292 |
|     | GO TO 695                                                                       |                     | INDA293 |



|   |                                                                           |         |
|---|---------------------------------------------------------------------------|---------|
|   | SUBROUTINE INQUAL                                                         | INQU 1  |
| C |                                                                           | INQU 2  |
| C | INPUT DATA SUBROUTINE                                                     | INQU 3  |
| C |                                                                           | INQU 4  |
| C |                                                                           | INQU 5  |
| C | SPECIFICATION STATEMENTS                                                  | INQU 6  |
| C |                                                                           | INQU 7  |
| C | GENERAL AND CONTROL                                                       | INQU 8  |
| C |                                                                           | INQU 9  |
|   | COMMON /TAPES/ INCNT, IOUTCT, JIN(10), JOUT(10), NSCRAT(5)                | INQU 10 |
| C |                                                                           | INQU 11 |
| C |                                                                           | INQU 12 |
|   | COMMON JGW, NTC, NQCYC, DELTQ, QE, QF, ALPHA(30), TITLSW(30), ICOL(10)    | INQU 13 |
|   | 1, ISWCH(10), REAER(6), DECAY(6), AR(6), XME(6), XMF(6), XMEC(6)          | INQU 14 |
|   | 2, N5, N6, N10, N20, N30, N40, NSTART, XRQD                               | INQU 15 |
| C |                                                                           | INQU 16 |
| C | JUNCTIONS                                                                 | INQU 17 |
| C |                                                                           | INQU 18 |
|   | COMMON NJ, ACHAN(100,8), QIN(100), QOU(100), VOL(100), VOL0(100)          | INQU 19 |
|   | 1, AS(100)                                                                | INQU 20 |
| C |                                                                           | INQU 21 |
| C | CHANNELS                                                                  | INQU 22 |
| C |                                                                           | INQU 23 |
|   | COMMON AC, NJUNC(225,2), Q(225), LEN(225), U(225)                         | INQU 24 |
| C |                                                                           | INQU 25 |
| C | STORMWATER                                                                | INQU 26 |
| C |                                                                           | INQU 27 |
|   | COMMON NJSW, JSW(20)                                                      | INQU 28 |
|   | 1, MJSW, ISW(20), TT(2), CT(6,20,2), INSTM                                | INQU 29 |
| C |                                                                           | INQU 30 |
| C | QUALITY                                                                   | INQU 31 |
| C |                                                                           | INQU 32 |
|   | COMMON KCON, KCOND, C2(6), CS(6), ICON(6), CSAT(6), C(100,6), SJMC(100,6) | INQU 33 |
|   | 1, CMAX(100,6), CMIN(100,6), MADD(100,6), DCDT(100,6), CE(6,20,2)         | INQU 34 |
|   | 2, TE(6), TEP(6), SLOPE(20), CSPIN(100,6), TITLE(6,6), TEU(6)             | INQU 35 |
| C |                                                                           | INQU 36 |
| C | PRINTING                                                                  | INQU 37 |
| C |                                                                           | INQU 38 |
|   | COMMON ACPRT, ITCPRT, LQCPRT, NSTPRT, NQCTOT, ISKIP, MSTPRT, NPRT, KPRT   | INQU 39 |
| C |                                                                           | INQU 40 |
|   | DATA DISOXY /4H(DO) /                                                     | INQU 41 |
| C |                                                                           | INQU 42 |
|   | REAL MADD, LEN                                                            | INQU 43 |
| C |                                                                           | INQU 44 |
|   | N5=5                                                                      | INQU 45 |
|   | N6=6                                                                      | INQU 46 |
| C |                                                                           | INQU 47 |
| C | READ N10, N30, N40 N10 SHOULD BE                                          | INQU 48 |
| C | DRUM OR DISC STORAGE, N30 AND                                             | INQU 49 |
| C | N40 SHOULD BE MAGNETIC TAPE                                               | INQU 50 |
| C | IF USED.                                                                  | INQU 51 |
| C |                                                                           | INQU 52 |
|   | N10 = NSCRAT(2)                                                           | INQU 53 |
|   | N30 = NSCRAT(3)                                                           | INQU 54 |
|   | N40 = NSCRAT(4)                                                           | INQU 55 |
|   | REWIND N10                                                                | INQU 56 |
|   | REWIND N20                                                                | INQU 57 |
| C |                                                                           | INQU 58 |
| C | READ -- ISWCH VALUES                                                      | INQU 59 |
| C |                                                                           | INQU 60 |

|      |                                                                |         |
|------|----------------------------------------------------------------|---------|
| C    | IF THE SWITCH VALUE IS 1, THEN                                 | INQJ 61 |
| C    | THE STATEMENT OF ACTION                                        | INQU 62 |
| C    | CONTROLLED BY THE SWITCH WILL                                  | INQJ 63 |
| C    | OCCUR.                                                         | INQU 64 |
| C    |                                                                | INQJ 65 |
| C    | SWITCH 1 - INPUT INITIAL CONCEN-                               | INQJ 66 |
| C    | TRATIONS FROM TAPE N30.                                        | INQU 67 |
| C    | SWITCH 2 - SKIP PRINTING MAXIMUM                               | INQJ 68 |
| C    | AND MINIMUM CONCENTRATIONS                                     | INQU 69 |
| C    | SWITCH 3 - WRITE CONCENTRATIONS                                | INQU 70 |
| C    | ON TAPE N40 FOR A RESTART.                                     | INQJ 71 |
| C    | (ALSO WRITES HYDRAULIC INFORMA-                                | INQJ 72 |
| C    | TION)                                                          | INQU 73 |
| C    | SWITCH 4 - BOD/DO IS AT LEAST                                  | INQU 74 |
| C    | ONE OF THE CONSTITUENTS.                                       | INQU 75 |
| C    | SWITCH 5 - RECEIVING WATER IS                                  | INQJ 76 |
| C    | TIDALLY INFLUENCED.                                            | INQU 77 |
| C    | SWITCH 10 - THIS SWITCH IS SET                                 | INQJ 78 |
| C    | BY N30 IF A RESTART TAPE IS USED                               | INQU 79 |
| C    | OR IT CAN BE READ IN IF THE                                    | INQU 80 |
| C    | FIRST TIDAL CYCLE ONLY IS                                      | INQJ 81 |
| C    | REPEATED                                                       | INQJ 82 |
| C    |                                                                | INQJ 83 |
|      | READ(5,555) ISWCH                                              | INQU 84 |
| 555  | FORMAT (10I5)                                                  | INQU 85 |
|      | IF(ISWCH(1).EQ.1) REWIND N30                                   | INQJ 86 |
|      | IF(ISWCH(3).EQ.1) REWIND N40                                   | INQU 87 |
|      | DO 11 I=1,10                                                   | INQU 88 |
| 11   | ICOL(I)=I                                                      | INQU 89 |
|      | WRITE(6,6) ICOL,ISWCH                                          | INQJ 90 |
| 6    | FORMAT(' SWITCH SETTINGS'/(10I10))                             | INQU 91 |
| C    |                                                                | INQU 92 |
| C    | INITIALIZATION                                                 | INQU 93 |
| C    |                                                                | INQJ 94 |
|      | DO 5100 I=1,6                                                  | INQU 95 |
|      | ICON(I)=0                                                      | INQU 96 |
|      | DECAY(I)=0.0                                                   | INQU 97 |
|      | CS(I)=C.C                                                      | INQU 98 |
|      | CSAT(I)=0.0                                                    | INQJ 99 |
|      | REAER(I)=0.0                                                   | INQU100 |
|      | DO 5090 J=1,100                                                | INQJ101 |
|      | C(J,I)=0.0                                                     | INQU102 |
|      | MADD(J,I) =0.0                                                 | INQU103 |
| 5090 | CONTINUE                                                       | INQU104 |
| 5100 | CONTINUE                                                       | INQU105 |
|      | DO 5150 I=1,6                                                  | INQJ106 |
|      | TE(I) =0.0                                                     | INQU107 |
|      | TEP(I)=0.0                                                     | INQJ108 |
|      | DO 5140 L=1,20                                                 | INQU109 |
|      | CE(I,L,1)=0.0                                                  | INQU110 |
|      | CE(I,L,2)=0.0                                                  | INQJ111 |
| 5140 | CONTINUE                                                       | INQU112 |
| 5150 | CONTINUE                                                       | INQU113 |
| C    |                                                                | INQJ114 |
| C    | SYSTEM DATA IS READ FROM                                       | INQU115 |
| C    | RECEIVING WATER QUANTITY PROGRAM                               | INQJ116 |
| C    |                                                                | INQU117 |
|      | READ (N2C) TITLSW,ALPHA,NJ,NC,NQCYC,DELTQ,((NCHAN(J,K),K=1,8), | INQU118 |
|      | 1 AS(J),J=1,NJ), (LEN(N),(NJUNC(N,K),K=1,2),N=1,NC)            | INQU119 |
| C    |                                                                | INQU120 |

|   |                                                                     |         |
|---|---------------------------------------------------------------------|---------|
| C |                                                                     | INQJ121 |
| C |                                                                     | INQJ122 |
| C | FLOW INFORMATION IS TRANSFERRED                                     | INQJ123 |
| C | TO FAST STORAGE, DRUM OR DISC.                                      | INQJ124 |
|   | DO 90 I=1,NQCYC                                                     | INQJ125 |
|   | READ(N20) AQ,(Q(N),U(N),N=1,NC),(VOL(J),QIN(J),QOU(J),J=1,NJ)       | INQJ126 |
|   | 90 WRITE(N10) AQ,(Q(N),U(N),N=1,NC),(VOL(J),QIN(J),QOU(J),J=1,NJ)   | INQJ127 |
|   | REWIND N10                                                          | INQJ128 |
| C |                                                                     | INQJ129 |
| C |                                                                     | INQJ130 |
| C | RESTART OPTION - ALL DATA READ                                      | INQJ131 |
| C | FROM TAPE N30.                                                      | INQJ132 |
|   | IF(IISWCH(1).NE.1) GO TO 95                                         | INQJ133 |
|   | IISWCH(10) = 1                                                      | INQJ134 |
|   | READ (A30) JGW,KCON,KCOND,NTC,NPRT,NJSW,TITLE,C2,CS,ICON,CSAT,      | INQJ135 |
|   | 1 REAER,DECAY,XR,XME,XMF,XMED,(VOLO(J),(C(J,K),SUMC(J,K),           | INQJ136 |
|   | 2 MADD(J,K), CMAX(J,K),CMIN(J,K),K=1,KCON),J=1,NJ)                  | INQJ137 |
|   | DO 4285 J=1,NQCYC                                                   | INQJ138 |
|   | READ(N30) AQ,(Q(N),U(N),N=1,NC),(VOL(J),QIN(J),QOU(J),J=1,NJ)       | INQJ139 |
|   | 4285 WRITE(N10) AQ,(Q(N),U(N),N=1,NC),(VOL(J),QIN(J),QOU(J),J=1,NJ) | INQJ140 |
|   | REWIND N30                                                          | INQJ141 |
|   | REWIND N10                                                          | INQJ142 |
|   | KPRT=NTC                                                            | INQJ143 |
|   | NSTART=NTC+1                                                        | INQJ144 |
| C |                                                                     | INQJ145 |
| C |                                                                     | INQJ146 |
| C | THIS IS A READ ONLY ON RESTART                                      | INQJ147 |
|   | READ (5,556) NTC                                                    | INQJ148 |
|   | WRITE (6,1105) NTC                                                  | INQJ149 |
|   | GO TO 39                                                            | INQJ150 |
|   | 95 CONTINUE                                                         | INQJ151 |
| C |                                                                     | INQJ152 |
| C |                                                                     | INQJ153 |
| C | NJSW - NUMBER OF STORM WATER                                        | INQJ154 |
| C | INPUT JUNCTIONS.                                                    | INQJ155 |
| C |                                                                     | INQJ156 |
| C | ITCPT - DAY CYCLE FOR START                                         | INQJ157 |
| C | OF DETAILED QUALITY CYCLE                                           | INQJ158 |
| C | PRINTED INFORMATION.                                                | INQJ159 |
| C |                                                                     | INQJ160 |
| C | NQPRT - QUALITY CYCLE INCREMENT                                     | INQJ161 |
| C | BETWEEN PRINTED CYCLES.                                             | INQJ162 |
| C |                                                                     | INQJ163 |
| C | LQCPRT - TOTAL NUMBER OF QUALITY                                    | INQJ164 |
| C | CYCLES PRINTED (PRESENTLY                                           | INQJ165 |
| C | LIMITED TO 50)                                                      | INQJ166 |
| C |                                                                     | INQJ167 |
| C |                                                                     | INQJ168 |
| C | READ PRINT INFORMATION FOR                                          | INQJ169 |
| C | DETAILED QUALITY PRINTING.                                          | INQJ170 |
|   | READ (N5,555) NJSW,ITCPT,NQPRT,LQCPRT                               | INQJ171 |
|   | WRITE (N6,4100) TITLSW                                              | INQJ172 |
|   | 4100 FORMAT(1H115A4,15A4)                                           | INQJ173 |
|   | WRITE (6,101) ALPHA                                                 | INQJ174 |
|   | 101 FORMAT(1HC15A4,36X,32H WATER RESOURCES ENGINEERS, INC./         | INQJ175 |
|   | 1 1F 15A4,36X,25H WALNUT CREEK, CALIFORNIA/                         | INQJ176 |
|   | 2 1H 96X,' DYNAMIC STORM WATER QUALITY '///)                        | INQJ177 |
| C |                                                                     | INQJ178 |
| C |                                                                     | INQJ179 |
| C | READ GENERAL CONTROL PARAMETERS                                     | INQJ180 |
|   | AND DATA.                                                           |         |

|      |                                                                        |         |
|------|------------------------------------------------------------------------|---------|
| C    |                                                                        | INQU181 |
|      | READ (N5,556) NTC,KCON,NPRT,XRQD                                       | INQU182 |
| 556  | FORMAT (315,F5.0)                                                      | INQJ183 |
|      | WRITE (6,1100) NJ                                                      | INQU184 |
| 1100 | FORMAT (24HOMAXIMUM JUNCTION NUMBER,15)                                | INQU185 |
|      | WRITE (6,1102) NC                                                      | INQU186 |
| 1102 | FORMAT (23HOMAXIMUM CHANNEL NUMBER,15)                                 | INQJ187 |
|      | WRITE (6,1104) NQCYC                                                   | INQJ189 |
| 1104 | FORMAT (33HONUMBER OF QUALITY CYCLES PER DAY,15)                       | INQU189 |
|      | WRITE (6,1105) NTC                                                     | INQU190 |
| 1105 | FORMAT (15HONUMBER OF DAYS,15)                                         | INQU191 |
|      | WRITE (6,1101) KCON                                                    | INQJ192 |
| 1101 | FORMAT (23HONUMBER OF CONSTITUENTS,14)                                 | INQU193 |
|      | WRITE (6,1106) DELTQ                                                   | INQU194 |
| 1106 | FORMAT (45HOLENGTH OF QUALITY INTEGRATION STEP (SECONDS),F7.0)         | INQU195 |
|      | WRITE (6,1110) NPRT                                                    | INQU196 |
| 1110 | FORMAT (16HOPRINT INTERVAL,,14,5H DAYS)                                | INQJ197 |
|      | WRITE (6,1112) XRQD                                                    | INQU198 |
| 1112 | FORMAT (3CHOEXCHANGE REQUIREMENT AT OCEAN,F7.2)                        | INQU199 |
|      | WRITE (N6,4200) NJSW                                                   | INQJ200 |
| 4200 | FORMAT (10HOTHER ARE,13,27H STORMWATER INPUT JUNCTIONS)                | INQJ201 |
|      | MTOTAL = LQCPRT*NQPRT                                                  | INQJ202 |
|      | WRITE (N6,5040) ITCPRT,NQPRT,MTOTAL                                    | INQU203 |
| 5040 | FORMAT (6CHOQUALITY CYCLE CONCENTRATIONS, PRINTOUT STARTS IN TIME      | INQU204 |
|      | 1CYCLE,13,16H , PRINTED EVERY,13,25H HOUR(S), FOR A TOTAL OF,13,6H     | INQJ205 |
|      | 2 HOJRS)                                                               | INQU206 |
| C    |                                                                        | INQJ207 |
| C    | READ QUALITY BOUNDARY DATA                                             | INQU208 |
| C    | KCONO IS NUMBER OF CONSTITUENTS                                        | INQJ209 |
| C    |                                                                        | INQU210 |
|      | KCONO=KCCN                                                             | INQU211 |
|      | DO 103 KC=1,KCONO                                                      | INQJ212 |
|      | READ (5,9) JGW,CS(KC),CSAT(KC),REAER(KC),DECAY(KC),                    | INQU213 |
|      | 1 (TITLE(I,KC),I=1,6)                                                  | INQU214 |
|      | 9 FORMAT (15,F10.0,3E5.0,5X,6A4)                                       | INQJ215 |
|      | WRITE (6,14) KC,(TITLE(I,KC),I=1,6),CS(KC)                             | INQJ216 |
|      | 14 FORMAT (19HICONSTITUENT NUMBER,12,20X,6A4//,19H SINK CONCENTRATION, | INQJ217 |
|      | 1F8.2)                                                                 | INQU218 |
| C    |                                                                        | INQU219 |
| C    |                                                                        | INQJ220 |
| C    | CONSERVATIVE/NONCONSERVATIVE                                           | INQU221 |
| C    | TAGS ARE SET UP.                                                       | INQJ222 |
| C    | KCON IS KCONO PLUS D.O. CONSTI-                                        | INQJ223 |
| C    | UENTS.                                                                 | INQJ224 |
| C    |                                                                        | INQJ225 |
|      | IF(DECAY(KC).LE.0.) GO TO 1130                                         | INQU226 |
|      | KCON=KCCN+1                                                            | INQJ227 |
|      | ICON(KC)=KCCN                                                          | INQU228 |
|      | IC=ICON(KC)                                                            | INQJ229 |
|      | WRITE (6,1113) CSAT(KC)                                                | INQU230 |
| 1113 | FORMAT (24HOXYGEN SATURATION (MGL),F6.2)                               | INQJ231 |
|      | WRITE (6,1114) REAER(KC)                                               | INQU232 |
| 1114 | FORMAT (37HCREAKATION COEFFICIENT (1/SQ FT/DAY),E10.3)                 | INQJ233 |
|      | REAER(KC)=REAER(KC)*DELTQ/86400.                                       | INQU234 |
|      | WRITE (6,1115) DECAY(KC)                                               | INQJ235 |
| 1115 | FORMAT (26HODECAY COEFFICIENT (1/DAY),F6.2)                            | INQU236 |
|      | DECAY(KC)=DECAY(KC)*DELTQ/86400.                                       | INQJ237 |
|      | WRITE (6,1116) IC                                                      | INQU238 |
| 1116 | FORMAT (53HODISSOLVED OXYGEN FOR THIS CONSTITUENT IS CONSTITUENT,      | INQU239 |
|      | 1 12)                                                                  | INQU240 |



