7.2 LISTING OF THE MODULES MODEL, INITIAL AND FINAL

```
4630 ! -----------MODEL
4640 REM MAIN SUBR TO CALC F(i),
     K(i,z)FOR ALL i AND z=1,2,3
     ,
4650 ! **Y*******UNSTEADY FLOWS
4660 ! ++++++++++CASCADE OUTFLOW
     &SETTLER INFLOW
4670 Z3=FP(T/7)*7 ! DAY OF WEEK
4680 IF Z3>5 THEN 4710 ! MON.=DA
     Y1
4690 ! WEEKDAY FLOW
4700 A(65)=FND(1)*B1 @ A(66)=FND
     (3)*B2 @ GOTO 4780
4710 ! WEEKEND FLOW
4720 A(65)=FND(2)*B1 @ A(66)=FND
     (   *B2
4780     ++++++++++++MPS
4790      1(67)>0 THEN 4830
480(     UMPS OFF
48     r A(1)>=A(42) THEN A(67)=A
     (20)
4820 GOTO 4850
4830 ! PUMPS ON
4840 IF A(1)<=A(43) THEN A(67)=0
4850 F(1)=A(66)-A(67)
4860 ! ++++++++++++++IPS
4870 IF A(68)>0 THEN 4910
4880 ! PUMPS OFF
4890 IF A(2)>=A(44) THEN A(68)=A
     (21)
4900 GOTO 4930
4910 ! PUMPS ON
4920 IF A(2)<=A(45) THEN A(68)=0
4930 F(2)=A(67)-A(68)
4940 ! ++++++++++SED.TANK
4950 A(80)=A(36)*A(66) ! MUD DIS
     P FLOW
4960 IF A(3)>=A(46) THEN A(69)=M
     AX(0,A(68)-A(80)) ELSE A(69
     )=0
4970 F(3)=A(68)-A(80)-A(69)
4980 ! +++++++++++HOTWELL
4990 ! MAKEUP-RWB&SUPPLY DAM
5000 IF A(4)<=A(47) AND G1=0 THE
     N A(74)=A(23) ELSE A(74)=0
5010 IF A(4)<=A(48) AND G2=0 THE
     N A(76)=A(25) ELSE A(76)=0
5020 ! OUTFLOW
5030 IF A(4)>0 AND A(5)<A(52) TH
     EN A(70)=A(22) ELSE A(70)=0
5035 F(4)=A(59)+A(74)+A(76)-A(70
     ) ! NET INFLW
5040 ! OVERFLOW
5050 A(78)=0 ! ASSUME
5060 IF A(4)<A(49) OR F(4)<=0 TH
     EN 5080
5070 A(78)=F(4) @ F(4)=0
5080 ! +++++++++REFRIGERATION
5090 A(71)=A(70)*(1-A(33)-A(34))
5100 ! +++++++++++COLDWELL
```

```
5110 ! MAKEUP-RWB&SUPPLY DAM
5120 IF A(5)>A(50) THEN 5150
5130 IF A(74)=0 THEN A(75)=A(24)
5140 G1=1 @ GOTO 5160
5150 A(75)=0 @ G1=0
5160 IF A(5)>A(51) THEN 5190
5170 IF A(76)=0 THEN A(77)=A(26)
5180 G2=1 @ GOTO 5200
5190 A(77)=0 @ G2=0
5200 F(5)=A(71)+A(75)+A(77) ! TO
     T INFLW
5210 ! OUTFLOW
5220 IF A(5)>0 THEN 5250
5225 Z3=MIN(A(27),A(65))
5230 IF A(6)<A(53) THEN A(72)=MI
     N(F(5),A(27)) ELSE A(72)=MI
     N(F(5),Z3)
5240 GOTO 5260
5250 IF A(6)<A(53) THEN A(72)=A(
     27) ELSE A(72)=MIN(A(65),A(
     27))
5260 F(5)=F(5)-A(72) ! NET INFLW
5270 ! ++++++++U/G STORAGE DAM
5280 F(6)=A(72)-A(65)
5285 ! +++++++CASCADES
5286 F(7)=0
5290 ! *******************SALTS
5300 IF G=1 THEN 5540
5310 ! ++++++++++WORKINGS
5320 Z3=A(66)+A(32)-A(37)
5330 A(32)=FNQ(1) ! Ccascade
5340 A(79)=A(60)*(A(61)-A(82))^A
     (62) ! mg/l LEACHED
5350 A(83)=(Z3*(A(79)+A(82))+A(3
     7)*A(59))/A(66) ! Cworkings
5360 ! +++++++++++SETTLER
5365 A(85)=A(8)/A(55)
5370 F(8)=A(66)*(A(83)-A(85))
5380 ! +++++++++++MPS
5385 A(86)=A(9)/A(1)
5390 F(9)=A(66)*A(85)-A(67)*A(86
     )
5400 ! +++++++++++IPS
5405 A(87)=A(10)/A(2)
5410 F(10)=A(67)*A(86)-A(68)*A(8
     7)