|                                                                              |         |
|------------------------------------------------------------------------------|---------|
| DO 1117 I=1,5                                                                | INQU241 |
| 1117 TITLE(I,IC)=TITLE(I,KC)                                                 | INQU242 |
| TITLE(6,IC) = DISOXY                                                         | INQU243 |
| CS(IC)=CSAT(KC)                                                              | INQU244 |
| C2(IC)=CSAT(KC)                                                              | INQU245 |
| DO 1120 J=1,NJ                                                               | INQU246 |
| C(J,IC)=CSAT(KC)                                                             | INQU247 |
| IF(NCHAN(J,1).EQ.0) C(J,IC)=0.                                               | INQU248 |
| 1120 MADD(J,IC)=CSAT(KC)                                                     | INQU249 |
| 1130 CONTINUE                                                                | INQU250 |
| C                                                                            | INQU251 |
| C                                                                            | INQU252 |
| C                                                                            | INQU253 |
| DO 102 J=1,100                                                               | INQU254 |
| READ (5,1220) JTT,CTT,CPP,CTTOX,CPPOX                                        | INQU255 |
| 1220 FORMAT(I5,4F10.0)                                                       | INQU256 |
| IF (JTT.GT.100) GO TO 1021                                                   | INQU257 |
| C(JTT,KC)=CTT                                                                | INQU258 |
| MADD(JTT,KC)=CPP                                                             | INQU259 |
| IF(ICCN(KC).EQ.0) GO TO 102                                                  | INQU260 |
| C(JTT,IC)=CTTOX                                                              | INQU261 |
| MADD(JTT,IC)=CPPOX                                                           | INQU262 |
| 102 CONTINUE                                                                 | INQU263 |
| 1021 CONTINUE                                                                | INQU264 |
| C                                                                            | INQU265 |
| C                                                                            | INQU266 |
| C                                                                            | INQU267 |
| WRITE (6,3050) ICOL                                                          | INQU268 |
| 3050 FORMAT(1H0,34X,42H INITIAL CONCENTRATIONS (MGL), BY JUNCTION, //        | INQU269 |
| 1 9X,10I10/,11H JUNCTION)                                                    | INQU270 |
| DO 3060 I=1,NJ,10                                                            | INQU271 |
| L=MINO(I+9,NJ)                                                               | INQU272 |
| 3060 WRITE (6,111) I,L,(C(J,KC),J=1,L)                                       | INQU273 |
| 111 FORMAT(I4,5H TC ,I3,1X,10E10.4)                                          | INQU274 |
| WRITE (6,3070) ICOL                                                          | INQU275 |
| 3070 FORMAT(1H0,33X,49H MASS LOADINGS (MILLIONS OF LBS/DAY), BY JUNCTION, // | INQU276 |
| 1N,/9X,10I10/,11H JUNCTION)                                                  | INQU277 |
| DO 3080 I=1,NJ,10                                                            | INQU278 |
| L=MINO(I+9,NJ)                                                               | INQU279 |
| 3080 WRITE (6,2111) I,L,(MADD(J,KC),J=1,L)                                   | INQU280 |
| 2111 FORMAT (I4,' TO 'I3,1X,-6P10F10.3)                                      | INQU281 |
| IF(ICCN(KC).EQ.0) GO TO 1040                                                 | INQU282 |
| WRITE (6,3100) ICOL                                                          | INQU283 |
| 3100 FORMAT(1H1,20X,59H INITIAL DISSOLVED OXYGEN CONCENTRATIONS (MGL),       | INQU284 |
| 1BY JUNCTION, //9X,10I10/,11H JUNCTION)                                      | INQU285 |
| DO 3110 I=1,NJ,10                                                            | INQU286 |
| L=MINO(I+9,NJ)                                                               | INQU287 |
| 3110 WRITE (6,111) I,L,(C(J,IC),J=1,L)                                       | INQU288 |
| WRITE (6,3120) ICOL                                                          | INQU289 |
| 3120 FORMAT(1H0,20X,57H DISSOLVED OXYGEN CONCENTRATION OF INFLOW (MGL),      | INQU290 |
| 1 JUNCTION, //9X,10I10/,11H JUNCTION)                                        | INQU291 |
| DO 3130 I=1,NJ,10                                                            | INQU292 |
| L=MINO(I+9,NJ)                                                               | INQU293 |
| 3130 WRITE (6,111) I,L,(MADD(J,IC),J=1,L)                                    | INQU294 |
| 1040 C2(KC)=C(1,KC)                                                          | INQU295 |
| 103 CONTINUE                                                                 | INQU296 |
| C                                                                            | INQU297 |
| C                                                                            | INQU298 |
| C                                                                            | INQU299 |
| C                                                                            | INQU300 |

|                                                                         |                                  |
|-------------------------------------------------------------------------|----------------------------------|
| IF(ISWCH(4),NE.1) GO TO 2290                                            | INQJ301                          |
| C                                                                       | INQJ302                          |
| C                                                                       | INQJ303                          |
| C                                                                       | INQJ304                          |
| C                                                                       | INQJ305                          |
| KSTART=KCCAC+1                                                          | INQJ306                          |
| DO 229 KC=KSTART,KCON                                                   | INQJ307                          |
| DO 229 J=1,NJ                                                           | INQJ308                          |
| 229 MADD(J,KC)=MADD(J,KC)*QIN(J)                                        | INQJ309                          |
| 2290 CONTINUE                                                           | INQJ310                          |
| C                                                                       | INQJ311                          |
| C                                                                       | READ AND INITIALIZE STORMWATER   |
| C                                                                       | INPUT CONCENTRATIONS.            |
| C                                                                       | INQJ313                          |
| C                                                                       | INQJ314                          |
| C                                                                       | READ STORMWATER JUNCTION NUMBERS |
| IF(NJSW.EQ.0) GO TO 4370                                                | INQJ315                          |
| READ (N5,4280) (JSW(L),L=1,NJSW)                                        | INQJ316                          |
| 4280 FORMAT (16I5)                                                      | INQJ317                          |
| DO 4300 K=1,KCGNO                                                       | INQJ318                          |
| C                                                                       | INQJ319                          |
| C                                                                       | INQJ320                          |
| C                                                                       | READ TIME AND LOADING RATE       |
| C                                                                       | INQJ321                          |
| READ (N5,4320) TE(K),(CE(K,L,2), L=1,NJSW)                              | INQJ322                          |
| 4320 FORMAT (8F10.0)                                                    | INQJ323                          |
| TEP(K) = TE(K)/3600.                                                    | INQJ324                          |
| 4300 CONTINUE                                                           | INQJ325                          |
| TIME = 0.0                                                              | INQJ326                          |
| DO 4360 K = 1,KCCNO                                                     | INQJ327                          |
| WRITE (N6,4380) K, TEP(K)                                               | INQJ328                          |
| DO 4360 L = 1,NJSW                                                      | INQJ329                          |
| WRITE (N6,4340) JSW(L),CE(K,L,2)                                        | INQJ330                          |
| 4380 FORMAT(1I010X,19H CONSTITUENT NUMBER,15,5H AT ,F5.2,17H HOURS FROM | INQJ331                          |
| 1H START/)                                                              | INQJ332                          |
| 4340 FORMAT(1H 10X,15,10E10.4)                                          | INQJ333                          |
| 4360 CONTINUE                                                           | INQJ334                          |
| 4370 CONTINUE                                                           | INQJ335                          |
| NSTART = 1                                                              | INQJ336                          |
| KPRT = 1                                                                | INQJ337                          |
| C                                                                       | INQJ338                          |
| C                                                                       | INQJ339                          |
| C                                                                       | NQCTOT AND ISKIP ARE COUNTERS    |
| C                                                                       | FOR QUALITY CYCLE, JUNCTION      |
| C                                                                       | CONCENTRATION PRINTOUT.          |
| C                                                                       | INQJ340                          |
| C                                                                       | INQJ341                          |
| C                                                                       | INQJ342                          |
| C                                                                       | INQJ343                          |
| NQCTOT = 1                                                              | INQJ344                          |
| ISKIP = 1                                                               | INQJ345                          |
| DO 230 J=1,NJ                                                           | INQJ346                          |
| VOLQ(J)=VCL(J)                                                          | INQJ347                          |
| DO 2301 KC = 1,KCONO                                                    | INQJ348                          |
| 2301 MADD(J,KC)=MADD(J,KC)*0.1857                                       | INQJ349                          |
| DO 230 KC = 1,KCON                                                      | INQJ350                          |
| CSPIN(J,KC)=MADD(J,KC)                                                  | INQJ351                          |
| SUMC(J,KC)=0.5*C(J,KC)                                                  | INQJ352                          |
| CMAJ(J,KC)=0.                                                           | INQJ353                          |
| 230 CMIN(J,KC)=C(J,KC)                                                  | INQJ354                          |
| 39 CONTINUE                                                             | INQJ355                          |
| RETURN                                                                  | INQJ356                          |
| END                                                                     | INQJ357                          |

|      |                                                                    |          |
|------|--------------------------------------------------------------------|----------|
|      | SUBROUTINE LOOPCL                                                  | LOOP 1   |
| C    |                                                                    | LOOP 2   |
| C    | QUALITY CYCLE LOOP                                                 | LOOP 3   |
| C    |                                                                    | LOOP 4   |
| C    |                                                                    | LOOP 5   |
| C    | SPECIFICATION STATEMENTS                                           | LOOP 6   |
| C    |                                                                    | LOOP 7   |
| C    | GENERAL AND CONTROL                                                | LOOP 8   |
| C    |                                                                    | LOOP 9   |
|      | COMMON /TAPES/ INCNT,IOUTCT,JIN(10),JOUT(10)                       | LOOP 10  |
|      | COMMON JGW,NTC,NQCYC,DELTQ,QE,QF,ALPHA(30),TITLSW(30),ICOL(10)     | LOOP 11  |
|      | 1,ISWCH(10),REAR(6),DECAY(6),XR(6),XME(6),XMF(6),XMED(6)           | LOOP 12  |
|      | 2,N5,N6,N10,N20,N30,N40,NSTART,XRQD                                | LOOP 13  |
| C    |                                                                    | LOOP 14  |
| C    | JUNCTIONS                                                          | LOOP 15  |
| C    |                                                                    | LOOP 16  |
|      | COMMON NJ,NCHAN(100,8),QIN(100),QOU(100),VOL(100),VOLO(100)        | LOOP 17  |
|      | 1,AS(100)                                                          | LOOP 18  |
| C    |                                                                    | LOOP 19  |
| C    | CHANNELS                                                           | LOOP 20  |
| C    |                                                                    | LOOP 21  |
|      | COMMON NC,NJUNC(225,2),Q(225),LEN(225),U(225)                      | LOOP 22  |
| C    |                                                                    | LOOP 23  |
| C    | STORMWATER                                                         | LOOP 24  |
| C    |                                                                    | LOOP 25  |
|      | COMMON NJSW,JSW(20)                                                | LOOP 26  |
|      | 1,MJSW,ISW(20),TT(2),CT(6,20,2),INSTM                              | LOOP 27  |
| C    |                                                                    | LOOP 28  |
| C    | QUALITY                                                            | LOOP 29  |
| C    |                                                                    | LOOP 30  |
|      | COMMON KCCN,KCCNO,C2(6),CS(6),ICCN(6),CSAT(6),C(100,6),SJMC(100,6) | LOOP 31  |
|      | 1,CMAX(100,6),CMIN(100,6),MADD(100,6),DCDT(100,6),CE(6,20,2)       | LOOP 32  |
|      | 2,TE( 6),TEP( 6),SLOPE(20),CSPIN(100,6),TITLE(6,6),TED(6)          | LOOP 33  |
| C    |                                                                    | LOOP 34  |
| C    | PRINTING                                                           | LOOP 35  |
| C    |                                                                    | LOOP 36  |
|      | COMMON KCPRT,ITCPRT,LQCPRT,NSTPRT,NQCTOT,ISKIP,MSTPRT,VPRT,KPRT    | LOOP 37  |
|      | COMMON /ST1/ TITEL(40)                                             | LOOP 38  |
| C    |                                                                    | LOOP 39  |
| C    |                                                                    | LOOP 40  |
|      | REAL MADD,LEN                                                      | LOOP 41  |
| C    |                                                                    | LOOP 42  |
| C    |                                                                    | LOOP 43  |
|      | IF(INSTM.LT.1) IFLG=0                                              | LOOP 44  |
|      | IF(INSTM.EQ.1) GO TO 190                                           | LOOP 45  |
|      | N21=JIN(INCNT)                                                     | LOOP 46  |
|      | IF(N21.EQ.0) GO TO 190                                             | LOOP 47  |
|      | REWIND N21                                                         | LOOP 48  |
|      | READ (N21) TITEL                                                   | LOOP 49  |
|      | WRITE (6,7093) TITEL                                               | LOOP 50  |
| 7093 | FORMAT(1H1,20A4)                                                   | LOOP 51  |
|      | READ(N21) NSTEPS,MJSW,NCON ,TDELT,TZERO,TAREA                      | LOOP 52X |
|      | TZ=TZERO/3600.                                                     | LOOP 53  |
|      | WRITE(6,7091) NSTEPS,MJSW,NCON,TDELT,TZ,TAREA                      | LOOP 54X |
| 7091 | FORMAT(1H0,' DATA TRANSMITTED FROM INPUT FILE'/                    | LOOP 55  |
|      | 1' NUMBER OF STEPS = 'I5/                                          | LOOP 56  |
|      | 2' NUMBER OF INPUT POINTS = 'I5/                                   | LOOP 57  |
|      | 3' NUMBER OF CONSTITUENTS = 'I5/                                   | LOOP 58  |
|      | 4' TIME INCREMENT = 'F8.0,' SECS'/                                 | LOOP 59  |
|      | 5' INITIAL TIME = 'F10.2,' HRS'/                                   | LOOP 60X |

|      |                                                                    |                                 |          |
|------|--------------------------------------------------------------------|---------------------------------|----------|
| 6*   | TOTAL AREA                                                         | =*F10.2,* ACREA*)               | LOOP 60A |
|      | READ(N21) (ISW(L),L=1,MJSW)                                        |                                 | LOOP 61  |
|      | WRITE (6,6501) (ISW(L),L=1,MJSW)                                   |                                 | LOOP 63  |
| 6501 | FORMAT(110,'INPUT POINTS ARE LISTED BELOW',/(10X,10I70))           |                                 | LOOP 64  |
|      | READ(N21) TT(1),(D,L=1,MJSW),((CT(K,L,1),K=1,NCON),L=1,MJSW)       |                                 | LOOP 65  |
|      | TH=TT(1)/3600.                                                     |                                 | LOOP 67  |
|      | WRITE(6,6503)                                                      |                                 | LOOP 68  |
| 6503 | FORMAT(110,' LOADINGS FROM DATA FILE'//7X,'TIME BUD LBS/MIN SS LBS |                                 | LOOP 69  |
|      | 1/MIN')                                                            |                                 | LOOP 70  |
|      | WRITE(6,6502) TH,((CT(K,L,1),K=1,NCON),L=1,MJSW)                   |                                 | LOOP 71  |
| 6502 | FORMAT(F12.2,10F11.3)                                              |                                 | LOOP 72  |
|      | READ(N21) TT(2),(D,L=1,MJSW),((CT(K,L,2),K=1,NCON),L=1,MJSW)       |                                 | LOOP 73  |
|      | TH=TT(2)/3600.                                                     |                                 | LOOP 75  |
|      | WRITE(6,6502) TH,((CT(K,L,2),K=1,NCON),L=1,MJSW)                   |                                 | LOOP 76  |
|      | NINREC=2                                                           |                                 | LOOP 77  |
|      | I1=1                                                               |                                 | LOOP 78  |
|      | I2=2                                                               |                                 | LOOP 79  |
|      | TIME=IZERO                                                         |                                 | LOOP 80  |
|      | TTP=TIME                                                           |                                 | LOOP 81  |
| 190  | CONTINUE                                                           |                                 | LOOP 82  |
| C    |                                                                    | MAIN LOOP                       | LOOP 83  |
| C    |                                                                    |                                 | LOOP 84  |
|      | DO 548 ICYC=1,NCCYC                                                |                                 | LOOP 85  |
|      | MSTPRT=ICYC                                                        |                                 | LOOP 86  |
| C    |                                                                    |                                 | LOOP 87  |
| C    |                                                                    | READ HYDRAULIC INFORMATION FROM | LOOP 88  |
| C    |                                                                    | FAST DRUM(DISC)                 | LOOP 89  |
| C    |                                                                    |                                 | LOOP 90  |
|      | READ(N10) NO,(Q(N),U(N),N=1,NC),(VOL(J),QIN(J),QUU(J),J=1,NJ)      |                                 | LOOP 91  |
|      | DO 200 J=1,NJ                                                      |                                 | LOOP 92  |
|      | IF(QIN(J).LT.0.) QIN(J)=0.                                         |                                 | LOOP 93  |
| 200  | CONTINUE                                                           |                                 | LOOP 94  |
| C    |                                                                    |                                 | LOOP 95  |
| C    |                                                                    |                                 | LOOP 96  |
| C    |                                                                    | VARIABLE FLOW INTERPOLATION     | LOOP 97  |
| C    |                                                                    | OR AVERAGING                    | LOOP 98  |
| C    |                                                                    |                                 | LOOP 99  |
|      | IF(INSTM.EQ.0) GO TO 4470                                          |                                 | LOOP100  |
|      | TIME=TIME+DELTQ                                                    |                                 | LOOP101  |
|      | IF(N21.EQ.0) GO TO 2050                                            |                                 | LOOP102  |
|      | DO 2010 L=1,MJSW                                                   |                                 | LOOP103  |
|      | J=ISW(L)                                                           |                                 | LOOP104  |
|      | DO 2010 K=1,NCON                                                   |                                 | LOOP105  |
| 2010 | MADD(J,K)=CSPIN(J,K)*DELTQ/.1857                                   |                                 | LOOP106  |
|      | WRITE(6,6503)                                                      |                                 | LOOP107  |
| 2012 | IF(TIME-TT(I2)) 2040,2015,2015                                     |                                 | LOOP108  |
| 2015 | DO 2020 L=1,MJSW                                                   |                                 | LOOP109  |
|      | J=ISW(L)                                                           |                                 | LOOP110  |
|      | DO 2020 K=1,NCON                                                   |                                 | LOOP111  |
| 2020 | MADD(J,K)=MADD(J,K)+CT(K,L,I1)*(TT(I2)-TTP)                        |                                 | LOOP112  |
|      | TTP=TT(I2)                                                         |                                 | LOOP113  |
|      | ITEMP=I2                                                           |                                 | LOOP114  |
|      | I2=I1                                                              |                                 | LOOP115  |
|      | I1=ITEMP                                                           |                                 | LOOP116  |
|      | IF(NINREC-NSTEPS) 2025,2030,2030                                   |                                 | LOOP117  |
| 2025 | READ(N21) TT(I2),(D,L=1,MJSW),((CT(K,L,I2),K=1,NCON),L=1,MJSW)     |                                 | LOOP118  |
|      | TH=TT(I2)/3600.                                                    |                                 | LOOP120  |
|      | WRITE(6,6502) TH,((CT(K,L,I2),K=1,NCON),L=1,MJSW)                  |                                 | LOOP121  |
|      | NINREC=NINREC+1                                                    |                                 | LOOP122  |
|      | GO TO 2012                                                         |                                 | LOOP123  |

|      |                                                                    |         |
|------|--------------------------------------------------------------------|---------|
| 2030 | TT(12)=TT(11)+TT(11)-TT(12)                                        | LOOP124 |
|      | DO 2035 L=1,MJSW                                                   | LOOP125 |
|      | DO 2035 K=1,NCON                                                   | LOOP126 |
| 2035 | CT(K,L,12)=0.                                                      | LOOP127 |
|      | GO TO 2012                                                         | LOOP128 |
| 2040 | DO 2045 L=1,MJSW                                                   | LOOP129 |
|      | J=JSW(L)                                                           | LOOP130 |
|      | DO 2045 K=1,NCON                                                   | LOOP131 |
| 2045 | MADD(J,K)=(MADD(J,K)+CT(K,L,11)*(TIME-TTP))/DELTQ * 267.408        | LOOP132 |
|      | TTP=TIME                                                           | LOOP133 |
| 2050 | IF(NJSW.EQ.0) GO TO 4470                                           | LOOP134 |
|      | DO 4400 K=1,KCONO                                                  | LOOP135 |
|      | IF (TIME.LE.TE(K)) GO TO 4400                                      | LOOP136 |
|      | TEU(K) = TE(K)                                                     | LOOP137 |
|      | DO 4440 L =1,NJSW                                                  | LOOP138 |
|      | CE(K,L,1) = CE(K,L,2)                                              | LOOP139 |
| 4440 | CONTINUE                                                           | LOOP140 |
| C    |                                                                    | LOOP141 |
| C    | READ TIME AND LOADING RATE                                         | LOOP142 |
| C    |                                                                    | LOOP143 |
|      | READ (N5,4320) TE(K),(CE(K,L,2), L = 1,NJSW)                       | LOOP144 |
| 4320 | FORMAT (8F10.0)                                                    | LOOP145 |
|      | TEP(K) = TE(K)/3600.                                               | LOOP146 |
|      | WRITE (N6,4380) K,TEP(K)                                           | LOOP147 |
| 4380 | FORMAT (1H010X,42H INPUT, POUNDS PER DAY, CONSTITUENT NUMBER,15,5H | LOOP148 |
|      | 1 AT ,F6.2,17H HOURS FROM START/)                                  | LOOP149 |
|      | DO 4420 L =1,NJSW                                                  | LOOP150 |
|      | WRITE (N6,4340) JSW(L), CE(K,L,2)                                  | LOOP151 |
| 4340 | FORMAT(1H 10X,15,10E10.4)                                          | LOOP152 |
| 4420 | CONTINUE                                                           | LOOP153 |
| 4400 | CONTINUE                                                           | LOOP154 |
|      | DO 4460 K=1,KCONO                                                  | LOOP155 |
|      | DO 4460 L =1,NJSW                                                  | LOOP156 |
|      | J = JSW(L)                                                         | LOOP157 |
|      | SLOPE(L)=(CE(K,L,2)-CE(K,L,1))/(TE(K)-TEO(K))                      | LOOP158 |
|      | MADD(J,K)=CSPIN(J,K)+(CE(K,L,1)+SLOPE(L)*(TIME-TEO(K)))*0.1857     | LOOP159 |
| 4460 | CONTINUE                                                           | LOOP160 |
| 4470 | CONTINUE                                                           | LOOP161 |
| C    |                                                                    | LOOP162 |
| C    |                                                                    | LOOP163 |
| C    | SET BOUNDARY CONCENTRATION                                         | LOOP164 |
| C    | CONDITIONS                                                         | LOOP165 |
| C    |                                                                    | LOOP166 |
|      | DO 230 KC=1,KCONO                                                  | LOOP167 |
|      | IC=ICCN(KC)                                                        | LOOP168 |
|      | IF(IFLG=1) 208,210,220                                             | LOOP169 |
| 208  | DO 209 KCC=1,KCON                                                  | LOOP170 |
|      | XME(KCC)=1.                                                        | LOOP171 |
| 209  | XMEU(KCC)=0.                                                       | LOOP172 |
| 210  | IF(QIN(JGW)) 211,211,215                                           | LOOP173 |
| 211  | IF(QOU(JGW)) 215,215,221                                           | LOOP174 |
| 215  | IFLG=1                                                             | LOOP175 |
|      | IF(IC.EQ.0) GO TO 218                                              | LOOP176 |
|      | MADD(JGW,IC)=((XRQD/XME(IC)*XMEU(IC)+CSAT(KC))*QIN(JGW))/DELTQ     | LOOP177 |
| 218  | MADD(JGW,KC)=((XRQD/XME(KC)*XMEU(KC)+CS(KC) )*QIN(JGW))/DELTQ      | LOOP178 |
|      | GO TO 230                                                          | LOOP179 |
| 220  | IF(QOU(JGW)) 215,215,226                                           | LOOP180 |
| 221  | IFLG=2                                                             | LOOP181 |
|      | DO 223 KCC=1,KCON                                                  | LOOP182 |
|      | XME(KCC)=1.                                                        | LOOP183 |

|     |                                                                    |         |
|-----|--------------------------------------------------------------------|---------|
| 223 | XME0(KCC)=0.                                                       | LOOP184 |
| 226 | IF(IC.EC.0) GO TO 227                                              | LOOP185 |
|     | XME(IC)=XME(IC)+QOU(JGW)                                           | LOOP186 |
|     | XME0(IC)=XME0(IC)+QOU(JGW)*(C(JGW,IC)-CSAT(KC))                    | LOOP187 |
| 227 | XME(KC)=XME(KC)+QOU(JGW)                                           | LOOP188 |
|     | XME0(KC)=XME0(KC)+QOU(JGW)*(C(JGW,KC)-CS(KC))                      | LOOP189 |
| 230 | CONTINUE                                                           | LOOP190 |
| C   |                                                                    | LOOP191 |
| C   |                                                                    | LOOP192 |
| C   | COMPUTE CONCENTRATION CHANGES                                      | LOOP193 |
| C   |                                                                    | LOOP194 |
|     | DO 280 J=1,NJ                                                      | LOOP195 |
|     | IF(NCHAN(J,1).EQ.0) GO TO 280                                      | LOOP196 |
|     | SUMQ=0.                                                            | LOOP197 |
|     | DO 245 KC=1,KCON                                                   | LOOP198 |
| 245 | DCDT(J,KC)=0.                                                      | LOOP199 |
| C   |                                                                    | LOOP200 |
| C   |                                                                    | LOOP201 |
| C   | ADVECTIVE MOVEMENT IN CHANNELS                                     | LOOP202 |
| C   |                                                                    | LOOP203 |
|     | DO 260 K=1,8                                                       | LOOP204 |
|     | N=NCHAN(J,K)                                                       | LOOP205 |
|     | IF(N.EQ.0) GO TO 260                                               | LOOP206 |
|     | JL=NJUNC(N,1)                                                      | LOOP207 |
|     | JH=NJUNC(N,2)                                                      | LOOP208 |
|     | IF((J.EQ.JL).AND.(Q(N).GE.0.)) GO TO 260                           | LOOP209 |
|     | IF((J.EQ.JH).AND.(Q(N).LE.0.)) GO TO 260                           | LOOP210 |
|     | IF((ABS(L(N)*DELTQ)).GT.LEN(N)) U(N)=SIGN(LEN(N)/DELTQ,U(N))       | LOOP211 |
|     | SUMQ=SUMQ+ABS(Q(N))                                                | LOOP212 |
|     | DO 250 KC=1,KCON                                                   | LOOP213 |
| 250 | DCDT(J,KC)=DCDT(J,KC)+ABS(Q(N))*(U(N)*(C(JL,KC)-C(JH,KC)))/LEN(N)) | LOOP214 |
| 260 | CONTINUE                                                           | LOOP215 |
| 265 | CONTINUE                                                           | LOOP216 |
|     | DO 270 KC=1,KCON                                                   | LOOP217 |
|     | IF (SUMQ.EQ.0.) GO TO 270                                          | LOOP218 |
|     | DCDT(J,KC)=(DCDT(J,KC)/SUMQ)*DELTQ                                 | LOOP219 |
| 270 | CONTINUE                                                           | LOOP220 |
| 280 | CONTINUE                                                           | LOOP221 |
| C   |                                                                    | LOOP222 |
| C   |                                                                    | LOOP223 |
| C   | UPDATE CONCENTRATION AND CHECK                                     | LOOP224 |
| C   | DEPLETION                                                          | LOOP225 |
| C   |                                                                    | LOOP226 |
|     | DO 285 J=1,NJ                                                      | LOOP227 |
|     | IF(NCHAN(J,1).EQ.0) GO TO 285                                      | LOOP228 |
|     | DO 284 KC=1,KCON                                                   | LOOP229 |
|     | C(J,KC)=C(J,KC)+DCDT(J,KC)                                         | LOOP230 |
|     | IF(C(J,KC).LT.0.) C(J,KC)=0.                                       | LOOP231 |
| 284 | CONTINUE                                                           | LOOP232 |
| 285 | CONTINUE                                                           | LOOP233 |
| C   |                                                                    | LOOP234 |
| C   |                                                                    | LOOP235 |
| C   | SOURCE CONTRIBUTION                                                | LOOP236 |
| C   |                                                                    | LOOP237 |
|     | DO 289 J=1,NJ                                                      | LOOP238 |
|     | IF (VCL(J) .LE. 0.0) GO TO 289                                     | LOOP239 |
|     | DO 288 KC=1,KCON                                                   | LOOP240 |
| 288 | C(J,KC) = C(J,KC)+(MADD(J,KC)/VOL(J))*DELTQ                        | LOOP241 |
| 289 | CONTINUE                                                           | LOOP242 |
| C   |                                                                    | LOOP243 |

|   |                                              |         |
|---|----------------------------------------------|---------|
| C |                                              | LJOP244 |
| C | DECAY AND REAERATION                         | LOOP245 |
| C |                                              | LOOP246 |
|   | IF(1SWCH(4).NE.1) GO TO 300                  | LOOP247 |
|   | DO 295 J=1,NJ                                | LOOP248 |
|   | IF(NCHAN(J,1).EQ.0) GO TO 295                | LOOP249 |
|   | DO 290 KC=1,KCONO                            | LOOP250 |
|   | IC=ICON(KC)                                  | LOOP251 |
|   | IF(IC.EQ.0) GO TO 290                        | LOOP252 |
|   | CLOSS=C(J,KC)*DECAY(KC)                      | LOOP253 |
|   | IF(CLOSS.GT.C(J,IC)) CLOSS=C(J,IC)           | LOOP254 |
|   | OGAIN=(CSAT(KC)-C(J,IC))*AS(J)*REAER(KC)     | LOOP255 |
|   | IF(OGAIN.LT.0.) OGAIN=0.                     | LOOP256 |
|   | C(J,KC)=C(J,IC)-CLOSS                        | LOOP257 |
|   | C(J,IC)=C(J,IC)-CLOSS+OGAIN                  | LOOP258 |
|   | IF(C(J,IC).GT.CSAT(KC)) C(J,IC)=CSAT(KC)     | LJOP259 |
|   | IF(C(J,IC).LT.0.) C(J,IC)=0.                 | LOOP260 |
|   | 290 CONTINUE                                 | LOOP261 |
|   | 295 CONTINUE                                 | LJOP262 |
|   | 300 CONTINUE                                 | LOOP263 |
| C |                                              | LOOP264 |
| C |                                              | LOOP265 |
| C | ACCUMULATE MINIMUM, MAXIMUM,                 | LOOP266 |
| C | AND MEAN CONCENTRATIONS                      | LOOP267 |
| C |                                              | LOOP268 |
|   | DO 320 J=1,NJ                                | LOOP269 |
|   | DO 320 KC=1,KCON                             | LOOP270 |
|   | IF(CMIN(J,KC).GT.C(J,KC)) CMIN(J,KC)=C(J,KC) | LOOP271 |
|   | IF(CMAX(J,KC).LT.C(J,KC)) CMAX(J,KC)=C(J,KC) | LOOP272 |
|   | 320 SUMC(J,KC)=SUMC(J,KC)+C(J,KC)            | LOOP273 |
|   | IF (NCPRT.EQ.0) GO TO 4500                   | LOOP274 |
|   | IF (NSTPRT.LT.ITCPRT) GO TO 4500             | LJOP275 |
|   | IF (LQCPRT.LT.NQCTOT) GO TO 4500             | LOOP276 |
|   | CALL CPRINT                                  | LOOP277 |
|   | 4500 CONTINUE                                | LOOP278 |
| C |                                              | LOOP279 |
| C |                                              | LOOP280 |
| C | END QUALITY CYCLE LOOP                       | LOOP281 |
| C |                                              | LOOP282 |
|   | 548 CONTINUE                                 | LOOP293 |
|   | RETURN                                       | LOOP284 |
|   | END                                          | LOOP285 |

|     |                                                                   |            |
|-----|-------------------------------------------------------------------|------------|
|     | SUBROUTINE OUTPUT(NTINT)                                          | OUTPUT 1   |
| C   | OUTPUT SUBROUTINE                                                 | OUTPUT 2   |
| C   | HYDRODYNAMICS PROGRAM                                             | OUTPUT 3   |
| C   | SPECIFICATION STATEMENTS                                          | OUTPUT 4   |
| C   |                                                                   | OUTPUT 5   |
| C   |                                                                   | OUTPUT 6   |
| C   | CONTROL                                                           | OUTPUT 7   |
| C   |                                                                   | OUTPUT 8   |
|     | COMMON /CONTR/ N5,N6,N20,N21, NTCYC,NQCYC,NHCYC, NT,NQSWRT        | OUTPUT 9   |
|     | 1, DELTQ,DELT,TZERO, ISWCH(10)                                    | OUTPUT 10  |
| C   |                                                                   | OUTPUT 11  |
| C   | GENERAL                                                           | OUTPUT 12  |
| C   |                                                                   | OUTPUT 13  |
|     | COMMON ALPHA(30), NJ,NC, ICYC,KCYC,NCYC, WIND,WDIR,EVAP           | OUTPUT 14  |
|     | 1, PRECP(50),NEXIT                                                | OUTPUT 15  |
| C   |                                                                   | OUTPUT 16  |
| C   | JUNCTIONS                                                         | OUTPUT 17  |
| C   |                                                                   | OUTPUT 18  |
|     | COMMON H(100),HN(100),HT(100),HBAR(100),HAVE(100)                 | OUTPUT 19  |
|     | 1, NCHAN(100,8),IPPOINT(100,8),AS(100),VUL(100),X(100),Y(100)     | OUTPUT 20  |
|     | 2, DEP(100),COF(100),CIN(100),QOU(100),QINST(100)                 | OUTPUT 21  |
|     | 3, QINBAR(100),QOUBAR(100)                                        | OUTPUT 22  |
| C   |                                                                   | OUTPUT 23  |
| C   | CHANNELS                                                          | OUTPUT 24  |
| C   |                                                                   | OUTPUT 25  |
|     | COMMON LEN(225),NJUNC(225,2),B(225),R(225),A(225),AT(225),AK(225) | OUTPUT 26  |
|     | 1, Q(225),CBAR(225),QAVE(225), V(225),VT(225),VBAR(225)           | OUTPUT 27  |
|     | 2, FWIND(225),NUMCH(225),NTEMP(8)                                 | OUTPUT 28  |
|     | 3,NCLOS(225)                                                      | OUTPUT 29  |
| C   |                                                                   | OUTPUT 30  |
| C   | PRINTOUT AND PLOTTING                                             | OUTPUT 31  |
| C   |                                                                   | OUTPUT 32  |
|     | COMMON NPRT,IPRT, NHPRT,JPRT(50),PRTH(30,50)                      | OUTPUT 33  |
|     | 1, NQPRT,CPRT(50),PRTV(30,50),PRTQ(30,50), IDUM(12),ICOL(10)      | OUTPUT 34  |
|     | 2, LTIME, NPLT,NPDEL,JPLT(50),HPLT(50)                            | OUTPUT 35  |
|     |                                                                   | OUTPUT 36  |
| C   |                                                                   | OUTPUT 37  |
| C   | STAGE-TIME COEFFICIENTS                                           | OUTPUT 38  |
| C   |                                                                   | OUTPUT 39  |
|     | COMMON YY(50),TT(50),AA(10),XX(10),SXX(10,10),SXY(10)             | OUTPUT 40  |
|     | 1,A1,A2,A3,A4,A5,A6,A7,PERIOD,JGW                                 | OUTPUT 41  |
| C   |                                                                   | OUTPUT 42  |
| C   | STORMWATER                                                        | OUTPUT 43  |
| C   |                                                                   | OUTPUT 44  |
|     | COMMON TITLE(30),NJSW,QE(20,2),JSW(20)                            | OUTPUT 45  |
|     | 2, RAIN(100),INTIME(100),INRAIN,JBOUND(20),JJBOUN                 | OUTPUT 46  |
| C   |                                                                   | OUTPUT 47  |
| C   | TAPES                                                             | OUTPUT 48  |
|     | COMMON /TAPES/ INCNT,IDUTCT,JIN(10),JOUT(10),NSCRAT(5)            | OUTPUT 49  |
| C   |                                                                   | OUTPUT 50  |
|     | COMMON /LAB/ TITL(18),XLAB(11),YLAB(6),HORIZ(20),VERT(7,5),IT     | OUTPUT 51X |
|     | COMMON/PLCT/T(5),NZ(5),AX(101,50),AY(101,50),NPT(50)              | OUTPUT 52  |
|     | IF(NTINT.EC.1) GO TO 210                                          | OUTPUT 53  |
|     | READ(5,1C2) TITL                                                  | OUTPUT 54  |
|     | READ(5,1C2) HORIZ                                                 | OUTPUT 55  |
|     | IT=1                                                              | OUTPUT 55A |
|     | READ(5,1C2) (VERT(11,IT),II=1,7)                                  | OUTPUT 56X |
| 102 | FORMAT(20A4)                                                      | OUTPUT 57  |
|     | RETURN                                                            | OUTPUT 58  |
| 210 | CONTINUE                                                          | OUTPUT 59  |



|                                                      |         |
|------------------------------------------------------|---------|
| DUMMY=0.                                             | OUTP 60 |
| TMAX=10000.                                          | OUTP 61 |
| N22 = JCLT(IOUTCT)                                   | OUTP 62 |
| WRITE(N22) TMAX,(DUMMY,J=1,NPLT)                     | OUTP 63 |
| N=0                                                  | OUTP 64 |
| REWIND N22                                           | OUTP 65 |
| 220 N=N+1                                            | OUTP 66 |
| IF (N .GT. 101) GO TO 225                            | OUTP 67 |
| READ(N22) AX(N,1),(AY(N,J),J=1,NPLT)                 | OUTP 68 |
| IF(AX(N,1).LT.1000.) GO TO 220                       | OUTP 69 |
| 225 CONTINUE                                         | OUTP 70 |
| NCURVE=N-1                                           | OUTP 71 |
| NPTN=0                                               | OUTP 72 |
| N=0                                                  | OUTP 73 |
| 240 N=N+1                                            | OUTP 74 |
| NDC=N                                                | OUTP 75 |
| 250 IF(JPLT(NDC).GE.0) GO TO 270                     | OUTP 76 |
| NDC=NDC+1                                            | OUTP 77 |
| GO TO 250                                            | OUTP 78 |
| 270 CONTINUE                                         | OUTP 79 |
| DO 290 J=N,NDC                                       | OUTP 80 |
| K=J-N+1                                              | OUTP 81 |
| NPT(K)=NCURVE                                        | OUTP 82 |
| DO 290 I=1,NCURVE                                    | OUTP 83 |
| AX(I,K)=AX(I,1)                                      | OUTP 84 |
| AY(I,K)=AY(I,J)                                      | OUTP 85 |
| 290 CONTINUE                                         | OUTP 86 |
| NX=NDC-N+1                                           | OUTP 87 |
| NPTN=NPTN+1                                          | OUTP 88 |
| CALL CURVE(AX,AY,NPT,NX,NPTN)                        | OUTP 89 |
| DO 300 J=N,NDC                                       | OUTP 90 |
| K=J-N+1                                              | OUTP 91 |
| NPT(K)=IABS(JPLT(J))                                 | OUTP 92 |
| 300 CONTINUE                                         | OUTP 93 |
| WRITE(6,50) (NPT(K),K=1,NX)                          | OUTP 94 |
| 50 FORMAT(1H0,4CX,20H PLUT LEGEND ,                  | OUTP 95 |
| 1 15,4H = *,15,4H = +,15,4H = *,15,4H = x,15,4H = .) | OUTP 96 |
| N=NDC                                                | OUTP 97 |
| IF(N.LT.NPLT) GO TO 240                              | OUTP 98 |
| RETURN                                               | OUTP 99 |
| END                                                  | OUTP100 |