5420 ! ++++++++++SED.TANK
5425 A(88)=A(11)/A(3)
5430 F(11)=A(68)*A(87)-(A(69)+A(
     80))*A(88)
5440 ! +++++++++HOTWELL
5445 A(89)=A(12)/A(4)
5450 F(12)=A(69)*A(88)+A(74)*A(5
     7)+A(76)*A(58)-(A(70)+A(78)
     )*A(89)
5460 ! +++++REFRIGERATION
5470 IF A(71)=0 THEN A(84)=0 ELS
     E A(84)=A(89)*A(70)*(1-A(34
     ))/A(71)
5480 ! +++++++++COLDWELL
```

```
5485 A(90)=A(13)/A(5)
5490 F(13)=A(71)*A(84)+A(75)*A(5
     7)+A(77)*A(58)-A(72)*A(90)
5500 ! ++++++++U/G STORAGE DAM
5505 A(91)=A(14)/A(6)
5510 F(14)=A(72)*A(90)-A(65)*A(9
     1)
5520 ! ++++++++CASCADE DAMS
5525 A(92)=A(15)/A(7)
5530 F(15)=A(65)*(A(91)-A(92))
5540 ! *********END OF SALTS
5550 ! ++INTEGRATE A(74)-A(80)
5560 FOR J3=74 TO 80
5570 Z5(J3-73,Z)=A(J3)
5580 NEXT J3
5590 IF F2#1 AND T<T1+H THEN 567
     0
5600 FOR J3=95 TO 101
5610 Z3=J3-94
5620 ON F2 GOTO 5630,5640,5650
5630 A(J3)=A(J3)+H*Z5(Z3,1) @ GO
     TO 5660
5640 A(J3)=A(J3)+H*(Z5(Z3,1)+Z5(
     Z3,2))/2 @ GOTO 5660
5650 A(J3)=A(J3)+H*(Z5(Z3,1)+2*(
     Z5(Z3,2)+Z5(Z3,3))+Z5(Z3,4)
     )/6
5660 NEXT J3
5662 ! +++++TOT.VOLUME&MASS
5663 A(93),A(81)=0
5664 FOR J3=1 TO 7
5665 A(93)=A(93)+A(J3) @ NEXT J3
5666 FOR J3=8 TO 15
5667 A(81)=A(81)+A(J3) @ NEXT J3
5670 ! ***********END OF MODEL
7960 FOR J=1 TO N0
7970 K(J,Z)=H*F(J) ! Z=1,2,3OR4
7980 NEXT J
7990 RETURN
8000 ! -----------INITIAL
8010 ! SUBR DONE BEFORE SIMUL'N
8020 DISP "1)UNSTEADY FLOWS&NO S
     ALTS"
8021 DISP "2)UNSTEADY FLOWS&UNST
     EADYSALTS  (1or2)";@ INPUT
     G
8025 IF G#1 AND G#2 THEN 8020
8026 G1,G2=0 ! NO COLD-W MAKEUP
8030 DISP USING 8040 ; "UNISEL G
     OLD MINE","Detailed unstead
     y flow and","unsteady salt
     model."
8040 IMAGE /,3(32A),32("=")
8050 DISP "UNITS:Ml,d,kg.(NOTE:k
     g/Ml=mg/l)"
8060 DISP @ DISP "CAPACITY FLOW
     RATES(Ml/d):"
8070 DISP " PUMPS:MPS,IPS,HOT-CO
     LDWELL     ";A(20);A(21);A(2
     2)
```

```
8080 DISP " MAKEUP WATER TO HOT&
     COLDWELLS:   RWB";A(23);A(2
     4)
8090 DISP "  SUP DAM";A(25);A(26
     )
8100 DISP " COLDWELL TO INTERMED
     .DAM";A(27)
8110 DISP @ DISP "AVERAGE FLOWS
                   (Ml/d)"
8120 DISP USING 8130 ; " INFILTR
     ATION",A(37)," EVAPORATION"
     ,A(32)
8130 IMAGE 24A,X,K,/
8135 DISP TAB(26);"RATIOS"
8140 DISP USING 8130 ; " TOWER E
     VAP/INFLOW",A(33)," TOWER W
     INDAGE/INFLOW",A(34)
8150 DISP USING 8130 ; " ORELOSS
     /CASCADE FLOW",A(35)," MUD/
     SETTLER INFLOW",A(36)
8160 ! +++++++HYDROGRAPH FACTORS
8170 B1=A(38)/A(39)
8180 Z3=A(38)+7*(A(37)-A(32))-A(
     35)*A(38) ! WEEKLY SETTLER
     INFLW
8190 B2=Z3/A(40)
8195 DISP TAB(23);"(Ml/week)"
8200 DISP USING 8130 ; " MINE IN
     FLOW",A(38)," SETTLER INFLO
     W",Z3
8210 DISP @ DISP "VOLUMES(Ml):"
8220 DISP " START-STOP VOLUMES:"
8230 DISP "  MPS PUMP";;A(42);A
     (43) @ DISP "  IPS PUMP";A(
     44);A(45)
8240 DISP " SWITCH ON VOLUMES FO
     R MAKEUP:"
8250 DISP "  HOTWELL-RWB&SUP.DAM
     ";A(47);A(48) @ DISP "  COL
     DWELL-RWB&SUP.DAM";A(50);A(
     51)
8260 DISP " FULL CAPACITY:"
8270 DISP "  SED.TANK,HOTWELL,CO
     LDWELL";A(46);A(49);A(52)
8280 DISP "  INTERMED&CASCADE DA
     MS,SETTLERS";A(53);A(54);A(
     55)
8290 IF G=1 THEN GOTO 8800
8300 DISP @ DISP "SALT CONCENTRA
     TION(kg/Ml=mg/l):"
8310 DISP " MAKEUP-RWB&SUP.DAM";
     A(57);A(58)
8320 DISP " INFILTRATION";A(59)
8330 DISP " LEACHING=X(Y-Ccascad
     e)^Z" @ DISP " X,Y,Z";A(60)
     ;A(61);A(62)
8800 ! DATA FOR 1)INIT VALS
8802 ! 1TO7
8804 DATA 1.3,.45,1.6,4,2,2,1.5
8806 ! 8TO19
```

```
8808 DATA 1260,910,315,1120,2900
     ,1400,1400,1050,0,0,0,0
8810 ! 20TO29
8812 DATA 14.7,14.7,11.5,.5,.5,5
     .4,5,4,26,0,0
8814 ! 30TO40
8816 DATA 0,0,.4,.05,.01,.2,.08,
     .5,70,2,619,1.478
8818 ! 41TO56
8820 DATA 0,2.9,1.6,1,.4,1.6,1.6
     ,1.6,4,.8,.8,2,2,1.5,1.8,0
8822 ! 57TO64
8824 DATA 700,12000,4000,.1,1600
     0,1,0,0
8826 ! 65TO73
8828 DATA 2,.1,0,0,0,0,0,2,0
8830 ! 74TO81
8832 DATA 0,0,0,0,0,1530,0,10250
8834 ! 82TO84
8836 DATA 700,700,700
8838 ! 85TO94
8840 DATA 700,700,700,700,700,70
     0,700,700,0,0
8842 ! 95TO101
8844 DATA 0,0,0,0,0,0,0
8850 ! DATA FOR 4)PRT LIST
8852 DATA 13,1,2,3,4,5,6,7,95,96
     ,97,98,99,101
8870 ! DATA FOR 6)PLOT INPT
8872 DATA 12,4000
8876 DATA 1
8878 DATA 1,0,4,1,2,0,1.5,1,3,1.
     5,1.7,.1,4,0,4.5,1,5,0,2.5,
     1,66,0,20,5
8880 DATA 67,0,20,5,69,0,20,5,70
     ,0,20,5,72,0,20,5,97,0,1,1,
     98,0,1,1
8900 ! DATA FOR 10)TABLES
8902 DATA 4
8904 DATA 11,6,2,4,11,5,2,6
8906 DATA 0,1,2,3,4,7,8,9,10,11,
     14
8908 DATA 5,12
8910 DATA 0,1,2,3,4,7,8,9,10,11,
     14
8912 DATA 5,12
8914 DATA 0,.2,.25,.2,.5,.8,.75,
     .5,.875,.7,1,.2
8916 DATA 0,.2,.25,.2,.5,.8,.75,
     0
8918 DATA 0,.01,.25,.01,.5,.5,.7
     5,.4,1,.01
8920 DATA 0,.01,.25,.01,.5,.5,.7
     5,.4,1,.01,2,.01
8970 ! DATA FOR 11)QUEUES
8972 DATA 1,92,.02,5,700
8999 RETURN
9000 ! --------------FINAL
9010 ! SUBR TO PERFORM ANY USER-
     DEFINED OPERATIONS AFTER A
     RUN
```

```
9020 DISP "TIME STEP";H;"DAYS"
9025 DISP "START&END TIME:";T0;T
     9
9029 DISP "ALGORITHM:";
9030 IF F2=1 THEN DISP "EULER"
9032 IF F2=2 THEN DISP "MOD.EULE
     R"
9034 IF F2=3 THEN DISP "4th ORDE
     R R-KUTTA"
9040 DISP @ DISP "CUMULATIVE OUT
     PUT(M1)"
9050 DISP " MAKE UP WATER TO HOT
     &COLDWELLS"
9060 DISP "  RWB    ";A(95);A(96
     ) @ DISP "  SUP.DAM";A(97);
     A(98)
9070 DISP " HOTWELL OVERFLOW";A(
     99)
9080 DISP " MUD REMOVED";A(101)
9090 DISP " SALTS LEACHED(kg)";A
     (100)
9999 RETURN
```