|   |                                                                        |         |
|---|------------------------------------------------------------------------|---------|
|   | SUBROUTINE PRTOUT                                                      | PRTD 1  |
| C |                                                                        | PRTD 2  |
| C | PRINTING OUTPUT ROUTINE                                                | PRTD 3  |
| C |                                                                        | PRTD 4  |
| C | HYDRODYNAMICS PROGRAM                                                  | PRTD 5  |
| C | SPECIFICATION STATEMENTS                                               | PRTD 6  |
| C |                                                                        | PRTD 7  |
| C | CONTROL                                                                | PRTD 8  |
| C |                                                                        | PRTD 9  |
|   | COMMON /CONTR/ N5,N6,N20,N21, NTCYC,NQCYC,NHCYC, NT,NQSWRT             | PRTD 10 |
|   | 1, DELTQ,DELT,TZERO, ISWCH(10)                                         | PRTD 11 |
| C |                                                                        | PRTD 12 |
| C | GENERAL                                                                | PRTD 13 |
| C |                                                                        | PRTD 14 |
|   | COMMON ALPHA(30), NJ,NC, ICYC,KCYC,NCYC, WIND,WDIR,EVAP                | PRTD 15 |
|   | 1, PRECP(50),NEXIT                                                     | PRTD 16 |
| C |                                                                        | PRTD 17 |
| C | JUNCTIONS                                                              | PRTD 18 |
| C |                                                                        | PRTD 19 |
|   | COMMON H(100),HN(100),HT(100),HBAR(100),HAVE(100)                      | PRTD 20 |
|   | 1, NCHAN(100,8),IPOINT(100,8),AS(100),VOL(100),X(100),Y(100)           | PRTD 21 |
|   | 2, DEP(100),COF(100),CIN(100),COU(100),QINST(100)                      | PRTD 22 |
|   | 3, QINBAR(100),COUBAR(100)                                             | PRTD 23 |
| C |                                                                        | PRTD 24 |
| C | CHANNELS                                                               | PRTD 25 |
| C |                                                                        | PRTD 26 |
|   | COMMON LEN(225),NJUNC(225,2),B(225),R(225),A(225),AT(225),AK(225)      | PRTD 27 |
|   | 1, Q(225),QBAR(225),QAVE(225), V(225),VT(225),VBAR(225)                | PRTD 28 |
|   | 2, FWIND(225),NUMCH(225),NTEMP(8)                                      | PRTD 29 |
|   | 3,NCLQS(225)                                                           | PRTD 30 |
| C |                                                                        | PRTD 31 |
| C | PRINTOUT AND PLOTTING                                                  | PRTD 32 |
| C |                                                                        | PRTD 33 |
|   | COMMON NPRT,IPRT, NHPRT,JPRT(50),PRTH(30,50)                           | PRTD 34 |
|   | 1, NQPRT,CPRT(50),PRTV(30,50),PRTQ(30,50), IDUM(12),ICOL(10)           | PRTD 35 |
|   | 2, LTIME, NPLT,NPDEL,JPLT(50),HPLT(50)                                 | PRTD 36 |
| C |                                                                        | PRTD 37 |
| C | STAGE-TIME COEFFICIENTS                                                | PRTD 38 |
| C |                                                                        | PRTD 39 |
|   | COMMON YY(50),TT(50),AA(10),XX(10),SXX(10,10),SKY(10)                  | PRTD 40 |
|   | 1,A1,A2,A3,A4,A5,A6,A7,PERIOD,JGW                                      | PRTD 41 |
| C |                                                                        | PRTD 42 |
| C | STORMWATER                                                             | PRTD 43 |
| C |                                                                        | PRTD 44 |
|   | COMMON TITLE(30),NJSW,QE(20,2),JSW(20)                                 | PRTD 45 |
|   | 2, RAIN(100),INTIME(100),INRAIN,JBOUND(20),JJBOUN                      | PRTD 46 |
| C |                                                                        | PRTD 47 |
| C | TAPES                                                                  | PRTD 48 |
| C |                                                                        | PRTD 49 |
|   | COMMON /TAPES/ INCNT,IOUTCT,JIN(10),JOUT(10),NSCRAT(5)                 | PRTD 50 |
| C |                                                                        | PRTD 51 |
| C | TYPE DESIGNATIONS                                                      | PRTD 52 |
| C |                                                                        | PRTD 53 |
|   | INTEGER CPRT                                                           | PRTD 54 |
|   | REAL LEN                                                               | PRTD 55 |
|   | 100 FORMAT(1H115A4,20X,31HWATER RESOURCES ENGINEERS, INC./1H 15A4,20X, | PRTD 56 |
|   | 124HWALNUT CREEK, CALIFORNIA/1H 80X,29HRECEIVING WATER HYDRODYNAMI     | PRTD 57 |
|   | 2CS//)                                                                 | PRTD 58 |
|   | DO 220 I=1,NHPRT,6                                                     | PRTD 59 |
|   |                                                                        | PRTD 60 |



```

280 CONTINUE
    NTINT=1
    CALL OUTPUT(NTINT)
    IF(NJSW.GT.0)
    1WRITE(N6,118) (JSW(L),L=1,NJSW)
118 FORMAT(33H1HYDROGRAPH INPUT NODES TO SYSTEM,/(6X,10I10))
    RETURN
END

```

```

PRT0120
PRT0121
PRT0122
PRT0123
PRT0124
PRT0125
PRT0126
PRT0127

```

|      |                                                                    |         |
|------|--------------------------------------------------------------------|---------|
|      | SUBROUTINE CPRINT                                                  | QPRI 1  |
| C    |                                                                    | QPRI 2  |
| C    |                                                                    | QPRI 3  |
| C    | PRINT ROUTINE FOR QUALITY                                          | QPRI 4  |
| C    | CYCLE OUTPUT.                                                      | QPRI 5  |
| C    |                                                                    | QPRI 6  |
| C    | SPECIFICATION STATEMENTS                                           | QPRI 7  |
| C    |                                                                    | QPRI 8  |
| C    | GENERAL AND CONTROL                                                | QPRI 9  |
| C    |                                                                    | QPRI 10 |
|      | COMMON JGW,NTC,NQCYC,DELTA,QE,QF,ALPHA(30),TITLSW(30),ICOL(10)     | QPRI 11 |
|      | 1,ISWCH(10),REAER(6),DECAY(6),XR(6),XME(6),XMF(6),XMEU(6)          | QPRI 12 |
|      | 2,N5,N6,N10,N20,N30,N40,NSTART,XRQD                                | QPRI 13 |
| C    |                                                                    | QPRI 14 |
| C    | JUNCTIONS                                                          | QPRI 15 |
| C    |                                                                    | QPRI 16 |
|      | COMMON NJ,NCHAN(100,8),QIN(100),QOU(100),VOL(100),VOLO(100)        | QPRI 17 |
|      | 1,AS(100)                                                          | QPRI 18 |
| C    |                                                                    | QPRI 19 |
| C    | CHANNELS                                                           | QPRI 20 |
| C    |                                                                    | QPRI 21 |
|      | COMMON KC,NJUNC(225,2),Q(225),LEN(225),U(225)                      | QPRI 22 |
| C    |                                                                    | QPRI 23 |
| C    | STORMWATER                                                         | QPRI 24 |
| C    |                                                                    | QPRI 25 |
|      | COMMON NJSW,JSW(20)                                                | QPRI 26 |
|      | 1,MJSW,ISW(20),TT(2),CT(6,20,2),INSTM                              | QPRI 27 |
| C    |                                                                    | QPRI 28 |
| C    | QUALITY                                                            | QPRI 29 |
| C    |                                                                    | QPRI 30 |
|      | COMMON KCON,KCOND,C2(6),CS(6),ICON(6),CSAT(6),C(100,6),SJM(100,6)  | QPRI 31 |
|      | 1,CMAX(100,6),CMIN(100,6),MADD(100,6),DCDT(100,6),CE(6,20,2)       | QPRI 32 |
|      | 2,TE(6),TEP(6),SLOPE(20),CSPIN(100,6),TITLE(6,6),TEU(6)            | QPRI 33 |
| C    |                                                                    | QPRI 34 |
| C    | PRINTING                                                           | QPRI 35 |
| C    |                                                                    | QPRI 36 |
|      | COMMON KCPRT,ITCPRT,LQCPRT,NSTPRT,NQCTOT,ISKIP,MSTPRT,NPRT,KPRT    | QPRI 37 |
| C    |                                                                    | QPRI 38 |
| C    |                                                                    | QPRI 39 |
|      | REAL MADD                                                          | QPRI 40 |
|      | IF (ISKIP.NE.NQPRT) GO TO 5020                                     | QPRI 41 |
|      | NQCTOT = NQCTOT + 1                                                | QPRI 42 |
| 4100 | FORMAT(1F015A4,15A4)                                               | QPRI 43 |
|      | WRITE(N6,101) ALPHA                                                | QPRI 44 |
|      | WRITE (N6,4100) TITLSW                                             | QPRI 45 |
| 101  | FORMAT(1H115A4,36X,32H FWQA STORMWATER MODEL /                     | QPRI 46 |
|      | 1 1F 15A4,36X,25H /                                                | QPRI 47 |
|      | 2 1F 96X,28H RECEIVING WATER QUALITY //)                           | QPRI 48 |
|      | WRITE(N6,321) NSTPRT,MSTPRT                                        | QPRI 49 |
| 321  | FORMAT(43H0JUNCTION CONCENTRATIONS, DURING TIME CYCLE,I4,15H ,QUAL | QPRI 50 |
|      | ITY CYCLE,I4//)                                                    | QPRI 51 |
|      | DO 322 KC=1,KCON                                                   | QPRI 52 |
|      | WRITE(N6,325) KC,(TITLE(II,KC),II=1,6),ICOL                        | QPRI 53 |
| 325  | FORMAT(//15X,20H CONSTITUENT NUMBER,I3,6A4/9X,10I10/12H JUNCTI     | QPRI 54 |
|      | ON)                                                                | QPRI 55 |
|      | DO 110 I=1,NJ,10                                                   | QPRI 56 |
|      | L=MINO(I+9,NJ)                                                     | QPRI 57 |
| 110  | WRITE(N6,111) I,L,(C(J,KC),J=1,L)                                  | QPRI 58 |
| 111  | FORMAT (I4,4H TO ,I3,1X,10F10.2)                                   | QPRI 59 |
| 322  | CONTINUE                                                           | QPRI 60 |

```
      ISKIP = 1
      GO TO 5010
5020 CONTINUE
      ISKIP = ISKIP + 1
5010 CONTINUE
      RETURN
      END
```

```
QPRI 61
QPRI 62
QPRI 63
QPRI 64
QPRI 65
QPRI 66
QPRI 67
```

|                                                              |         |
|--------------------------------------------------------------|---------|
| SUBROUTINE RECEIV                                            | RECE 1  |
| COMMON /TAPES/ INCNT, IOUTCT, JIN(10), JOUT(10), NSCRAT(5)   | RECE 2  |
| DIMENSION QUAN(2), QUAL(2), ANAME(4)                         | RECE 3  |
| DATA QUAN/4HQUAN, 4HTITY/                                    | RECE 4  |
| DATA QUAL/4HQUAL, 4HITY /                                    | RECE 5  |
| N5=5                                                         | RECE 6  |
| N6=6                                                         | RECE 7  |
| INCNT=INCNT+1                                                | RECE 8  |
| READ (N5,100) (ANAME(I), I=1,4)                              | RECE 9  |
| 100 FORMAT(4A4,I4)                                           | RECE 10 |
| IF (ANAME(1).EQ.QUAN(1).AND.ANAME(2).EQ.QUAN(2)) GO TO 200   | RECE 11 |
| 150 IF(ANAME(3).EQ.QUAL(1).AND.ANAME(4).EQ.QUAL(2))GO TO 300 | RECE 12 |
| GO TO 400                                                    | RECE 13 |
| 200 CALL SHFLOW                                              | RECE 14 |
| GO TO 150                                                    | RECE 15 |
| 300 CALL SHQUAL                                              | RECE 16 |
| 400 WRITE(N6,500)                                            | RECE 17 |
| 500 FORMAT (31HORECEIVING SIMULATION COMPLETED)              | RECE 18 |
| RETURN                                                       | RECE 19 |
| END                                                          | RECE 20 |

|   |                                                                   |         |
|---|-------------------------------------------------------------------|---------|
|   | SUBROUTINE SWFLGW                                                 | SWFL 1  |
| C |                                                                   | SWFL 2  |
| C | HYDRODYNAMICS PROGRAM                                             | SWFL 3  |
| C | TIDAL OPTION                                                      | SWFL 4  |
| C |                                                                   | SWFL 5  |
| C | SPECIFICATION STATEMENTS                                          | SWFL 6  |
| C |                                                                   | SWFL 7  |
| C | CONTROL                                                           | SWFL 8  |
| C |                                                                   | SWFL 9  |
|   | COMMON /CONTR/ N5,N6,N20,N21, NTCYC,NQCYC,NHCYC, NT,NQSWRT        | SWFL 10 |
|   | 1, DELTC,DELTA,TZERO, ISWCH(10)                                   | SWFL 11 |
| C |                                                                   | SWFL 12 |
| C | GENERAL                                                           | SWFL 13 |
| C |                                                                   | SWFL 14 |
|   | COMMON ALPHA(3C), NJ,NC, ICYC,KCYC,NCYC, WIND,WDIR,EVAP           | SWFL 15 |
|   | 1, PRECP(50),NEXIT                                                | SWFL 16 |
| C |                                                                   | SWFL 17 |
| C | JUNCTIONS                                                         | SWFL 18 |
| C |                                                                   | SWFL 19 |
|   | COMMON H(100),FN(100),HT(100),HBAR(100),HAVE(100)                 | SWFL 20 |
|   | 1, NCHAN(100,8),IPUINT(100,8),AS(100),VOL(100),X(100),Y(100)      | SWFL 21 |
|   | 2, DEP(100),COF(100),QIN(100),QOU(100),QINST(100)                 | SWFL 22 |
|   | 3, QINBAR(100),COUBAR(100)                                        | SWFL 23 |
| C |                                                                   | SWFL 24 |
| C | CHANNELS                                                          | SWFL 25 |
| C |                                                                   | SWFL 26 |
|   | COMMON LEN(225),NJUNC(225,2),B(225),R(225),A(225),AT(225),AK(225) | SWFL 27 |
|   | 1, Q(225),QBAR(225),QAVE(225), V(225),VT(225),VBAR(225)           | SWFL 28 |
|   | 2, FWIND(225),NUMCH(225),NTEMP(8)                                 | SWFL 29 |
|   | 3,NCLDS(225)                                                      | SWFL 30 |
| C |                                                                   | SWFL 31 |
| C | PRINTOUT AND PLOTTING                                             | SWFL 32 |
| C |                                                                   | SWFL 33 |
|   | COMMON APRT,IPRT, NHPRT,JPRT(50),PRTH(30,50)                      | SWFL 34 |
|   | 1, NQPRT,CPRT(50),PRTV(30,50),PRTQ(30,50), IDUM(12),ICOL(10)      | SWFL 35 |
|   | 2, LTIME, NPLT,NPDEL,JPLT(50),HPLT(50)                            | SWFL 36 |
| C |                                                                   | SWFL 37 |
| C | STAGE-TIME COEFFICIENTS                                           | SWFL 38 |
| C |                                                                   | SWFL 39 |
|   | COMMON YY(50),TT(50),AA(10),XX(10),SXX(10,10),SXY(10)             | SWFL 40 |
|   | 1,A1,A2,A3,A4,A5,A6,A7,PERIOD,JGW                                 | SWFL 41 |
| C |                                                                   | SWFL 42 |
| C | STORMWATER                                                        | SWFL 43 |
| C |                                                                   | SWFL 44 |
|   | COMMON TITLE(30),NJSW,QE(20,2),JSW(20)                            | SWFL 45 |
|   | 2, RAIN(100),INTIME(100),INRAIN,JBOUND(20),JJBOUN                 | SWFL 46 |
| C |                                                                   | SWFL 47 |
| C | TAPES                                                             | SWFL 48 |
| C |                                                                   | SWFL 49 |
|   | COMMON /TAPES/ INCNT, IOUTCT,JIN(10),JOUT(10),NSCRAT(5)           | SWFL 50 |
|   | COMMON/ST2/ TITEL2(40)                                            | SWFL 51 |
| C |                                                                   | SWFL 52 |
|   | DIMENSION ENDER(2)                                                | SWFL 53 |
|   | COMMON QT(20,2),ISW(20)                                           | SWFL 54 |
|   | DATA ENDER / 4HENDQ, 4HUANT /                                     | SWFL 55 |
| C |                                                                   | SWFL 56 |
| C | TYPE DESIGNATIONS                                                 | SWFL 57 |
|   | INTEGER CPRT                                                      | SWFL 58 |
|   | REAL LEN,INTIME                                                   | SWFL 59 |
| C |                                                                   | SWFL 60 |



|     |                                   |                             |         |
|-----|-----------------------------------|-----------------------------|---------|
| C   |                                   | INITIALIZATION              | SWFL 61 |
| C   | N20 = NSCRAT(1)                   |                             | SWFL 62 |
|     | NEXIT=0                           |                             | SWFL 63 |
|     | DO 205 I=1,100                    |                             | SWFL 64 |
|     | DEP(I)=0.0                        |                             | SWFL 65 |
|     | AS(I)=0.                          |                             | SWFL 66 |
|     | QIN(I)=0.                         |                             | SWFL 67 |
|     | QUU(I)=0.                         |                             | SWFL 68 |
|     | DO 204 J=1,8                      |                             | SWFL 69 |
|     | IPOINT(I,J)=0                     |                             | SWFL 70 |
|     | NCHAN(I,J)=0                      |                             | SWFL 71 |
| 204 | CONTINUE                          |                             | SWFL 72 |
| 205 | CONTINUE                          |                             | SWFL 73 |
|     | DO 210 I=1,225                    |                             | SWFL 74 |
|     | FWIND(I) = 0.0                    |                             | SWFL 75 |
|     | DO 209 J=1,2                      |                             | SWFL 76 |
|     | NJUNC(I,J)=0                      |                             | SWFL 77 |
| 209 | CONTINUE                          |                             | SWFL 78 |
| 210 | CONTINUE                          |                             | SWFL 79 |
| C   |                                   |                             | SWFL 80 |
| C   |                                   | SUBROUTINE INDATA CALLED TO | SWFL 81 |
| C   |                                   | READ INPUT DATA             | SWFL 82 |
| C   |                                   |                             | SWFL 83 |
| C   | CALL INDATA                       |                             | SWFL 84 |
| C   |                                   |                             | SWFL 85 |
| C   |                                   | FURTHER INITIALIZATION      | SWFL 86 |
| C   |                                   |                             | SWFL 87 |
|     | IOUTCT = ICUTCT + 1               |                             | SWFL 88 |
|     | N21=JIN(INCNT)                    |                             | SWFL 89 |
|     | N22 = JCUT(IOUTCT)                |                             | SWFL 90 |
|     | REWIND N22                        |                             | SWFL 91 |
|     | NTINT = 0                         |                             | SWFL 92 |
|     | TT(1) = C.0                       |                             | SWFL 93 |
|     | TT(2) = C.0                       |                             | SWFL 94 |
|     | NSTEPS = 0                        |                             | SWFL 95 |
|     | MJSW = 0                          |                             | SWFL 96 |
|     | NQUAL = C                         |                             | SWFL 97 |
|     | TDELT = C                         |                             | SWFL 98 |
|     | WEIR1 = A1                        |                             | SWFL 99 |
|     | WEIR2 = A2                        |                             |         |
|     | WEIR3 = A3                        |                             |         |
|     | IF (NPLT.GT.C) CALL OUTPUT(NTINT) |                             | SWFL100 |
|     | DO 220 I=1,10                     |                             | SWFL101 |
| 220 | ICOL(I)=I                         |                             | SWFL102 |
|     | DO 222 I=1,20                     |                             | SWFL103 |
|     | ISW(I) = 0                        |                             | SWFL104 |
|     | QT(I,1) = 0.0                     |                             | SWFL105 |
|     | QT(I,2) = 0.0                     |                             | SWFL106 |
|     | QE(I,1) = 0.0                     |                             | SWFL107 |
| 222 | QE(I,2) = 0.0                     |                             | SWFL108 |
|     | TE=0.                             |                             | SWFL109 |
|     | TEP=0.                            |                             | SWFL110 |
|     | DELT2=DELT/2.0                    |                             | SWFL111 |
|     | TZERG=TZER0*3600.0                |                             | SWFL112 |
|     | W=6.2832/(3600.*PERIOD)           |                             | SWFL113 |
|     | EVAP=EVAP/(12.*30.*86400.)        |                             | SWFL114 |
|     | TOLD=0.                           |                             | SWFL115 |
|     | KRAIN=1                           |                             | SWFL116 |
|     | PREC = 0.0                        |                             | SWFL117 |