7.3 LIST OF SYMBOLS USED IN THE MODEL

In this model 101 program variables (A(1) to A(101)) and 5 *standard* BASIC variables are used. The values of program variables can easily be entered, changed or displayed as output using the facilities of the simulation program, whereas programming changes are required to alter BASIC variables. This can be seen when studying the model. The program variables A(1) to A(101) are divided into groups of different types of variables (see section 5.2). Each group is discussed and each program variable described. The simulation program requires that each program variable has an initial value before simulation commences therefore, The initial value of each variable and its units are also given. The program variables are discussed first. Fig. 7.1 indicates the location of the system components referred to.

7.3.1 List of Program variables

* Indicates that the variable is not used at present but can be used in future modifications to the model.

The 15 dependent state variables (A(1) to A(15)) result from the simultaneous solution of the 15 differential equations during each iteration. There are two types; storage element volumes and mass of salts in the storage elements.

VOLUMES (Mℓ)		(Initial value)
A(1)	Main pump station	1,3
A(2)	Intermediate pumpst[illegible] [illegible]PS)	0,45
A(3)	Sedimentation tank	1,6
A(4)	Hotwell	4
A(5)	Coldwell	2
A(6)	Underground storage dam	2
A(7)	Cascade dams	1,5

SALT MASS (kg)		
A(8)	Settlers	1260
A(9)	MPS	910
A(10)	IPS	315
A(11)	Sedimentation tank	1120
A(12)	Hotwell	2800
A(13)	Coldwell	1400
A(14)	Underground storage dam	1400
A(15)	Cascade dams	1050
A(16)	*	0
A(17)	*	0
A(18)	*	0
A(19)	*	0

The following independent variables or parameters are specified by the user and do not change during simulation:

CAPACITY FLOW RATES (Mℓ/d)

A(20)	MPS pump	14,7
A(21)	IPS pump	14,7
A(22)	Hotwell to ref igeration pump	11,5
A(23)	Make-up:R.W.B. to Hotwell	0,5
A(24)	to Coldwell	0,5
A(25)	Supply dam to Hotwell	5,4
A(26)	to Coldwell	5,4
A(27)	Coldwell to underground storage dam	26,0
A(28)	*	0
A(29)	*	0
A(30)	*	0
A(31)	*	0

AVERAGE FLOWS AND RATIOS

A(32)	Evaporation out of the mine (Mℓ/d)	0,4
A(33)	Evaporation loss from refrigeration / flow from Hotwell	0,05
A(34)	Windage loss from refrigeration / flow from Hotwell	0,01
A(35)	Water removed with ore / cascade outflow	0,2
A(36)	Mud removed from settlers / Settler inflow	0,08
A(37)	Groundwater infiltration (Mℓ/d)	0,5
A(38)	Total weekly outflow fromthe coldwell,	70,0

RELATIVE FLOWS

A(39)	Total area of the relative-cascade-outflow versus time-graph for 1 week (see 7.4.1.1)	2,0187
A(40)	Total area of the relative-settler-inflow versus time graph for 1 week	1,4798

A(41)	*	0

STORAGE ELEMENT VOLUMES (Mℓ)

A(42)	MPS pump start	2,9
A(43)	stop	1,6
A(44)	IPS pump start	1,0
A(45)	stop	0,4
A(46)	Sedimentation Tank - full	1,6
A(47)	Start/stop make-up to Hotwell from RWB	1,6
A(48)	Start/stop make-up to Hotwell from Supply Dam	1,6
A(49)	Hotwell - full	4,0
A(50)	Start/stop make-up to Coldwell from RWB	0,8
A(51)	Start/stop make-up to Coldwell from Supply Dam	0,8
A(52)	Coldwell - full	2,0
A(53)	Underground storage dam - full	2,0
A(54)	Cascade dams - full	1,5
A(55)	Settlers - full (permanently)	1,8
A(56)	*	0

SALT CONCENTRATIONS (kg/Mℓ

A(57)	Make-up water: RWB	700
A(58)	Supply Dam	12000
A(59)	Groundwater infiltration	4000

LEACHING REACTION PARAMETERS (see 7.4.2.1&A(79))

A(60)	X factor	0,1
A(61)	Y saturation salt concentration (kg/Mℓ)	16000
A(62)	Z order of the reaction	1,0
A(63)	*	0

The following dependent internal variables are generated by the system from a knowledge of the operating rates and relationships programmed into the model. These variables affect the state of the system and may be of interest as output:

	INTERNAL FLOW RATES (Mℓ/d)	
A(64)	*	0
A(65)	Cascade outflow	2
A(66)	Settler inflow from the workings	0,1
A(67)	From: MPS	0
A(66)	IPS	0
A(69)	Sedimentation Tank to Hotwell	0
A(70)	Hotwell to Refrigeration Plant	0
A(71)	Refrigeration to Coldwell	0
A(72)	Coldwell to underground	2
	EXTERNAL FLOWRATES (Mℓ/d)	
A(73)	*	0
A(74)	Make-up from RWB to Hotwell	0
A(75)	to Coldwell	0
A(76)	from Supply Dam to Hotwell	0
A(77)	from Supply Dam to Coldwell	0
A(78)	Overflow from the Hotwell when full	0
A(79)	LEACHING SALTS ADDED to used mine service water (kg/Mℓ)=$X(Y-A(82))^Z$ X,Y and Z are A(60),A(61) and A(62) above	1530
A(80)	Mud pumped from the mine	0
A(81)	TOTAL SALT MASS IN THE SYSTEM (kg)	10250
	SALT CONCENTRATIONS (kg/Mℓ)	
A(82)	Cascade outflow at the lag time (or travel time) earlier in the simulation	700
A(83)	To the settlers	700
A(84)	Coldwell	700
A(85)	Of the settlers	700
A(86)	MPS	700
A(87)	IPS	700
A(88)	Sedimentation Tank	700
A(89)	Hotwell	700
A(90)	Coldwell	700
A(91)	Underground storage Dam	700
A(92)	Cascade Dams	700

A(93) TOTAL WATER VOLUME IN THE SYSTEM (Mℓ) 0

The following dependent output variables are calculated by the model during simulation by integrating variables A(74) to A(80) over time. These variables are generated for output purposes and do not affect the state of the system:

CUMULATIVE FLOWS (Mℓ)

A(94)	*	0
A(95)	Make-up from RWB to Hotwell	0
A(96)	to Coldwell	0
A(97)	from Supply Dam to Hotwell	0
A(98)	to Coldwell	0
A(99)	Overflow from Hotwell	0
A(100)	Mud pumped from the mine	0

A(101) CUMULATIVE SALT LEACHED (kg) 0

7.3.2 LIST OF STANDARD BASIC VARIABLES

G Flag set in INITIAL and used by MODEL to find which of the two sections of the model are executed: 1 - only flows and no salts in the simulation; 2 - both salts and flows.

G1 Flag used in the model: 0 - coldwell not using make-up from RWB therefore hotwell can use if required; 1 - coldwell is using RWB make-up therefore hotwell may not use. Coldwell has priority.

G2 Flag for make-up water from surface supply dam. Used as G1 above.

Z3 Temporary storage variable

J3 Do-loop counter

FNQ(1) Queue number one is used for storing earlier values of cascade dam salt concentration.

The following external variables of time-flowrate relationships are specified by the user and act upon the system. These relationships are stored as data pairs in tables and are called at various times by the model:

FND(1) Weekend demand from the cascade dams.

FND(2) Weekday settler inflow.

FND(4) Weekend settler inflow.

7.4 DISCUSSION OF THE MODULE-MODEL

The model is divided into two sections; the first dealing with water flows and volumes (lines 4650-5286) and the second dealing with salt masses and concentrations (lines 5290-5540). Before simulation begins, the flag G is set to 1 or 2 depending on whether the user wants to simulate flow with no salts or both salts and flows (see line 8021 of module INITIAL). When simulation begins and the module MODEL is called during iterations, the check on flag G (line 5300) will result in the salt section of the model being included or excluded. The two sections of the model are discussed separately.

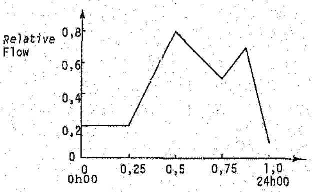
7.4.1 Unsteady Water Flows and Storage Volumes (lines 4650-5286)

7.4.1.1 Cascade Outflow and Settler Inflow (lines 4660-4720)

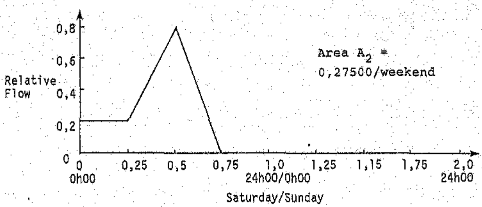
The water reticulation system in Unisel Gold Mine is designed to satisfy the demand for mine service water from the mine workings and to cope with the used mine service water and infiltration which leaves the workings and gravitates to the bottom of the mine. The water demand and hence the used water flows are a function of the time of day and of the particular day of the week. The water flow, collected in the settlers at the bottom of the mine, consists of infiltration and used mine service water. The flowrate-time-day-of-the week relationship of used mine service water eventually entering the settlers is somewhat different to that of the mine service water demand relationship. Differences occur because of different travel times from different parts of the mine and because of the effects of evaporation and water removed with the ore. An attempt was made to determine a relationship between the mine service water demand and the settler inflow but this involved too many unknowns. The model as it now stands therefore relies on in situ measurements of the flowrate-time-day of the week relationships for mine service water demand and settler inflow.

The graphs 7.1 to 7.4 show the relative flows at different times of the day for water leaving the cascade dam and for water entering the settlers. The cascade outflow or settler inflow pattern for each weekday, (Monday to Friday) and for each weekend is assumed to repeat itself from week to week. The graphs 7.1 to 7.4 have been estimated from other graphs of hotwell and coldwell levels and graphs of hotwell inflow and coldwell outflow. The latter graphs are automatically plotted 24 hours a day using level and flow recorders which are part of the monitor and control system used on the mine. Unfortunately the graphs can be used only to obtain a rough estimate of the relationships for cascade outflow and settler inflow. The simulation results are very dependent on these relationships and in situ measurem ts are currently being done to determi.. better estimates of the true relationships. The valt es shown are relative values and are converted to actual flow rates using the correction factors B1 and B2 discussed below.