|                                                          |          |
|----------------------------------------------------------|----------|
| T=TZERO                                                  | SWFL118  |
| 223 DO 224 I = 1,NHPRT                                   | SWFL118A |
| HJPRT = JPRT(I)                                          | SWFL118B |
| PRTH (1,I) = H(HJPRT)                                    | SWFL118C |
| 224 CONTINUE                                             | SWFL118D |
| DO 225 I = 1,NCPRT                                       | SWFL118E |
| MCPRT = CPRT(I)                                          | SWFL118F |
| PRTQ(1,I) = Q(MCPRT)                                     | SWFL118G |
| PRTV(1,I) = V(MCPRT)                                     | SWFL118H |
| 225 CONTINUE                                             | SWFL118I |
| C                                                        | SWFL119  |
| C                                                        | SWFL120  |
| C                                                        | SWFL121  |
| C                                                        | SWFL122  |
| IF(N21.EQ.0) GO TO 230                                   | SWFL123  |
| REWIND N21                                               | SWFL124  |
| READ (N21)           TITEL2                              | SWFL125  |
| WRITE (N6,7097)    TITEL2                                | SWFL126  |
| 7097 FORMAT (20A4)                                       | SWFL127  |
| READ (N21)           NSTEPS,MJSW,NQUAL,TDELT,TZERO,TAREA | SWFL128X |
| WRITE (N6,7091) NSTEPS,MJSW,NQUAL,TDELT,TZERO,TAREA      | SWFL129X |
| 7091 FORMAT (3I10,3F10.2)                                | SWFL130X |
| READ(N21) (ISW(L),L=1,MJSW)                              | SWFL131  |
| WRITE(6,7092) (ISW(L),L=1,MJSW)                          | SWFL133  |
| 7092 FORMAT(5I15)                                        | SWFL134  |
| READ(N21) TT(1),(QT(L,1),L=1,MJSW)                       | SWFL135  |
| WRITE(6,7093) TT(1),(QT(L,1),L=1,MJSW)                   | SWFL137  |
| READ(N21) TT(2),(QT(L,2),L=1,MJSW)                       | SWFL138  |
| WRITE(6,7093) TT(2),(QT(L,2),L=1,MJSW)                   | SWFL140  |
| NINREC=2                                                 | SWFL141  |
| I1=1                                                     | SWFL142  |
| I2=2                                                     | SWFL143  |
| TTP=0.                                                   | SWFL144  |
| 230 IF(NJSW.EQ.0) GO TO 235                              | SWFL145  |
| REAC(N5,100) (JSW(L),L=1,NJSW)                           | SWFL146  |
| READ(N5,104) TE,(QE(L,2),L=1,NJSW)                       | SWFL147  |
| WRITE(N6,102) (JSW(L),L=1,NJSW)                          | SWFL148  |
| TEP=TE/3600.                                             | SWFL149  |
| WRITE(N6,106) TEP,(QE(L,2),L=1,NJSW)                     | SWFL150  |
| 100 FORMAT (16I5)                                        | SWFL151  |
| 102 FORMAT('1HYDROGRAPH INPUTS TO SYSTEM'//(6X,10I10))   | SWFL152  |
| 235 CONTINUE                                             | SWFL153  |
| TIME=TZERO                                               | SWFL154  |
| C                                                        | SWFL155  |
| C                                                        | SWFL156  |
| C                                                        | SWFL157  |
| C                                                        | SWFL158  |
| IF (ISWCH(1).NE.1) GO TO 236                             | SWFL158A |
| H(JGW)=A1+A2*SIN(W*T)+A3*SIN(2.*W*T)+A4*SIN(3.*W*T)      | SWFL159  |
| 1       +A5*COS(W*T)+A6*COS(2.*W*T)+A7*COS(3.*W*T)       | SWFL160  |
| 236 CONTINUE                                             | SWFL160A |
| C                                                        | SWFL161  |
| C                                                        | SWFL162  |
| C                                                        | SWFL163  |
| CHANNEL CONSTANTS COMPUTED                               | SWFL164  |
| 260 DO 280 N=1,NC                                        | SWFL165  |
| IF(NJUNC(N,1).LE.0)GO TO 280                             | SWFL166  |
| AK(N)=32.1739*AK(N)**2/2.208196                          | SWFL167  |
| NL=NJUNC(N,1)                                            | SWFL168  |
| NH=NJUNC(N,2)                                            | SWFL169  |
| R(N)=R(N)+(H(NL)*H(NH))/2.                               |          |

|                                                                    |         |
|--------------------------------------------------------------------|---------|
| A(N)=B(N)*R(N)                                                     | SWFL170 |
| IF (WIND.LE.0.0) GO TO 270                                         | SWFL171 |
| FWIND(N)=-WIND**2*COS(WDIR/57.-ATAN2((X(NH)-X(NL)),(Y(NH)-Y(NL)))) | SWFL172 |
| 1 *8.64E-6                                                         | SWFL173 |
| 270 CONTINUE                                                       | SWFL174 |
| AT(N)=A(N)                                                         | SWFL175 |
| QAVE(N)=C.                                                         | SWFL176 |
| 280 CONTINUE                                                       | SWFL177 |
| C                                                                  | SWFL178 |
| C                                                                  | SWFL179 |
| C                                                                  | SWFL180 |
| DO 340 J=1,NJ                                                      | SWFL181 |
| VOL(J)=0.                                                          | SWFL182 |
| IF(AS(J).EQ.0.) GO TO 340                                          | SWFL183 |
| AREA=0.                                                            | SWFL184 |
| VOLUME=0.                                                          | SWFL185 |
| DO 300 K=1,8                                                       | SWFL186 |
| N=NCHAN(J,K)                                                       | SWFL187 |
| IF(N.LE.0) GO TO 300                                               | SWFL188 |
| AREA=AREA+B(N)*LEN(N)                                              | SWFL189 |
| VOLUME=VOLUME+B(N)*LEN(N)*R(N)                                     | SWFL190 |
| 300 CONTINUE                                                       | SWFL191 |
| 320 DEPTH=VOLUME/AREA                                              | SWFL192 |
| VOL(J)=DEPTH*AS(J)                                                 | SWFL193 |
| 340 CONTINUE                                                       | SWFL194 |
| C                                                                  | SWFL195 |
| C                                                                  | SWFL196 |
| C                                                                  | SWFL197 |
| C                                                                  | SWFL198 |
| C                                                                  | SWFL199 |
| C                                                                  | SWFL200 |
| C                                                                  | SWFL201 |
| DO 1300 NT=1,NTCYC                                                 | SWFL202 |
| IF(NT.LT.NQSWRT) GO TO 350                                         | SWFL203 |
| REWIND A22                                                         | SWFL204 |
| DO 345 J=1,NPLT                                                    | SWFL205 |
| I=IABS(JPLT(J))                                                    | SWFL206 |
| 345 HPLT(J)=H(I)                                                   | SWFL207 |
| HOUR=TZERO/3600.                                                   | SWFL208 |
| WRITE(A22) HCUR,(HPLT(J),J=1,NPLT)                                 | SWFL209 |
| NPTOT=1                                                            | SWFL210 |
| 350 CONTINUE                                                       | SWFL211 |
| LTIME = 1                                                          | SWFL212 |
| C                                                                  | SWFL213 |
| C                                                                  | SWFL214 |
| C                                                                  | SWFL215 |
| DO 1240 NQ=1,NQCYC                                                 | SWFL216 |
| C                                                                  | SWFL217 |
| C                                                                  | SWFL218 |
| C                                                                  | SWFL219 |
| C                                                                  | SWFL220 |
| C                                                                  | SWFL221 |
| IF (NT.LT.NQSWRT) GO TO 380                                        | SWFL222 |
| DO 360 N=1,NC                                                      | SWFL223 |
| VBAR(N)=0.                                                         | SWFL224 |
| 360 QBAR(N)=0.                                                     | SWFL225 |
| DO 370 J=1,NJ                                                      | SWFL226 |
| HBAR(J)=0.                                                         | SWFL227 |
| QINBAR(J)=0.                                                       | SWFL228 |
| QOUBAR(J)=0.                                                       | SWFL229 |

|      |                                                            |         |
|------|------------------------------------------------------------|---------|
| 370  | CONTINUE                                                   | SWFL230 |
| 380  | CONTINUE                                                   | SWFL231 |
| C    |                                                            | SWFL232 |
| C    |                                                            | SWFL233 |
| C    | START OF HYDRAULIC DO LOOP,                                | SWFL234 |
| C    | INNERMOST DO LOOP OF 3 NESTED                              | SWFL235 |
| C    | DO LOOPS                                                   | SWFL236 |
|      | DO 1040 NHH=1,NHCYC                                        | SWFL237 |
|      | IF(NT.LE.NQSWRT) GO TO 520                                 | SWFL238 |
|      | TIME=TIME+DELT                                             | SWFL239 |
| C    |                                                            | SWFL240 |
| C    |                                                            | SWFL241 |
| C    | PRECIPITATION COMPUTATIONS FOR                             | SWFL242 |
| C    | EACH TIME STEP                                             | SWFL243 |
|      | PREC=0.                                                    | SWFL244 |
| 390  | IF(KRAIN-INRAIN) 395,410,410                               | SWFL245 |
| 395  | IF(TIME-INTIME(KRAIN+1)) 405,400,400                       | SWFL246 |
| 400  | PREC=PREC+RAIN(KRAIN)*((INTIME(KRAIN+1)-TOLD)/(12.*3600.)) | SWFL247 |
|      | KRAIN=KRAIN+1                                              | SWFL248 |
|      | TOLD=INTIME(KRAIN)                                         | SWFL249 |
|      | GO TO 390                                                  | SWFL250 |
| 405  | PREC=(PREC+RAIN(KRAIN)*((TIME-TOLD)/(12.*3600.)))/DELT     | SWFL251 |
|      | TOLD=TIME                                                  | SWFL252 |
| 410  | CONTINUE                                                   | SWFL253 |
|      | IF(N21.EQ.0) GO TO 445                                     | SWFL254 |
|      | DO 418 L=1,MJSW                                            | SWFL255 |
|      | J=ISW(L)                                                   | SWFL256 |
| 418  | QIN(J)=QINST(J)*DELT                                       | SWFL257 |
| 420  | IF(TIME-TT(I2)) 435,425,425                                | SWFL258 |
| 425  | DO 430 L=1,MJSW                                            | SWFL259 |
|      | J=ISW(L)                                                   | SWFL260 |
| 430  | QIN(J)=QIN(J)+QT(L,I1)*((TT(I2)-TTP)                       | SWFL261 |
|      | TTP=TT(I2)                                                 | SWFL262 |
|      | ITEMP=I2                                                   | SWFL263 |
|      | I2=I1                                                      | SWFL264 |
|      | I1=ITEMP                                                   | SWFL265 |
|      | IF(NINREC-NSTEPS) 431,432,432                              | SWFL266 |
| 431  | READ(N21) TT(I2),(QT(L,I2),L=1,MJSW)                       | SWFL267 |
|      | NINREC=NINREC+1                                            | SWFL269 |
| 7093 | FORMAT(F10.1,6F15.2)                                       | SWFL270 |
|      | WRITE(6,7094) TT(I2),(QT(L,I2),L=1,MJSW) ,NINREC           | SWFL271 |
| 7094 | FORMAT(F10.1,F10.1,110)                                    | SWFL272 |
|      | GO TO 420                                                  | SWFL273 |
| 432  | TT(I2)=1000000.                                            | SWFL274 |
|      | DO 433 L=1,MJSW                                            | SWFL275 |
| 433  | QT(L,I2)=0.                                                | SWFL276 |
|      | GO TO 420                                                  | SWFL277 |
| 435  | DO 440 L=1,MJSW                                            | SWFL278 |
|      | J=ISW(L)                                                   | SWFL279 |
| 440  | QIN(J)=(QIN(J)+QT(L,I1)*(TIME-TTP))/DELT                   | SWFL280 |
|      | TTP=TIME                                                   | SWFL281 |
| 445  | CONTINUE                                                   | SWFL282 |
|      | IF(NJSW.EQ.0) GO TO 520                                    | SWFL283 |
| C    |                                                            | SWFL284 |
| C    |                                                            | SWFL285 |
| C    | READ HYDROGRAPH INPUT OR AVERAGES                          | SWFL286 |
| C    | OR INTERPOLATE FOR TIME STEP                               | SWFL287 |
|      | IF(TIME.LE.TE) GO TO 480                                   | SWFL288 |
|      | TEU=TE                                                     | SWFL289 |
|      | DO 460 L=1,NJSW                                            | SWFL290 |

|     |                                                             |         |
|-----|-------------------------------------------------------------|---------|
| 460 | QE(L,1)=QE(L,2)                                             | SWFL291 |
| C   |                                                             | SWFL292 |
| C   | READ HYDROGRAPHS                                            | SWFL293 |
| C   |                                                             | SWFL294 |
|     | READ (N5,104) TE,(QE(JJ,2),JJ=1,NJSW)                       | SWFL295 |
| 104 | FORMAT (8F10.0)                                             | SWFL296 |
| 470 | CONTINUE                                                    | SWFL297 |
|     | TEP=TE/3600.                                                | SWFL298 |
|     | WRITE(N6,106) TEP,(QE(L,2),L=1,NJSW)                        | SWFL299 |
| 106 | FORMAT(' F7.2,10F10.1/(8X,10F10.1))                         | SWFL300 |
| C   |                                                             | SWFL301 |
| C   | INTERPOLATE HYDROGRAPH                                      | SWFL302 |
| C   |                                                             | SWFL303 |
| 480 | DO 500 L=1,NJSW                                             | SWFL304 |
|     | J=JSW(L)                                                    | SWFL305 |
|     | SLOPE=(QE(L,2)-QE(L,1))/(TE-TEO)                            | SWFL306 |
| 500 | QIN(J)=CINST(J)+QE(L,1)+SLOPE*(TIME-TEO)                    | SWFL307 |
| 520 | CONTINUE                                                    | SWFL308 |
| C   |                                                             | SWFL309 |
| C   | INITIALIZATION                                              | SWFL310 |
| C   |                                                             | SWFL311 |
|     | T2=T+DELT2                                                  | SWFL312 |
|     | T=T+DELT                                                    | SWFL313 |
|     | DO 525 N=1,NC                                               | SWFL314 |
|     | NCLOS(N)=0                                                  | SWFL315 |
|     | DO 525 M=1,2                                                | SWFL316 |
| 525 | NJUNC(N,M)=IABS(NJUNC(N,M))                                 | SWFL317 |
|     | DO 530 J=1,NJ                                               | SWFL318 |
|     | AS(J)=ABS(AS(J))                                            | SWFL319 |
|     | DO 530 K=1,8                                                | SWFL320 |
| 530 | NCHAN(J,K)=IABS(NCHAN(J,K))                                 | SWFL321 |
|     | NTIMS=0                                                     | SWFL322 |
| C   |                                                             | SWFL323 |
| C   | COMPUTATIONS OF VELOCITIES AT                               | SWFL324 |
| C   | HALF TIME STEP, AND FLOWS AT                                | SWFL325 |
| C   | QUARTER TIME STEP                                           | SWFL326 |
| C   |                                                             | SWFL327 |
| 540 | CONTINUE                                                    | SWFL328 |
|     | NDRY=0                                                      | SWFL329 |
|     | NTIMS=NTIMS+1                                               | SWFL330 |
|     | DO 580 N=1,NC                                               | SWFL331 |
|     | IF(NJUNC(N,1).LE.0)GO TO 580                                | SWFL332 |
| C   |                                                             | SWFL333 |
| C   | DRY CHANNEL CHECK (UNDER 0.1 FT)                            | SWFL334 |
| C   |                                                             | SWFL335 |
|     | IF(R(N).GT.0.1) GO TO 560                                   | SWFL336 |
|     | VT(N)=0.0                                                   | SWFL337 |
|     | Q(N)=0.0                                                    | SWFL338 |
|     | GO TO 580                                                   | SWFL339 |
| 560 | CONTINUE                                                    | SWFL340 |
|     | NL=NJUNC(N,1)                                               | SWFL341 |
|     | NH=NJUNC(N,2)                                               | SWFL342 |
|     | DELV2=V(N)*(1.-AT(N)/A(N))                                  | SWFL343 |
|     | 1 +DELT2*((V(N)**2*B(N)/A(N))-32.1739)*(H(NH)-H(NL))/LEV(N) | SWFL344 |
|     | 2+FWIND(N)/R(N)*DELT2                                       | SWFL345 |
|     | V2=V(N)+DELV2                                               | SWFL346 |
|     | TEMP=DELT2*AK(N)/R(N)**1.3333333                            | SWFL347 |
|     | DELV1=0.5*((1./TEMP+2.*ABS(V2))-                            | SWFL348 |
|     | 1SQRT((1./TEMP+ABS(2.*V2))**2-4.*V2**2))                    | SWFL349 |
|     | DELV1=-SIGN(DELV1,V2)                                       | SWFL350 |

|     |                                                               |          |
|-----|---------------------------------------------------------------|----------|
|     | VT(N)=V(N)+DELV1+DELV2                                        | SWFL351  |
|     | Q(N)=VT(N)*A(N)                                               | SWFL352  |
| 580 | CONTINUE                                                      | SWFL353  |
| C   |                                                               | SWFL354  |
| C   |                                                               | SWFL355  |
| C   | COMPUTATION OF NODAL STAGE AT                                 | SWFL356  |
| C   | HALF TIME STEP                                                | SWFL357  |
|     | DO 660 J=1,NJ                                                 | SWFL358  |
|     | SUMQ=0.                                                       | SWFL359  |
|     | DO 620 K=1,8                                                  | SWFL360  |
|     | IF(NCHAN(J,K).LE.0) GO TO 620                                 | SWFL361  |
|     | N=NCHAN(J,K)                                                  | SWFL362  |
|     | IF(J.NE.NJUNC(N,1))GO TO 600                                  | SWFL363  |
|     | SUMQ=SUMQ+Q(N)                                                | SWFL364  |
|     | GO TO 620                                                     | SWFL365  |
| 600 | SUMQ=SUMQ-Q(N)                                                | SWFL366  |
| 620 | CONTINUE                                                      | SWFL367  |
|     |                                                               | SWFL368  |
| 640 | IF(AS(J).LE.0.) GO TO 660                                     | SWFL369  |
|     | SUMQ=QOU(J)-QIN(J)+(EVAP-PREC)*AS(J)+SUMQ                     | SWFL370  |
|     | IF (J.EQ.JGW.AND.ISWCH(1).NE.1) GO TO 650                     | SWFL370T |
|     | HT(J)=H(J)-DELT2*SUMQ/AS(J)                                   | SWFL371  |
|     | IF(HT(J)+DEP(J).GT.0.) GO TO 660                              | SWFL372  |
|     | HT(J)=-DEP(J)                                                 | SWFL373  |
|     | VOL(J)=0.                                                     | SWFL374  |
|     | AS(J)=-AS(J)                                                  | SWFL375  |
|     | DO 645 K=1,8                                                  | SWFL376  |
|     | NX=NCHAN(J,K)                                                 | SWFL377  |
|     | IF(NX.LE.0) GO TO 645                                         | SWFL378  |
|     | NCLOS(NX)=1                                                   | SWFL379  |
| 645 | CONTINUE                                                      | SWFL380  |
|     | NDRY=NDRY+1                                                   | SWFL381  |
|     | GO TO 660                                                     |          |
| 650 | CONTINUE                                                      |          |
|     | DELHH=0.                                                      |          |
|     | DO 655 ICT=1,3                                                |          |
|     | DELHH=DELT2/AS(J)*(-SUMQ-WEIR1*(H(JGW)-WEIR2+DELHH/2.)*WEIR3) |          |
| 655 | CONTINUE                                                      |          |
|     | HT(J)=H(J)+DELHH                                              |          |
| 660 | CONTINUE                                                      | SWFL382  |
|     | IF(NDRY.EQ.0) GO TO 675                                       | SWFL383  |
|     | IF(NTI*S.GT.2) GO TO 675                                      | SWFL384  |
|     | DO 670 N=1,NC                                                 | SWFL385  |
|     | IF(NJUNC(N,1).LE.0) GO TO 670                                 | SWFL386  |
|     | IF(NCLOS(N).NE.1) GO TO 670                                   | SWFL387  |
|     | Q(N)=0.                                                       | SWFL388  |
|     | V(N)=0.                                                       | SWFL389  |
|     | DO 668 I=1,2                                                  | SWFL390  |
|     | II=NJUNC(N,I)                                                 | SWFL391  |
|     | DO 664 J=1,8                                                  | SWFL392  |
|     | IF(NCHAN(II,J).EQ.N) GO TO 666                                | SWFL393  |
| 664 | CONTINUE                                                      | SWFL394  |
|     | GO TO 668                                                     | SWFL395  |
| 666 | NCHAN(II,J)=-N                                                | SWFL396  |
| 668 | NJUNC(N,I)=-II                                                | SWFL397T |
| 670 | CONTINUE                                                      | SWFL398  |
|     | GO TO 540                                                     | SWFL399  |
| 675 | CONTINUE                                                      | SWFL400  |
| C   |                                                               | SWFL401  |
| C   | BOUNDARY STAGE CONDITION AT                                   | SWFL402  |

|     |                                                                   |                                  |          |
|-----|-------------------------------------------------------------------|----------------------------------|----------|
| C   |                                                                   | HALF TIME STEP                   | SWFL403  |
| C   |                                                                   |                                  | SWFL404  |
|     | IF (ISWCH(1).NE.1) GO TO 676                                      |                                  | SWFL404T |
|     | HT(JGW)=A1+A2*SIN(W*T2)+A3*SIN(2.*W*T2)+A4*SIN(3.*W*T2)           |                                  | SWFL405  |
|     | 1 +A5*COS(W*T2)+A6*COS(2.*W*T2)+A7*COS(3.*W*T2)                   |                                  | SWFL406  |
|     | 676 CONTINUE                                                      |                                  | SWFL406T |
| C   |                                                                   |                                  | SWFL407  |
| C   |                                                                   | COMPUTATION OF CHANNEL CROSS-    | SWFL408  |
| C   |                                                                   | SECTIONAL AREAS AT HALF TIME     | SWFL409  |
| C   |                                                                   | STEP, FLOWS AT HALF TIME STEP,   | SWFL410  |
| C   |                                                                   | AND VELOCITIES AT FULL TIME STEP | SWFL411  |
| C   |                                                                   |                                  | SWFL412  |
|     | DO 740 N=1,NC                                                     |                                  | SWFL413  |
|     | IF (NJUNC(N,1).LE.0) GO TO 740                                    |                                  | SWFL414  |
|     | NL=NJUNC(N,1)                                                     |                                  | SWFL415  |
|     | NH=NJUNC(N,2)                                                     |                                  | SWFL416  |
|     | DELH=0.5*(HT(NH)-H(NH)+HT(NL)-H(NL))                              |                                  | SWFL417  |
|     | RNT=R(N)+DELH                                                     |                                  | SWFL418  |
|     | AT(N)=A(N)+B(N)*DELH                                              |                                  | SWFL419  |
| C   |                                                                   |                                  | SWFL420  |
| C   |                                                                   | DRY CHANNEL CHECK (UNDER 0.1 FT) | SWFL421  |
| C   |                                                                   |                                  | SWFL422  |
|     | IF(RNT.GT.0.1) GO TO 680                                          |                                  | SWFL423  |
|     | V(N)=0.                                                           |                                  | SWFL424  |
|     | Q(N)=0.                                                           |                                  | SWFL425  |
|     | GO TO 700                                                         |                                  | SWFL426  |
| 680 | CONTINUE                                                          |                                  | SWFL427  |
|     | DELV2=2.*VT(N)*(1.-A(N)/AT(N))                                    |                                  | SWFL428  |
|     | 1 +DELH*((VT(N)**2*B(N)/AT(N))-32.1739)*(HT(NH)-HT(NL))/LEN(N)    |                                  | SWFL429  |
|     | 1+FWIND(N)/RNT*DELH                                               |                                  | SWFL430  |
|     | V2= V(N)+DELV2                                                    |                                  | SWFL431  |
|     | TEMP=DELH*AK(N)/RNT**1.3333333                                    |                                  | SWFL432  |
|     | DELV1=0.5*((1./TEMP+2.*ABS(V2))-                                  |                                  | SWFL433  |
|     | 1SQRT((1./TEMP+2.*ABS(V2))**2-4.*V2**2))                          |                                  | SWFL434  |
|     | DELV1=-SIGN(DELV1,V2)                                             |                                  | SWFL435  |
|     | V(N)=V(N)+DELV1+DELV2                                             |                                  | SWFL436  |
|     | Q(N)=0.5*(C(N)+V(N)*AT(N))                                        |                                  | SWFL437  |
| 700 | CONTINUE                                                          |                                  | SWFL438  |
| C   |                                                                   |                                  | SWFL439  |
| C   |                                                                   | CHANNEL FLOWS SUMMED             | SWFL440  |
| C   |                                                                   |                                  | SWFL441  |
|     | IF (NT.LT.NCSWRT) GO TO 720                                       |                                  | SWFL442  |
|     | QBAR(N)=QBAR(N)+Q(N)                                              |                                  | SWFL443  |
|     | VBAR(N)=VBAR(N)+V(N)                                              |                                  | SWFL444  |
| 720 | CONTINUE                                                          |                                  | SWFL445  |
| C   |                                                                   |                                  | SWFL446  |
| C   |                                                                   | EXCESSIVE VELOCITY CHECK         | SWFL447  |
| C   |                                                                   |                                  | SWFL448  |
|     | IF(ABS(V(N)).LE.20.0) GO TO 740                                   |                                  | SWFL449  |
|     | WRITE(6,1C8) NT,NC,NHH,R(N),V(N),N                                |                                  | SWFL450  |
| 108 | FORMAT('O V OVER 20 FPS, TIDAL CYCLE',I4,' QUAL CYCLE',I4,' HYDRO |                                  | SWFL451  |
|     | 1CYCLE',I4,' DEPTH',E10.4,' V',E10.4,' CHANNEL',I5)               |                                  | SWFL452  |
|     | NEXIT=1                                                           |                                  | SWFL453  |
| 740 | CONTINUE                                                          |                                  | SWFL454  |
|     | IF (NEXIT.EQ.1) GO TO 1260                                        |                                  | SWFL455  |
| C   |                                                                   |                                  | SWFL456  |
| C   |                                                                   | COMPUTATION OF NODAL STAGE AND   | SWFL457  |
| C   |                                                                   | VOLUME AT FULL TIME STEP         | SWFL458  |
| C   |                                                                   |                                  | SWFL459  |
|     | 760 DO 900 J=1,NJ                                                 |                                  | SWFL460  |

|                                                              |                                 |
|--------------------------------------------------------------|---------------------------------|
| SUMQ=0.                                                      | SWFL461                         |
| HN(J)=-DEP(J)                                                | SWFL462                         |
| IF(AS(J).LE.0.) GO TO 900                                    | SWFL463                         |
| DO 800 K=1,8                                                 | SWFL464                         |
| IF(NCHAN(J,K).LE.0) GO TO 800                                | SWFL465                         |
| N=NCHAN(J,K)                                                 | SWFL466                         |
| IF(J.NE.NJUNC(N,1))GO TO 780                                 | SWFL467                         |
| SUMQ=SUMQ+C(N)                                               | SWFL468                         |
| GO TO 800                                                    | SWFL469                         |
| 780 SUMQ=SUMQ-C(N)                                           | SWFL470                         |
| 800 CONTINUE                                                 | SWFL471                         |
| IF(J.NE.JGW) GO TO 820                                       | SWFL472                         |
| IF (ISWCH(1).NE.1) GO TO 802                                 | SWFL472T                        |
| HN(JGW)=A1+A2*SIN(W*T)+A3*SIN(2.*W*T)+A4*SIN(3.*W*T)         | SWFL473                         |
| 1 +A5*CGS(W*T)+A6*CGS(2.*W*T)+A7*CGS(3.*W*T)                 | SWFL474                         |
| GO TO 814                                                    | SWFL474T                        |
| 802 CONTINUE                                                 | SWFL474A                        |
| DELHH=DELHH*2.                                               | SWFL474B                        |
| DO 808 ICT=1,3                                               | SWFL474C                        |
| DELHH=DEL/AS(J)*(-SUMQ-WEIR1*(H(JGW)-WEIR2+DELHH/2.))*WEIR3) | SWFL474D                        |
| 808 CONTINUE                                                 | SWFL474E                        |
| HN(J)=F(J)+DELHH                                             | SWFL474F                        |
| 814 CONTINUE                                                 | SWFL474G                        |
| DVOL=(HN(JGW)-H(JGW))*AS(JGW)                                | 2WVCM75                         |
| QOU(JGW)=0.                                                  | SWFL476                         |
| QIN(JGW)=(DVOL/DELT)+SUMQ                                    | SWFL477                         |
| IF (QIN(JGW).GT.0.) GO TO 815                                | SWFL478                         |
| QOU(JGW)=-QIN(JGW)                                           | SWFL479                         |
| QIN(JGW)=0.                                                  | SWFL480                         |
| 815 VOL(JGW)=VOL(JGW)+DVOL                                   | SWFL481                         |
| GO TO 825                                                    | SWFL482                         |
| C                                                            | SWFL483                         |
| C                                                            | COMPUTATION OF 'ORDINARY' NODES |
| C                                                            | VOLUME AND STAGE                |
| C                                                            | SWFL484                         |
| 820 SUMQ=COU(J)-QIN(J)+(EVAP-PREC)*AS(J)+SUMQ                | SWFL485                         |
| HN(J)=F(J)-DELT*SUMQ/AS(J)                                   | SWFL486                         |
| VOL(J)=VCL(J)-DELT*SUMQ                                      | SWFL487                         |
| C                                                            | SWFL488                         |
| 825 CONTINUE                                                 | SWFL489                         |
| 900 CONTINUE                                                 | SWFL490                         |
| C                                                            | SWFL491                         |
| C                                                            | SWFL492                         |
| C                                                            | SWFL493                         |
| C                                                            | NODAL VOLUMES AND FLOWS SUMMED  |
| IF (NT.LT.NQSWRT) GO TO 940                                  | SWFL494                         |
| DO 920 J=1,NJ                                                | SWFL495                         |
| HBAR(J)=HBAR(J)+HN(J)                                        | SWFL496                         |
| QINBAR(J)=QINBAR(J)+QIN(J)                                   | SWFL497                         |
| QUUBAR(J)=QCUBAR(J)+QOU(J)                                   | SWFL498                         |
| 920 CONTINUE                                                 | SWFL499                         |
| 940 CONTINUE                                                 | SWFL500                         |
| C                                                            | SWFL501                         |
| C                                                            | SWFL502                         |
| C                                                            | SWFL503                         |
| C                                                            | FULL TIME STEP COMPUTATION OF   |
| C                                                            | HYDRAULIC RADIUS AND CHANNEL    |
| C                                                            | CROSS-SECTIONAL AREAS           |
| C                                                            | SWFL504                         |
| 960 DO 980 K=1,NC                                            | SWFL505                         |
| IF(NJUNC(N,1).EQ.0) GO TO 980                                | SWFL506                         |
| NL=IABS(NJUNC(N,1))                                          | SWFL507                         |
| NH=IABS(NJUNC(N,2))                                          | SWFL508                         |
|                                                              | SWFL509                         |
|                                                              | SWFL510                         |
|                                                              | SWFL511                         |



|                                                      |         |
|------------------------------------------------------|---------|
| DELH=0.5*(HN(NH)-H(NH)+HN(NL)-H(NL))                 | SWFL512 |
| R(N)=R(N)+DELH                                       | SWFL513 |
| A(N)=A(N)+B(N)*DELH                                  | SWFL514 |
| 980 CONTINUE                                         | SWFL515 |
| C                                                    | SWFL516 |
| C                                                    | SWFL517 |
| C                                                    | SWFL518 |
| 1000 DO 1020 J=1,NJ                                  | SWFL519 |
| 1020 H(J)=HN(J)                                      | SWFL520 |
| IF(NT.LT.NQSWRT) GO TO 1040                          | SWFL521 |
| IF(NPTOT.NE.NPDEL) GO TO 1030                        | SWFL522 |
| DO 1025 J=1,NPLT                                     | SWFL523 |
| I=IA3S(JPLT(J))                                      | SWFL524 |
| 1025 HPLT(J)=H(I)                                    | SWFL525 |
| HOUR=HCUR+DELT/3600.*NPDEL                           | SWFL526 |
| WRITE(N22) HOUR,(HPLT(J),J=1,NPLT)                   | SWFL527 |
| NPTOT=0                                              | SWFL528 |
| 1030 NPTOT=NPTOT+1                                   | SWFL529 |
| C                                                    | SWFL530 |
| C                                                    | SWFL531 |
| C                                                    | SWFL532 |
| C                                                    | SWFL533 |
| 1040 CONTINUE                                        | SWFL534 |
| C                                                    | SWFL535 |
| C                                                    | SWFL536 |
| C                                                    | SWFL537 |
| C                                                    | SWFL538 |
| IF(NT.LT.NQSWRT) GO TO 1100                          | SWFL539 |
| DO 1060 N=1,NC                                       | SWFL540 |
| IF(NJUNC(N,1).LE.0) GO TO 1060                       | SWFL541 |
| QBAR(N)=QBAR(N)/FLOAT(NHCYC)                         | SWFL542 |
| VBAR(N)=VBAR(N)/FLOAT(NHCYC)                         | SWFL543 |
| QAVE(N)=QAVE(N)+QBAR(N)/FLOAT(NQCYC)                 | SWFL544 |
| 1060 CONTINUE                                        | SWFL545 |
| DO 1080 J=1,NJ                                       | SWFL546 |
| QINBAR(J)=QINBAR(J)/FLOAT(NHCYC)                     | SWFL547 |
| QOUBAR(J)=QOUBAR(J)/FLOAT(NHCYC)                     | SWFL548 |
| HBAR(J)=HBAR(J)/FLOAT(NHCYC)                         | SWFL549 |
| IF(QINBAR(J).EQ.0.) GOTO 1080                        | SWFL550 |
| IF(QOUBAR(J).EQ.0.) GOTO 1080                        | SWFL551 |
| QINBAR(J)=QINBAR(J)-QOUBAR(J)                        | SWFL552 |
| QOUBAR(J)=0.                                         | SWFL553 |
| IF(QINBAR(J).GT.0.) GO TO 1080                       | SWFL554 |
| QOUBAR(J)=-QINBAR(J)                                 | SWFL555 |
| QINBAR(J)=0.                                         | SWFL556 |
| 1080 CONTINUE                                        | SWFL557 |
| C                                                    | SWFL558 |
| C                                                    | SWFL559 |
| C                                                    | SWFL560 |
| C                                                    | SWFL561 |
| WRITE(N20) NQ,(QBAR(N),VBAR(N),N=1,NC),              | SWFL562 |
| 1 (VOL(J),QINBAR(J),QOUBAR(J),J=1,NJ)                | SWFL563 |
| C                                                    | SWFL564 |
| C                                                    | SWFL565 |
| C                                                    | SWFL566 |
| C                                                    | SWFL567 |
| 1100 IF(NT.EQ.(NQSWRT-1).AND.NQ.EQ.NQCYC) GO TO 1120 | SWFL568 |
| GO TO 1180                                           | SWFL569 |
| 1120 DO 1140 I = 1,NHPRT                             | SWFL570 |
| MJPRT = JPRT(I)                                      | SWFL571 |

|                                                                               |          |
|-------------------------------------------------------------------------------|----------|
| PRTH (1,1) = H(MJPRT)                                                         | SWFL572  |
| 1140 CONTINUE                                                                 | SWFL573  |
| DO 1160 I = 1,NQPR                                                            | SWFL574  |
| MCPRT = CPRT(I)                                                               | SWFL575  |
| PRTQ(1,I) = Q(MCPRT)                                                          | SWFL576  |
| PRTV(1,I) = V(MCPRT)                                                          | SWFL577  |
| 1160 CONTINUE                                                                 | SWFL578  |
| GO TO 1240                                                                    | SWFL579  |
| 1180 IF(NT.LT.NQSWRT) GO TO 1240                                              | SWFL580  |
| LTIME = LTIME + 1                                                             | SWFL581  |
| C                                                                             | SWFL582  |
| C                                                                             | SWFL583  |
| C                                                                             | SWFL584  |
| DO 1200 I=1,NMPRT                                                             | SWFL585  |
| MJPRT=JPRT(I)                                                                 | SWFL586  |
| 1200 PRTH(LTIME,I)=H(MJPRT)                                                   | SWFL587  |
| C                                                                             | SWFL588  |
| C                                                                             | SWFL589  |
| C                                                                             | SWFL590  |
| DO 1220 I=1,NCPRT                                                             | SWFL591  |
| MCPRT=CPRT(I)                                                                 | SWFL592  |
| PRTQ(LTIME,I)=Q(MCPRT)                                                        | SWFL593  |
| 1220 PRTV(LTIME,I)=V(MCPRT)                                                   | SWFL594  |
| C                                                                             | SWFL595  |
| C                                                                             | SWFL596  |
| C                                                                             | SWFL597  |
| 1240 CONTINUE                                                                 | SWFL598  |
| IF(ISWCH(1).NE.1) GO TO 1280                                                  | SWFL599  |
| IF (NT.NE.NQSWRT) GO TO 1280                                                  | SWFL600  |
| 1260 CONTINUE                                                                 | SWFL601  |
| 1280 CONTINUE                                                                 | SWFL602  |
| C                                                                             | SWFL603  |
| C                                                                             | SWFL604  |
| C                                                                             | SWFL605  |
| C                                                                             | SWFL606  |
| C                                                                             | SWFL607  |
| IF (NT.LT.NQSWRT) GO TO 1300                                                  | SWFL608  |
| CALL PRTCUT                                                                   | SWFL609  |
| C                                                                             | SWFL610  |
| C                                                                             | SWFL611  |
| C                                                                             | SWFL612  |
| 1300 CONTINUE                                                                 | SWFL613  |
| END FILE N20                                                                  | SWFL614  |
| REWIND N20                                                                    | SWFL615  |
| 1320 CONTINUE                                                                 | SWFL616  |
| MCCOUNT = 0                                                                   | SWFL617  |
| 1340 READ (N5,110) FINAL,CARD                                                 | SWFL618  |
| 110 FORMAT (2A4)                                                              | SWFL619  |
| IF (FINAL.EQ.ENDER(1)) GO TO 1360                                             | SWFL620  |
| MCCOUNT = MCCOUNT + 1                                                         | SWFL621  |
| IF (MCCOUNT.GT.30) GO TO 1380                                                 | SWFL622  |
| GO TO 1340                                                                    | SWFL623  |
| 1360 IF (CARD.EQ.ENDER(2)) GO TO 1400                                         | SWFL624  |
| MCCOUNT = MCCOUNT + 1                                                         | SWFL625  |
| IF (MCCOUNT.GT.30) GO TO 1380                                                 | SWFL626  |
| GO TO 1340                                                                    | SWFL627  |
| 1380 WRITE (N6,112)                                                           | SWFL628  |
| 112 FORMAT (62HOQUALITY PROGRAM HAS READ MORE THAN 30 CARDS AFTER COMPLETION) | SWFL629  |
| 1PLETICA)                                                                     | SWFL630  |
| STOP 4444                                                                     | SWFL631X |

|      |                                               |         |
|------|-----------------------------------------------|---------|
| 1400 | WRITE (N6,114)                                | SWFL632 |
| 114  | FORMAT (33H0COMPLETION OF RECEIVING QUANTITY) | SWFL633 |
| 1420 | CONTINUE                                      | SWFL634 |
| C    |                                               | SWFL635 |
| C    | RETURN TO SUBROUTINE RECEIV                   | SWFL636 |
| C    |                                               | SWFL637 |
|      | RETURN                                        | SWFL638 |
|      | END                                           | SWFL639 |