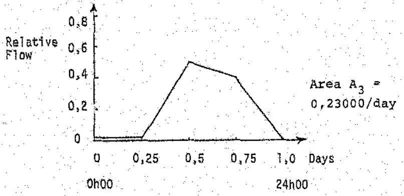
Graphs of relative flow rates versus time



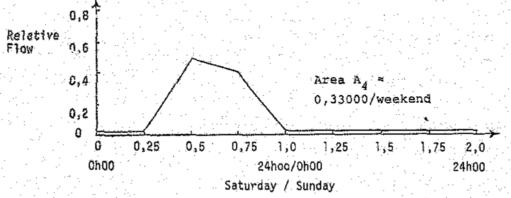
Graph 7.1 - Weekday .ascade Outflow



Graph 7.2 - Weekend Cascade Outflow



Graph 7.3 - Weekday Settler Inflow



Graph 7.4 - Weekend Settler Inflow

The coordinates of the 4 graphs are stored in 4 separate tables by module, STO TABLE of the simulation progam. Actual flows are calculated at any time in the simulation by interpolating the relative flow from the relevant table (module USE TABLE) and multiplying it by a correction factor (B1 or B2). By converting the simulation time (which may be in excess of 7 days) to a value between 1 and 7 days (Monday to Sunday) it can be seen whether the weekend or weekday tables are to be used (lines 4660 to4680). The table values are interpolated at the simulation time by using the function statement FND (table number). For example,

A(66)=FND(3)*B2

means that the settler inflow, A(66) equals the relative flow from table 3 multiplied by the correction factor, B2. The simulation time, T falls within a weekday and therefore table 3 (graph 73 above) is used and not table 4.

```
4630 ! ------------MODEL
4640 REM MAIN SUBR TO CALC F(i),
     K(i,z)FOR ALL i AND z=1,2,3
     ...
4650 ! **********UNSTEADY FLOWS
4660 ! ++++++++++CASCADE OUTFLOW
     &SETTLER INFLOW
4670 Z3=FP(T/7)*7 ! DAY OF WEEK
4680 IF Z3>5 THEN 4710 ! MON.=DA
     Y1
4690 ! WEEKDAY FLOW
4700 A(65)=FND(1)*B1 @ A(66)=FND
     (3)*B2 @ GOTO 4780
4710 ! WEEKEND FLOW
4720 A(65)=FND(2)*B1 @ A(66)=FND
     (4)*B2
```

The correction factors B1 and B2 are applicable to cascade outflow and settler inflow calculations respectively, for both weekday and weekend flows. They are both constants for any simulation and are therefore calculated in the module INITIAL, just before simulation begins (lines 8170 to 8190).

$$B1 = \frac{\text{total weekly outflow from the coldwell } A(38)}{\text{total graph area for 1 week } A(39)}$$

where A(39) is calculated by the user as follows and entered as a parameter of the model:

A(39)= 5x area graph F1 (Mon.to Fri.) + area graph F2 (Sat. and Sun.) Similarly,

$$B2 = \frac{\text{total weekly inflow to the settlers } Z3}{\text{total graph area for 1 week } A(39)}$$

where Z3 equals the weekly outflow from the cascade dams (and coldwell)A(38) plus the weekly infiltration 7*A(37) less the weekly mine evaporation,7*A(32), less water removed with the ore, A(35)*A(38).

7.4.1.2 Main Pump Station (MPS)(lines 4780-4850)

At the start of each iteration, the pump will either be pumping at its capacity A(20) or not, i.e. A(67)=A(20) or 0. If the pump was off but the volume of the pump dam has risen to the switch-on-volume A(42) then the pump will be switched on, i.e. A(67) is set to A(20). If the pump was on but the dam volume has dropped to the switch-off-volume A(43) then the pump is switched off during the current iteration, i.e. A(67) set to zero. The rate of change of the volume of water in the pump dam is then calculated on line 8540 and equals the outflow from the settlers A(66) into the pump dam, less the water being pumped out, A(67).

```
4780 ! ++++++++++++++MPS
4790 IF A(67)>0 THEN 4830
4800 ! PUMPS OFF
4810 IF A(1)>=A(42) THEN A(67)=A
     (20)
4820 GOTO 4850
4830 ! PUMPS ON
4840 IF A(1)<=A(43) THEN A(67)=0
4850 F(1)=A(66)-A(67)
```

Water is pumped from the MPS, approximately 1920m below surface, up to the intermediate pump station (IPS), 960m above.

7.4.1.3 Intermediate Pump Station (IPS) (lines 4860-4930)

This element operates in exactly the same way as the MPS above (7.4.1.2). Water pumped into the pump dam from the MPS below is lifted a further 960m to the sedimentation tank on surface.

```
4860 ! ++++++++++++++IPS
4870 IF A(68)>0 THEN 4910
4880 ! PUMPS OFF
4890 IF A(2)>=A(44) THEN A(68)=A
     (21)
4900 GOTO 4930
4910 ! PUMPS ON
4920 IF A(2)<=A(45) THEN A(68)=0
4930 F(2)=A(67)-A(68)
```

7.4.1.4 Surface Sedimentation Tank (lines 4940-4970)

The head difference of approximately 1920m between the tank and the settlers at the bottom of the mine is used in two stages to displace mud from the settlers to surface. The displacement flow A(80) from the tank is assumed to be a certain fraction A(36) of the variable settler inflow A(66). Outflow from the sedimentation tank A(69) to the hotwell passes over v-notch weirs into a launder in the sedimentation tank. If the volume of water in the tank A(3) is slightly less than the full volume A(46) then outflow to the hotwell A(69) equals 0, otherwise it equals the inflow from IPS, A(68), less the mud displacement flow A(80). The rate of change of water volume in the tank F(3) equals the inflow from IPS less the outflow to the hotwell and for mud displacement.

```
4940 ! +++++++++++SED.TANK
4950 A(80)=A(36)*A(66) ! MUD DIS
     P FLOW
4960 IF A(3)>=A(46) THEN A(69)=M
     AX(0,A(68)-A(80)) ELSE A(69
     )=0
4970 F(3)=A(68)-A(80)-A(69)
```