|   |                                                                    |         |
|---|--------------------------------------------------------------------|---------|
|   | SUBROUTINE SWQUAL                                                  | SWQU 1  |
| C |                                                                    | SWQU 2  |
| C | RECEIVING WATER QUALITY                                            | SWQU 3  |
| C |                                                                    | SWQU 4  |
| C |                                                                    | SWQU 5  |
| C | SPECIFICATION STATEMENTS                                           | SWQU 6  |
| C |                                                                    | SWQU 7  |
| C | GENERAL AND CONTROL                                                | SWQU 8  |
| C |                                                                    | SWQU 9  |
|   | COMMON JGW,NTC,NQCYC,DELTQ,QE,QF,ALPHA(30),TITLSW(30),ICJL(10)     | SWQU 10 |
|   | 1 ,ISWCH(10),REAER(6),DECAY(6),XR(6),XME(6),XMF(6),XMED(6)         | SWQU 11 |
|   | 2 ,N5,N6,N10,N20,N30,N40,NSTART,XRQD                               | SWQU 12 |
| C |                                                                    | SWQU 13 |
| C | JUNCTIONS                                                          | SWQU 14 |
| C |                                                                    | SWQU 15 |
|   | COMMON NJ,NCHAN(100,8),QIN(100),QOU(100),VOL(100),VOLO(100)        | SWQU 16 |
|   | 1 ,AS(100)                                                         | SWQU 17 |
| C |                                                                    | SWQU 18 |
| C | CHANNELS                                                           | SWQU 19 |
| C |                                                                    | SWQU 20 |
|   | COMMON NC,NJUNC(225,2),Q(225),LEN(225),U(225)                      | SWQU 21 |
| C |                                                                    | SWQU 22 |
| C | STORMWATER                                                         | SWQU 23 |
| C |                                                                    | SWQU 24 |
|   | COMMON NJSW,JSW(20)                                                | SWQU 25 |
|   | 1,MJSW,ISW(20),TT(2),CT(6,20,2),INSTH                              | SWQU 26 |
| C |                                                                    | SWQU 27 |
| C | QUALITY                                                            | SWQU 28 |
| C |                                                                    | SWQU 29 |
|   | COMMON KCON,KCCNO,C2(6),CS(6),ICON(6),CSAT(6),C(100,6),SUMC(100,6) | SWQU 30 |
|   | 1 ,CMAX(100,6),CMIN(100,6),MADD(100,6),DCDT(100,6),CE(6,20,2)      | SWQU 31 |
|   | 2 ,TE( 6),TEP( 6),SLOPE(20),CSPIN(100,6),TITLE(6,6),TED(6)         | SWQU 32 |
| C |                                                                    | SWQU 33 |
| C | PRINTING                                                           | SWQU 34 |
| C |                                                                    | SWQU 35 |
|   | COMMON NCPRT,ITCPRT,LQCPRT,NSTPRT,NQCTOT,ISKIP,MSTPRT,NPRT,KPRT    | SWQU 36 |
| C |                                                                    | SWQU 37 |
| C |                                                                    | SWQU 38 |
| C | TAPES                                                              | SWQU 39 |
| C |                                                                    | SWQU 40 |
|   | COMMON /TAPES/ INCNT,IOUTC,TJIN(10),JOUT(10),NSCRAT(5)             | SWQU 41 |
| C |                                                                    | SWQU 42 |
| C | START PROGRAM                                                      | SWQU 43 |
| C | INPUT ROUTINE                                                      | SWQU 44 |
| C |                                                                    | SWQU 45 |
|   | REAL MADD,LEN                                                      | SWQU 46 |
|   | N20 = NSCRAT(1)                                                    | SWQU 47 |
|   | INSTH=C                                                            | SWQU 48 |
| C |                                                                    | SWQU 49 |
| C | CALL TO SUBROUTINE INQJAL                                          | SWQU 50 |
| C |                                                                    | SWQU 51 |
|   | CALL INCLAL                                                        | SWQU 52 |
| C |                                                                    | SWQU 53 |
| C | MAIN QUALITY LOOP                                                  | SWQU 54 |
| C |                                                                    | SWQU 55 |
|   | DO 751 NTAG=NSTART,NTC                                             | SWQU 56 |
|   | REWIND NIO                                                         | SWQU 57 |
|   | NSTPRT = NTAG                                                      | SWQU 58 |
| C |                                                                    | SWQU 59 |
| C |                                                                    | SWQU 60 |

|      |                                                                           |          |
|------|---------------------------------------------------------------------------|----------|
| C    | CALL TO QUALITY CYCLE SUBROUTINES                                         | SWQU 61  |
| C    |                                                                           | SWQJ 62  |
|      | CALL LCCPCL                                                               | SWQU 63  |
|      | INSTM=INSTM+1                                                             | SWQU 64  |
| C    |                                                                           | SWQU 65  |
| C    | PRINT DAY AVERAGE CONCENTRATIONS                                          | SWQU 66  |
| C    |                                                                           | SWQU 67  |
|      | DO 359 J = 1,NJ                                                           | SWQU 68  |
|      | DO 359 KC=1,KCON                                                          | SWQJ 69  |
| 359  | SUMC(J,KC)=(SUMC(J,KC)-0.5*C(J,KC))/NQCYC                                 | SWQU 70  |
|      | IF (NTAG.LT.KPRT) GO TO 3220                                              | SWQU 71  |
|      | IF (NTAG.GT.KPRT.AND.KPRT.EQ.1) GO TO 1600                                | SWQU 72  |
|      | GO TO 4300                                                                | SWQU 73  |
| 1600 | WRITE(N6,2030)                                                            | SWQU 74  |
| 3030 | FORMAT(42H KPRT IS ALWAYS GOING TO BE LESS THAN NTAG)                     | SWQU 75  |
|      | STOP 5555                                                                 | SWQU 76X |
| 4300 | CONTINUE                                                                  | SWQJ 77  |
|      | KPRT=KPRT+NPRT                                                            | SWQU 78  |
| 752  | CONTINUE                                                                  | SWQJ 79  |
|      | DO 322 KC=1,KCCN                                                          | SWQU 80  |
|      | WRITE (N6,4100) TITLSW                                                    | SWQU 81  |
| 4100 | FORMAT(1H15A4,15A4)                                                       | SWQJ 82  |
|      | WRITE(6,101) ALPHA                                                        | SWQU 83  |
| 101  | FORMAT(1H015A4,36X,32H WATER RESOURCES ENGINEERS, INC./                   | SWQJ 84  |
| 1    | 1F 15A4,36X,25H WALNUT CREEK, CALIFORNIA/                                 | SWQU 85  |
| 2    | 1H 96X,28H DYNAMIC STORM WATER QUALITY//)                                 | SWQJ 86  |
|      | WRITE (6,321) NTAG,KC,(TITLE(11,KC),11=1,6),ICOL                          | SWQU 87  |
| 321  | FORMAT('0'10X,'AVERAGE JUNCTION CONCENTRATIONS DURING TIDAL OR TIMSWQJ 88 |          |
|      | 1E CYCLE'14,', CCNSTITUTENT NUMBER'13,5X,6A4//9X,10110/' JUNCTIONSWQJ 89  |          |
|      | 2')                                                                       | SWQU 90  |
|      | DO 110 I=1,NJ,10                                                          | SWQU 91  |
|      | L=MINO(I+9,NJ)                                                            | SWQJ 92  |
| 110  | WRITE (6,111) I,L,(SUMC(J,KC),J=I,L)                                      | SWQU 93  |
| 111  | FORMAT(14,5H TC ,13,1X,10E10.4)                                           | SWQU 94  |
|      | IF(1SWCF(2).EQ.1) GO TO 322                                               | SWQJ 95  |
|      | WRITE (6,112)                                                             | SWQU 96  |
| 112  | FORMAT('0'50X,'MAXIMUMS/' JUNCTION')                                      | SWQU 97  |
|      | DO 113 I=1,NJ,10                                                          | SWQU 98  |
|      | L=MINO(I+9,NJ)                                                            | SWQU 99  |
| 113  | WRITE (6,111) I,L,(CMAX(J,KC),J=I,L)                                      | SWQU100  |
|      | WRITE (6,114)                                                             | SWQU101  |
| 114  | FORMAT('0'50X,'MINIMUMS/' JUNCTION')                                      | SWQU102  |
|      | DO 115 I=1,NJ,10                                                          | SWQU103  |
|      | L=MINO(I+9,NJ)                                                            | SWQU104  |
| 115  | WRITE (6,111) I,L,(CMIN(J,KC),J=I,L)                                      | SWQU105  |
| 322  | CONTINUE                                                                  | SWQU106  |
| C    |                                                                           | SWQJ107  |
| C    | RESET SUMS FOR NEXT DAY CYCLE                                             | SWQU108  |
| C    |                                                                           | SWQJ109  |
| 3220 | CONTINUE                                                                  | SWQU110  |
|      | DO 323 J=1,NJ                                                             | SWQU111  |
|      | VOL0(J)=VOL(J)                                                            | SWQJ112  |
|      | DO 323 KC=1,KCON                                                          | SWQU113  |
|      | CMAX(J,KC)=0.                                                             | SWQU114  |
|      | CMIN(J,KC)=C(J,KC)                                                        | SWQU115  |
| 323  | SUMC(J,KC)=0.5*C(J,KC)                                                    | SWQU116  |
|      | IF (NTAG.EC.NTC) GO TO 4140                                               | SWQJ117  |
|      | REWIND N10                                                                | SWQJ118  |
| C    |                                                                           | SWQU119  |
| C    | ISWCH 10 SET BY N30 READ-IN                                               | SWQJ120  |

|      |                                                                 |         |
|------|-----------------------------------------------------------------|---------|
| C    | IF (ISWCH(10).EQ.1) GO TO 4140                                  | SWQU121 |
|      | DO 4120 I=1,NQCYC                                               | SWQU122 |
|      | READ(N20) NQ,(Q(N),U(N),N=1,NC),(VOL(J),QIN(J),QOU(J),J=1,NJ)   | SWQU123 |
|      | WRITE(N10)NQ,(Q(N),U(N),N=1,NC),(VOL(J),QIN(J),QOU(J),J=1,NJ)   | SWQU124 |
| 4120 | CONTINUE                                                        | SWQU125 |
|      | REWIND N10                                                      | SWQU126 |
| 4140 | CONTINUE                                                        | SWQU127 |
| C    |                                                                 | SWQU128 |
| C    |                                                                 | SWQU129 |
| C    | END OF MAIN DO-LOOP                                             | SWQU130 |
| C    |                                                                 | SWQU131 |
|      | 751 CONTINUE                                                    | SWQU132 |
|      | IF (ISWCH(3).EQ.1) GO TO 4160                                   | SWQU133 |
|      | RETURN                                                          | SWQU134 |
| C    |                                                                 | SWQU135 |
| C    | WRITE A RESTART TAPE                                            | SWQU136 |
| C    |                                                                 | SWQU137 |
|      | 4160 CONTINUE                                                   | SWQU138 |
|      | WRITE (N40) JGW,KCON,KCCNC,NTC,NPRT,NJSW,TITLE,C2,CS,ICON,CSAT, | SWQU139 |
|      | 1 REAER,DECAY,XR,XME,XMF,XMEO,(VOLO(J),(C(J,K),SUMC(J,K),       | SWQU140 |
|      | 2 MADD(J,K),CMAX(J,K),CMIN(J,K),K=1,KCON),J=1,NJ)               | SWQU141 |
|      | DO 4465 I=1,NQCYC                                               | SWQU142 |
|      | REWIND N10                                                      | SWQU143 |
|      | READ(N10) NQ,(Q(N),U(N),N=1,NC),(VOL(J),QIN(J),QOU(J),J=1,NJ)   | SWQU144 |
|      | WRITE(N40)NQ,(Q(N),U(N),N=1,NC),(VOL(J),QIN(J),QOU(J),J=1,NJ)   | SWQU145 |
| 4465 | CONTINUE                                                        | SWQU146 |
|      | END FILE N40                                                    | SWQU147 |
|      | REWIND N40                                                      | SWQU148 |
|      | RETURN                                                          | SWQU149 |
|      | END                                                             | SWQU150 |
|      |                                                                 | SWQU151 |

|   |                                                                   |         |
|---|-------------------------------------------------------------------|---------|
| C | SUBROUTINE TIDCF(KO,NI,MAXIT,NCHTID)                              | TIDC 1  |
| C |                                                                   | TIDC 2  |
| C | STAGE TIME COEFFICIENTS                                           | TIDC 3  |
| C | HYDRODYNAMICS PROGRAM                                             | TIDC 4  |
| C | SPECIFICATION STATEMENTS                                          | TIDC 5  |
| C |                                                                   | TIDC 6  |
| C | CONTROL                                                           | TIDC 7  |
| C |                                                                   | TIDC 8  |
|   | COMMON /CCNTR/ N5,N6,N20,N21, NTCYC,NQCYC,NHCYC, NT,NQSWRT        | TIDC 9  |
|   | 1, DELTQ,DELT,TZERO, ISWCH(10)                                    | TIDC 10 |
| C |                                                                   | TIDC 11 |
| C | GENERAL                                                           | TIDC 12 |
| C |                                                                   | TIDC 13 |
|   | COMMON ALPHA(30), NJ,NC, ICYC,KCYC,NCYC, WIND,WDIR,EVAP           | TIDC 14 |
|   | 1, PRECP(50),NEXIT                                                | TIDC 15 |
| C |                                                                   | TIDC 16 |
| C | JUNCTIONS                                                         | TIDC 17 |
| C |                                                                   | TIDC 18 |
|   | COMMON F(100),HN(100),HT(100),HBAR(100),HAVE(100)                 | TIDC 19 |
|   | 1, NCHAN(100,8),IPDINT(100,8),AS(100),VOL(100),X(100),Y(100)      | TIDC 20 |
|   | 2, DEP(100),COF(100),QIN(100),QOU(100),QINST(100)                 | TIDC 21 |
|   | 3, QINBAR(100),QOLBAR(100)                                        | TIDC 22 |
| C |                                                                   | TIDC 23 |
| C | CHANNELS                                                          | TIDC 24 |
| C |                                                                   | TIDC 25 |
|   | COMMON LEN(225),NJUNC(225,2),B(225),R(225),A(225),AT(225),AK(225) | TIDC 26 |
|   | 1, Q(225),QBAR(225),CAVE(225), V(225),VT(225),VBAR(225)           | TIDC 27 |
|   | 2, FWINC(225),NUMCH(225),NTEMP(8)                                 | TIDC 28 |
|   | 3,NCLOS(225)                                                      | TIDC 29 |
| C |                                                                   | TIDC 30 |
| C | PRINTOUT AND PLOTTING                                             | TIDC 31 |
| C |                                                                   | TIDC 32 |
|   | COMMON NPRT,IPRT, NHPRT,JPRT(50),PRTH(30,50)                      | TIDC 33 |
|   | 1, NQPRT,CPRT(50),PRTV(30,50),PRTQ(30,50), IDUM(12),ICOL(10)      | TIDC 34 |
|   | 2, LTIME, NPLT,NPDEL,JPLT(50),HPLT(50)                            | TIDC 35 |
| C |                                                                   | TIDC 36 |
| C | STAGE-TIME COEFFICIENTS                                           | TIDC 37 |
| C |                                                                   | TIDC 38 |
|   | COMMON YY(50) ,TT(50) ,AA(10),XX(10),SXX(10,10),SXY(10)           | TIDC 39 |
|   | 1,A1,A2,A3,A4,A5,A6,A7,PERIOD,JGW                                 | TIDC 40 |
| C |                                                                   | TIDC 41 |
| C | STORMWATER                                                        | TIDC 42 |
| C |                                                                   | TIDC 43 |
|   | COMMON TITLE(30),NJSW,QE(20,2),JSW(20)                            | TIDC 44 |
|   | 2, RAIN(100),INTIME(100),INRAIN,JBOUND(20),JJBOUN                 | TIDC 45 |
| C |                                                                   | TIDC 46 |
| C | TAPES                                                             | TIDC 47 |
| C |                                                                   | TIDC 48 |
|   | COMMON /TAPES/ INCNT,IOUTCT,JIN(10),JOUT(10),NSCRAT(5)            | TIDC 49 |
| C |                                                                   | TIDC 50 |
| C |                                                                   | TIDC 51 |
| C | TIDE COEFFICIENTS                                                 | TIDC 52 |
| C |                                                                   | TIDC 53 |
| C |                                                                   | TIDC 54 |
| C |                                                                   | TIDC 55 |
| C | TYPE DESIGNATION                                                  | TIDC 56 |
| C |                                                                   | TIDC 57 |
|   | INTEGER CPRT                                                      | TIDC 58 |
|   | REAL LEN                                                          | TIDC 59 |
| C |                                                                   | TIDC 60 |

|   |                                                                       |         |
|---|-----------------------------------------------------------------------|---------|
| C | TIDAL CURVE FIT, 7 TERM                                               | TIDC 61 |
| C | SINUSOIDAL EQUATION                                                   | TIDC 62 |
| C |                                                                       | TIDC 63 |
|   | WRITE (N6,140) KO,NI,MAXIT,NCHTID                                     | TIDC 64 |
|   | 140 FORMAT (7H0 KO IS,I3,19H NUMBER OF TERMS IS,I4,32H MAXIMUM NUMBER | TIDC 65 |
|   | 1 OF ITERATIONS IS,I4,21H TIDE CHECK SWITCH IS,I2)                    | TIDC 66 |
| C | IF KO EQUALS ONE, PROGRAM WILL                                        | TIDC 67 |
| C | READ FOUR POINTS OF INFORMATION                                       | TIDC 68 |
| C | AND EXPAND THEM FOR A FULL TIDE                                       | TIDC 69 |
| C |                                                                       | TIDC 70 |
| C | NT IS THE NUMBER OF INFORMATION                                       | TIDC 71 |
| C | POINTS                                                                | TIDC 72 |
| C | MAXIT IS THE MAXIMUM NUMBER OF                                        | TIDC 73 |
| C | ITERATIONS                                                            | TIDC 74 |
| C | IF NCHTID EQUALS ONE, TIDAL                                           | TIDC 75 |
| C | INPUT-OUTPUT WILL BE PRINTED                                          | TIDC 76 |
| C |                                                                       | TIDC 77 |
| C | DELTA IS THE ACCURACY                                                 | TIDC 78 |
| C | LIMIT IN FEET                                                         | TIDC 79 |
| C |                                                                       | TIDC 80 |
|   | DELTA = 0.005                                                         | TIDC 81 |
|   | NTT=7                                                                 | TIDC 82 |
|   | W = 2.*3.14159 /PERIOD                                                | TIDC 83 |
|   | IF(KO.EQ.0) GO TO 225                                                 | TIDC 84 |
|   | TT(50) =TT(1)+PERIOD                                                  | TIDC 85 |
|   | YY(50)=YY(1)                                                          | TIDC 86 |
|   | DO 220 I=1,4                                                          | TIDC 87 |
|   | J=I+1                                                                 | TIDC 88 |
|   | IF (J.GT.4) J=50                                                      | TIDC 89 |
|   | NI=NI+1                                                               | TIDC 90 |
|   | TT(NI)=(3.*TT(I)+TT(J))/4.                                            | TIDC 91 |
|   | YY(NI)=0.8535*YY(I)+0.1465*YY(J)                                      | TIDC 92 |
|   | NI=NI+1                                                               | TIDC 93 |
|   | TT(NI)=(TT(I)+TT(J))/2.                                               | TIDC 94 |
|   | YY(NI)=(YY(I)+YY(J))/2.                                               | TIDC 95 |
|   | NI=NI+1                                                               | TIDC 96 |
|   | TT(NI)=(TT(I)+3.*TT(J))/4.                                            | TIDC 97 |
|   | YY(NI)=0.1465*YY(I)+0.8535*YY(J)                                      | TIDC 98 |
|   | 220 CONTINUE                                                          | TIDC 99 |
|   | 225 CONTINUE                                                          | TIDC100 |
|   | IF (NCHTID.NE.1) GO TO 240                                            | TIDC101 |
|   | WRITE (N6,146)                                                        | TIDC102 |
|   | 146 FORMAT (29H0 NO.            TIME            VALUE )               | TIDC103 |
|   | WRITE (N6,148) (I,TT(I), YY(I), I=1,NI)                               | TIDC104 |
|   | 148 FORMAT (I4, 2F12.3 )                                              | TIDC105 |
|   | 240 CONTINUE                                                          | TIDC106 |
|   | DO 280 J=1,NTT                                                        | TIDC107 |
|   | DO 260 K=1,NTT                                                        | TIDC108 |
|   | 260 SXX(K,J) = 0.                                                     | TIDC109 |
|   | AA(J) = 0.                                                            | TIDC110 |
|   | 280 SXY(J) = 0.                                                       | TIDC111 |
|   | NJ2 = NTT/2 + 1                                                       | TIDC112 |
|   | DO 360 I = 1,NI                                                       | TIDC113 |
|   | DO 320 J = 1,NTT                                                      | TIDC114 |
|   | FJ1 = FLCAT(J-1)                                                      | TIDC115 |
|   | FJ3 = FLCAT ( J-NJ2 )                                                 | TIDC115 |
|   | IF ( J.LE.NJ2 ) GO TO 300                                             | TIDC117 |
|   | XX(J) = COS(FJ3*W*TT(I))                                              | TIDC118 |
|   | GO TO 320                                                             | TIDC119 |
|   | 300 XX(J) = SIN(FJ1*W*TT(I))                                          | TIDC120 |



|                                                                                   |          |
|-----------------------------------------------------------------------------------|----------|
| IF( J.EQ.1 )XX(J) = 1.                                                            | TIDC121  |
| 320 SXY(J) = SXY(J) +XX(J) *YY(I)                                                 | TIDC122  |
| DO 340 J = 1,NTT                                                                  | TIDC123  |
| DO 340 K = 1,NTT                                                                  | TIDC124  |
| 340 SXX(K,J) = SXX(K,J) +XX(K) *XX(J)                                             | TIDC125  |
| 360 CONTINUE                                                                      | TIDC126  |
| IT = 0                                                                            | TIDC127  |
| 380 IT = IT + 1                                                                   | TIDC128  |
| DELMAX = 0.                                                                       | TIDC129  |
| DO 420 K = 1,NTT                                                                  | TIDC130  |
| SUM = 0.                                                                          | TIDC131  |
| DO 400 J = 1,NTT                                                                  | TIDC132  |
| IF (J.EQ.K) GO TO 400                                                             | TIDC133  |
| SUM = SUM -AA(J)*SXX(K,J)                                                         | TIDC134  |
| 400 CONTINUE                                                                      | TIDC135  |
| SUM = (SUM+SXY(K))/SXX(K,K)                                                       | TIDC136  |
| DEL = ABS(SUM-AA(K))                                                              | TIDC137  |
| IF (DEL.GT.DELMAX ) DELMAX = DEL                                                  | TIDC138  |
| 420 AA(K) = SUM                                                                   | TIDC139  |
| IF ( IT.GE.MAXIT ) GO TO 440                                                      | TIDC140  |
| IF (DELMAX.GT.DELTA ) GO TO 380                                                   | TIDC141  |
| GO TO 460                                                                         | TIDC142  |
| 440 WRITE(N6,150)                                                                 | TIDC143  |
| 150 FORMAT (69H CANNOT REACH DESIRED DELTA, INCREASE EITHER NI OR DELTA           | TIDC144  |
| 1 AND TRY AGAIN)                                                                  | TIDC145  |
| STOP 6666                                                                         | TIDC146X |
| 460 CONTINUE                                                                      | TIDC147  |
| A1 = AA(1)                                                                        | TIDC148  |
| A2 = AA(2)                                                                        | TIDC149  |
| A3 = AA(3)                                                                        | TIDC150  |
| A4 = AA(4)                                                                        | TIDC151  |
| A5 = AA(5)                                                                        | TIDC152  |
| A6 = AA(6)                                                                        | TIDC153  |
| A7 = AA(7)                                                                        | TIDC154  |
| IF (NCHTID.NE.1) GO TO 540                                                        | TIDC155  |
| WRITE (N6,152)                                                                    | TIDC156  |
| 152 FORMAT (46H0            TIME        OBSERVED        COMPUTED        DIFF    ) | TIDC157  |
| RES = 0.                                                                          | TIDC158  |
| DO 520 I = 1,NI                                                                   | TIDC159  |
| SUM = 0.                                                                          | TIDC160  |
| DO 500 J = 2,NTT                                                                  | TIDC161  |
| FJ1 = FLOAT ( J-1 )                                                               | TIDC162  |
| FJ3 = FLOAT ( J-NJ2 )                                                             | TIDC163  |
| IF ( J.LE.NJ2 ) GO TO 480                                                         | TIDC164  |
| SUM = SUM +AA(J) *COS(FJ3*W*TT(I))                                                | TIDC165  |
| GO TO 500                                                                         | TIDC166  |
| 480 SUM = SUM +AA(J) *SIN(FJ1*W*TT(I))                                            | TIDC167  |
| 500 CONTINUE                                                                      | TIDC168  |
| SUM = SUM +AA(1)                                                                  | TIDC169  |
| DIFF = SUM -YY(I)                                                                 | TIDC170  |
| RES = RES + ABS(DIFF)                                                             | TIDC171  |
| 520 WRITE(N6,154) TT(I),YY(I),SUM,DIFF                                            | TIDC172  |
| 154 FORMAT ( 4F12.4 )                                                             | TIDC173  |
| WRITE (N6,156) RES                                                                | TIDC174  |
| 156 FORMAT (6HCTCTAL , 30X, F12.4 )                                               | TIDC175  |
| 540 CONTINUE                                                                      | TIDC176  |
| C                                                                                 | TIDC177  |
| C                                                                                 | TIDC178  |
| C                                                                                 | TIDC179  |
| CONSTANTS FOR INPUT WAVE FORM                                                     | TIDC180  |
| WRITE(N6,158)JGW,A1,A2,A3,A4,A5,A6,A7,PERIOD                                      |          |

```

158 FORMAT(///46H COEFFICIENTS FOR TIDAL INPUT WAVE AT JUNCTION I6//85HTIDC181
1      A1      A2      A3      A4      A5      A6      TIDC182
2A7      PERIOD(HRS)//7F10.3,F12.2///31H WHERE THE WAVEFORM IS GIVETIDC183
3N BY//92H  $H(J) = A1 + A2.SIN(WT) + A3.SIN(2WT) + A4.SIN(3WT) + A5.TIDC184$ 
4COS(WT) + A6.COS(2WT) + A7.COS(3WT)) TIDC185
      RETURN TIDC186
      END TIDC187

```

|   |                                                                   |         |
|---|-------------------------------------------------------------------|---------|
| C | SUBROUTINE TRIAN(II,JJ,KK,LL)                                     | TRIA 1  |
| C |                                                                   | TRIA 2  |
| C | SUBROUTINE TRIAN                                                  | TRIA 3  |
| C | HYDRODYNAMICS PROGRAM                                             | TRIA 4  |
| C | SPECIFICATION STATEMENTS                                          | TRIA 5  |
| C |                                                                   | TRIA 6  |
| C | CONTROL                                                           | TRIA 7  |
| C |                                                                   | TRIA 8  |
| C | COMMON /CCNTR/ N5,N6,N20,N21, NTCYC,NQCYC,NHCYC, NT,NQSWRT        | TRIA 9  |
| C | 1, DELTC,CELT,TZERO, ISWCH(10)                                    | TRIA 10 |
| C |                                                                   | TRIA 11 |
| C | GENERAL                                                           | TRIA 12 |
| C |                                                                   | TRIA 13 |
| C | COMMON ALPHA(30), NJ,NC, ICYC,KCYC,NCYC, WIND,WDIR,EVAP           | TRIA 14 |
| C | 1, PRECP(50),NEXIT                                                | TRIA 15 |
| C |                                                                   | TRIA 16 |
| C | JUNCTIONS                                                         | TRIA 17 |
| C |                                                                   | TRIA 18 |
| C | COMMON H(100),HN(100),HT(100),HBAR(100),HAVE(100)                 | TRIA 19 |
| C | 1, NCHAN(100,8),IPOINT(100,8),AS(100),VOL(100),X(100),Y(100)      | TRIA 20 |
| C | 2, DEP(100),COF(100),CIN(100),QOU(100),QINST(100)                 | TRIA 21 |
| C | 3, QINBAR(100),COUBAR(100)                                        | TRIA 22 |
| C |                                                                   | TRIA 23 |
| C | CHANNELS                                                          | TRIA 24 |
| C |                                                                   | TRIA 25 |
| C | COMMON LEN(225),NJUNC(225,2),B(225),R(225),A(225),AT(225),AK(225) | TRIA 26 |
| C | 1, Q(225),QBAR(225),QAVE(225), V(225),VT(225),VBAR(225)           | TRIA 27 |
| C | 2, FWIND(225),NUMCH(225),NTEMP(8)                                 | TRIA 28 |
| C | 3,NCLOS(225)                                                      | TRIA 29 |
| C |                                                                   | TRIA 30 |
| C | PRINTOUT AND PLOTTING                                             | TRIA 31 |
| C |                                                                   | TRIA 32 |
| C | COMMON NPRT,IPRT, NHPRT,JPRT(50),PRTH(30,50)                      | TRIA 33 |
| C | 1, NQPRT,CPRT(50),PRTV(30,50),PRTQ(30,50), IDUM(12),ICUL(10)      | TRIA 34 |
| C | 2, LTIME, NPLT,NPDEL,JPLT(50),HPLT(50)                            | TRIA 35 |
| C |                                                                   | TRIA 36 |
| C | STAGE-TIME COEFFICIENTS                                           | TRIA 37 |
| C |                                                                   | TRIA 38 |
| C | COMMON YY(50),TT(50),AA(10),XX(10),SXX(10,10),SXY(10)             | TRIA 39 |
| C | 1,A1,A2,A3,A4,A5,A6,A7,PERIOD,JGW                                 | TRIA 40 |
| C |                                                                   | TRIA 41 |
| C | STORMWATER                                                        | TRIA 42 |
| C |                                                                   | TRIA 43 |
| C | COMMON TITLE(30),NJSW,QE(20,2),JSW(20)                            | TRIA 44 |
| C | 2, RAIN(100),INTIME(100),INRAIN,JBOUND(20),JJBOUN                 | TRIA 45 |
| C |                                                                   | TRIA 46 |
| C | TAPES                                                             | TRIA 47 |
| C |                                                                   | TRIA 48 |
| C | COMMON /TAPES/ INCNT,IQUTCT,JIN(10),JOUT(10),NSCRAT(5)            | TRIA 49 |
| C |                                                                   | TRIA 50 |
| C |                                                                   | TRIA 51 |
| C | COMMON /TRI/T(5),NX(5)                                            | TRIA 52 |
| C |                                                                   | TRIA 53 |
| C |                                                                   | TRIA 54 |
| C | TYPE DESIGNATIONS                                                 | TRIA 55 |
| C |                                                                   | TRIA 56 |
| C | INTEGER CPRT                                                      | TRIA 57 |
| C | REAL LEN                                                          | TRIA 58 |
| C | IF(II.NE.0) GO TO 300                                             | TRIA 59 |
| C |                                                                   | TRIA 60 |

|     |                                                           |                                  |         |
|-----|-----------------------------------------------------------|----------------------------------|---------|
| C   |                                                           | ZERO POINTER ARRAY               | TRIA 61 |
|     | DO 250 I=1,NJ                                             |                                  | TRIA 62 |
|     | DO 250J=1,8                                               |                                  | TRIA 63 |
|     | IPOINT(I,J)=0                                             |                                  | TRIA 64 |
|     | NCHAN(I,J)=0                                              |                                  | TRIA 65 |
|     | 250 CONTINUE                                              |                                  | TRIA 66 |
|     | RETURN                                                    |                                  | TRIA 67 |
| C   |                                                           |                                  | TRIA 68 |
| C   |                                                           | SET UP TRIANGLE PARAMETERS       | TRIA 69 |
| C   |                                                           |                                  | TRIA 70 |
|     | 300 CONTINUE                                              |                                  | TRIA 71 |
|     | NX(1)=II                                                  |                                  | TRIA 72 |
|     | NX(2)=JJ                                                  |                                  | TRIA 73 |
|     | NX(3)=KK                                                  |                                  | TRIA 74 |
|     | NX(4)=II                                                  |                                  | TRIA 75 |
|     | NX(5)=JJ                                                  |                                  | TRIA 76 |
|     | T(1) = (X(JJ) - X(KK))**2 + (Y(JJ) - Y(KK))**2            |                                  | TRIA 77 |
|     | T(2) = (X(KK) - X(II))**2 + (Y(KK) - Y(II))**2            |                                  | TRIA 78 |
|     | T(3) = (X(II) - X(JJ))**2 + (Y(II) - Y(JJ))**2            |                                  | TRIA 79 |
|     | T(4)=T(1)                                                 |                                  | TRIA 80 |
|     | T(5)=T(2)                                                 |                                  | TRIA 81 |
| C   |                                                           |                                  | TRIA 82 |
| C   |                                                           | DO ALL THREE SIDES               | TRIA 83 |
| C   |                                                           |                                  | TRIA 84 |
|     | NB=2                                                      |                                  | TRIA 85 |
|     | IF(LL.EC.0) NB=1                                          |                                  | TRIA 86 |
|     | DO 600 N=1,3,NB                                           |                                  | TRIA 87 |
| C   |                                                           |                                  | TRIA 88 |
| C   |                                                           | LOCATE CHANNELS ON POINTER ARRAY | TRIA 89 |
| C   |                                                           |                                  | TRIA 90 |
|     | I=MIND(NX(N+1),NX(N+2))                                   |                                  | TRIA 91 |
|     | J=MAXD(NX(N+1),NX(N+2))                                   |                                  | TRIA 92 |
|     | DO 350 K=1,8                                              |                                  | TRIA 93 |
|     | IF(IPOINT(I,K).EQ.J) GO TO 370                            |                                  | TRIA 94 |
|     | IF(IPOINT(I,K).EQ.0) GO TO 360                            |                                  | TRIA 95 |
|     | 350 CONTINUE                                              |                                  | TRIA 96 |
|     | 360 IPOINT(I,K)=J                                         |                                  | TRIA 97 |
|     | NC=NC+1                                                   |                                  | TRIA 98 |
|     | NCHAN(I,K)=NC                                             |                                  | TRIA 99 |
|     | 370 N=NCHAN(I,K)                                          |                                  | TRIA100 |
| C   |                                                           |                                  | TRIA101 |
| C   |                                                           | M IS CHANNEL NUMBER JUST         | TRIA102 |
| C   |                                                           | ASSIGNED                         | TRIA103 |
| C   |                                                           |                                  | TRIA104 |
|     | NJUNC(M,1)=I                                              |                                  | TRIA105 |
|     | NJUNC(M,2)=J                                              |                                  | TRIA106 |
|     | SUB=T(N+1)+T(N+2)-T(N)                                    |                                  | TRIA107 |
|     | G=SQRT(T(N))/2.                                           |                                  | TRIA108 |
|     | LEN(M)=2.*G                                               |                                  | TRIA109 |
|     | C=G/SQRT(4.*T(N+2)*T(N+1)-SUB**2)*SUB                     |                                  | TRIA110 |
|     | G=G/2.*C                                                  |                                  | TRIA111 |
|     | AS(I)=AS(I)+G                                             |                                  | TRIA112 |
|     | AS(J)=AS(J)+G                                             |                                  | TRIA113 |
|     | IF(C.LE.C.) WRITE(6,102) M,C                              |                                  | TRIA114 |
| 102 | FORMAT(' NEGATIVE WIDTH CHANNEL NO.',I5,' WIDTH =',E12.4) |                                  | TRIA115 |
|     | B(M)=E(M)+C                                               |                                  | TRIA116 |
|     | R(M)=(CEP(I)+DEP(J))/2.                                   |                                  | TRIA117 |
|     | A(M)=B(M)*R(M)                                            |                                  | TRIA118 |
|     | AK(M)=(COF(I)+COF(J))/2.                                  |                                  | TRIA119 |
|     | V(M)=C.                                                   |                                  | TRIA120 |

|                                   |         |
|-----------------------------------|---------|
| 600 CONTINUE                      | TRIA121 |
| IF(LL.EQ.0) RETURN                | TRIA122 |
| DO 750 NN=3,4                     | TRIA123 |
| I=MINO(NX(NN),LL)                 | TRIA124 |
| J=MAXO(NX(NN),LL)                 | TRIA125 |
| DO 620 K=1,8                      | TRIA126 |
| IF(IPOINT(I,K).EQ.J) GO TO 640    | TRIA127 |
| IF(IPOINT(I,K).EQ.0) GO TO 630    | TRIA128 |
| 620 CONTINUE                      | TRIA129 |
| 630 IPOINT(I,K)=J                 | TRIA130 |
| NC=NC+1                           | TRIA131 |
| NCHAN(I,K)=NC                     | TRIA132 |
| 640 M=NCHAN(I,K)                  | TRIA133 |
| NJUNC(M,1)=I                      | TRIA134 |
| NJUNC(M,2)=J                      | TRIA135 |
| SUB=T(3)+T(4)-T(2)                | TRIA136 |
| G=SQRT(T(2))/2.                   | TRIA137 |
| LEN(M)=G                          | TRIA138 |
| C=G/SQRT(4.*T(3)*T(4)-SUB**2)*SUB | TRIA139 |
| G=G/2.*C                          | TRIA140 |
| AS(I)=AS(I)+G/2.                  | TRIA141 |
| AS(J)=AS(J)+G/2.                  | TRIA142 |
| IF(C.LE.0.) WRITE(6,102) M,C      | TRIA143 |
| B(M)=B(M)+C                       | TRIA144 |
| R(M)=(DEP(I)+DEP(J))/2.           | TRIA145 |
| A(M)=B(M)*R(M)                    | TRIA146 |
| AK(M)=(CCF(I)+COF(J))/2.          | TRIA147 |
| V(M)=0.                           | TRIA148 |
| 750 CONTINUE                      | TRIA149 |
| RETURN                            | TRIA150 |
| END                               | TRIA151 |

|   |                  |   |                       |                                                              |
|---|------------------|---|-----------------------|--------------------------------------------------------------|
| 1 | Accession Number | 2 | Subject Field & Group | SELECTED WATER RESOURCES ABSTRACTS<br>INPUT TRANSACTION FORM |
|   |                  |   | 013B                  |                                                              |

|   |              |                                                                                                                                                                                 |
|---|--------------|---------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| 5 | Organization | Metcalf & Eddy, Inc., Palo Alto, California<br>Florida University, Gainesville, Dept. of Environmental Engineering<br>Water Resources Engineers, Inc., Walnut Creek, California |
|---|--------------|---------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|

|   |       |                              |
|---|-------|------------------------------|
| 6 | Title | STORM WATER MANAGEMENT MODEL |
|---|-------|------------------------------|

|    |                                                                  |    |                                                                                                                                                   |
|----|------------------------------------------------------------------|----|---------------------------------------------------------------------------------------------------------------------------------------------------|
| 10 | Author(s)                                                        | 16 | Project Designation                                                                                                                               |
|    | Lager, John A.,<br>Pyatt, Edwin E., and<br>Shubsinski, Robert P. |    | EPA Contract Nos. 14-12-501, 502, 503                                                                                                             |
|    |                                                                  | 21 | Note                                                                                                                                              |
|    |                                                                  |    | Set of four volumes: Volume I - Final Report,<br>Volume II - Verification and Testing, Volume III -<br>User's Manual, Volume IV - Program Listing |

|    |          |
|----|----------|
| 22 | Citation |
|----|----------|

|    |                                                                                                                                                                                |
|----|--------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| 23 | Descriptors (Starred First)                                                                                                                                                    |
|    | Water Quality Control*, Computer Model*, Storm Water*, Simulation Analysis, Rainfall-<br>Runoff Relationships, Sewerage, Storage, Waste Water Treatment, Cost Benefit Analysis |

|    |                                         |
|----|-----------------------------------------|
| 25 | Identifiers (Starred First)             |
|    | Combined Sewer Overflows*, Urban Runoff |

|    |                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                           |
|----|-------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| 27 | Abstract                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                  |
|    | <p>A comprehensive mathematical model, capable of representing urban storm water runoff, has been developed to assist administrators and engineers in the planning, evaluation, and management of overflow abatement alternatives. Hydrographs and pollutographs (time varying quality concentrations or mass values) were generated for real storm events and systems from points of origin in real time sequence to points of disposal (including travel in receiving waters) with user options for intermediate storage and/or treatment facilities. Both combined and separate sewerage systems may be evaluated. Internal cost routines and receiving water quality output assisted in direct cost-benefit analysis of alternate programs of water quality enhancement. Demonstration and verification runs on selected catchments, varying in size from 180 to 5,400 acres, in four U.S. cities (approximately 20 storm events, total) were used to test and debug the model. The amount of pollutants released varied significantly with the real time occurrence, runoff intensity duration, pre-storm history, land use, and maintenance. Storage-treatment combinations offered best cost-effectiveness ratios. A user's manual and complete program listing were prepared.</p> |

|                                  |               |                                                                                                                      |                                       |
|----------------------------------|---------------|----------------------------------------------------------------------------------------------------------------------|---------------------------------------|
| Abstractor                       | John A. Lager | Institution                                                                                                          | Project Manager, Metcalf & Eddy, Inc. |
| WR-102 (REV. JULY 1969)<br>WRSIC |               | SEND TO: WATER RESOURCES SCIENTIFIC INFORMATION CENTER<br>U.S. DEPARTMENT OF THE INTERIOR<br>WASHINGTON, D. C. 20240 |                                       |