7.4.1.5 Hotwell (lines 4980-5060)

1) <u>Make-up water</u> is added to the hotwell or the coldwell to replace water losses from the system, whenever the dam volumes drop below certain levels. The coldwell has priority over the hotwell for make-up water because if the coldwell empties then water cannot be supplied to underground. Flags G1 and G2 are set to 1 whenever the coldwell is using make-up from the regional water board (RWB) and supply dam respectively. If the hotwell water volume A(4) is below the switched-on-volumes A(47) or A(48), and flags G1 and G2 equal 0 then make-up water A(74) and/or A(76) will be added to the hotwell at the flowrates A(23) and A(25) respectively.

2) <u>The outflow</u> from the hotwell to the cooling tower A(70) equals the pump capacity A(22) if the hotwell is not empty (A(4) > 0) and the coldwell is not full (A(5) < A(52)). The net inflow rate F(4) equals the inflow from the sedimentation tank A(69) plus the make-up inflow, A(74) and A(76) less flow to the cooling tower A(70). (line 5035)

3) <u>Overflow</u> from the hotwell A(78) will not result if the hotwell is not full (A(4) < A(49) or there is a nett outflow (F(4) < =0). During periods of overflow water is lost from the system and the volume of water in the system is constant, i.e., the nett inflow F(4) equals 0.

```
4980 ! +++++++++++HOTWELL
4990 ! MAKEUP-RWB&SUPPLY DAM
5000 IF A(4)<=A(47) AND G1=0 THE
     N A(74)=A(23) ELSE A(74)=0
5010 IF A(4)<=A(48) AND G2=0 THE
     N A(76)=A(25) ELSE A(76)=0
5020 ! OUTFLOW
5030 IF A(4)>0 AND A(5)<A(52) TH
     EN A(70)=A(22) ELSE A(70)=0
5035 F(4)=A(69)+A(74)+A(76)-A(70
     ) ! NET INFLW
5040 ! OVERFLOW
5050 A(78)=0 ! ASSUME
5060 IF A(4)<A(49) OR F(4)<=0 TH
     EN 5080
```

7.4.1.6 Refrigeration Plant (lines 5070-5090)

When the coldwell is not full and the hotwell not empty, water is pumped from the hotwell to the top of the cooling tower, collected at the bottom of the tower and then pumped through the refrigeration plant to be stored in the coldwell. The flowrate to the coldwell A(70) is less than the hotwell outflow A(70) because a fraction of the inflow is lost due to evaporation A(33) and windage A(34).

```
5070 A(70)=F(4) @ F(4)=0
5080 ! +++++++++REFRIGERATION
5090 A(71)=A(70)*(1-A(33)-A(34))
```

7.4.1.7 Coldwell

1) Make-up water from the regional water board (RWB) (lines 5100-5260) and the supply dam is added to the coldwell if the volume of water in the coldwell A(5) is less than the corresponding switch-on-volumes A(50) and A(51). Make-up water flows A(75) and A(77) either equals 0 or the respective capacity flowrates A(24) and A(26). If make-up water is being added to the hotwell then this flow is first stopped, by setting flags G1 and/or G2 to 0 before make-up is diverted to the coldwell during the next iteration (see lines 5130 and 5170). The rate of inflow to the coldwell F(4) equals the flow from the cooling tower A(71) plus the make-up water flows A(75) and A(77).

2) Outflow from the coldwell A(72) depends on whether the coldwell is empty or not and whether the underground storage dam is full or not. The value of the outflow A(72) is shown in the decision table below and can either be the coldwell inflow rate F(5), the storage dam outflow rate A(65) or the capacity flowrate A(27) from the coldwell to the underground

dam. In practice a sensor monitors the water level of the underground dam and automatically opens or closes a valve at the outlet from the coldwell.

Coldwell	Underground Storage Dam: full	Underground Storage Dam: not full
empty	A(72)=lesser of F(5),A(27)&A(65)	A(72)=lesser of F(5) & A(27)
not empty	A (72)=lesser of A(65)&A(27)	A(72) = A(27)

The net inflow rate to the coldwell F(5) equals the total inflow F(5) less the outflow A(72) (line 5260).

```
5100 ! ++++++++++++COLDWELL
5110 ! MAKEUP-RWB&SUPPLY DAM
5120 IF A(5)>A(50) THEN 5150
5130 IF A(74)=0 THEN A(75)=A(24)
5140 G1=1 @ GOTO 5160
5150 A(75)=0 @ G1=0
5160 IF A(5)>A(51) THEN 5190
5170 IF A(76)=0 THEN A(77)=A(26)
5180 G2=1 @ GOTO 5200
5190 A(77)=0 @ G2=0
5200 F(5)=A(71)+A(75)+A(77) ! TO
     T INFLW
5210 ! OUTFLOW
5220 IF A(5)>0 THEN 5250
5225 Z3=MIN(A(27),A(65))
5230 IF A(6)<A(53) THEN A(72)=MI
     N(F(5),A(27)) ELSE A(72)=MI
     N(F(5),Z3)
5240 GOTO 5260
5250 IF A(6)<A(53) THEN A(72)=A(
     27) ELSE A(72)=MIN(A(65),A(
     27))
5260 F(5)=F(5)-A(72) ! NET INFLW
```

7.4.1.8 Underground Storage Dam (lines 5270,5280)

The rate of change of water volume stored in the dam F(6) equals the inflow from the coldwell A(72) less the demand from the cascade dams A(65). The outflow from the coldwell A(72) is adjusted in 7.4.1.7 above to ensure that the storage dam does not overflow.

7.4.1.9 Cascade Dams (lines 5285, 5286)

The volume of water in the cascade dams is assumed to be constant, that is, the outflow (or demand A(65) - see 7.4.1.1) equals the inflow from the underground storage dam. The rate of change of volume F(7) is zero.

```
5270 ! +++++++++U/G STORAGE DAM
5280 F(6)=A(72)-A(65)
5285 ! +++++++CASCADES
5286 F(7)=0
```

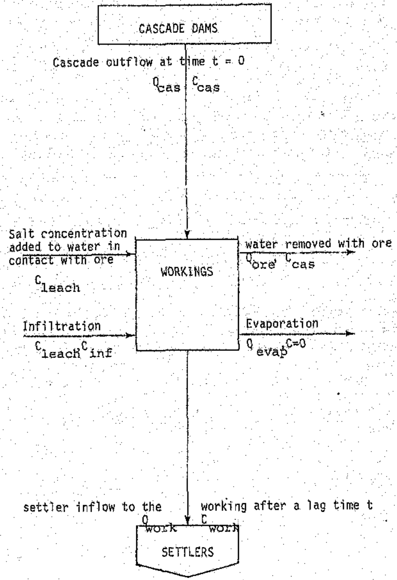
7.4.2 Unsteady Salt Masses in Water Storage Volumes and Unsteady Salt Concentrations of Water Flows (lines 5310-5540)

If the option to use unsteady flows and salts has been selected (flag G=3), then all flows between elements and volumes of water in the elements are calculated at each iteration by the first half of the model described in 7.4.1 above. If only the unsteady salts option was selected then all flows and volumes are constants set to the initial values specified by the user at the start of the simulation. Inflows into storage elements are assumed to have mixed completely during the period of one iteration time-step (normally greater than a few hours).

7.4.2.1 Settler Inflow From The Workings - Salts (lines 5310, 5350).

Consider the diagram below where mine service water leaves the cascade dams to satisfy demand in the workings at each level at time t=0. Some of the water passing through the workings is removed with the ore by evaporation. The remaining water combines with the groundwater infiltration and gravitates down until it arrives at the settlers at time t. The outflow

from the cascade dams and the inflow to the settlers are function of the time of day and the day of the week and are derived from in situ measurements (see 7.4.1.1 above). An equation for the concentration of salts in the settler inflow is derived below.
Q refers to flowrate and C to salt concentration.



Assume

$$Q_{work\atop t=t} = Q_{cas\atop t=0} + Q_{inf} - Q_{ore} - Q_{evap} \qquad (7-1)$$

The user specifies the average travel or lag t t, as well as the independent time relationship Q_{work} and Q_{cas} therefore the equation (7-1) is an assumption.

The leaching of salts from the ore by mine service water is taken into account by assuming that a certain concentration of salts C_{leach} will be added to the existing concentration of Q_{cas} at t=0. The amount added C_{leach} is assumed to be a function of C_{cas} at t=0 and of some saturation concentration ,

$$C_{leach} = X(Y - C_{cas\atop t=0})^{Z} \qquad (7-2)$$

X and Z are constants which can be changed to calibrate a model or be determined from experiments. Assuming that infiltration is not responsible for any significant leaching of salts from the ore, the mass of leached salts entering the mine water per unit time is therefore

$$(Q_{cas} - Q_{ore}) \cdot C_{leach} \qquad (7-3)$$

The total mass of salts entering the settlers per unit time at time t, is the original mass rate of inflow from the cascade to the workings less an amount for ore loss, plus the rates for leaching and infiltration

$$Q_{work}.C_{work} = (Q_{cas} - Q_{ore})(C_{cas} + C_{leach}) + Q_{inf}C_{inf}. \qquad (7-4)$$

substituting from (7-1) to replace Q_{cas} by Q_{work} and from (7-2) and rearranging

$$C_{work} = \frac{(Q_{work} + Q_{evap} - Q_{inf})(C_{cas} + X\left[Y - C_{cas}\right]^{Z}) + Q_{inf}.C_{inf}}{Q_{work}} \qquad (7-5)$$

and appears on line 5350. It should be noted that C_{cas} is the salt concentration of the mine service water in the cascade dam, at a time t (days) earlier in the simulation. C_{cas} is determined on line 5350 by calling the module USE Q which interpolates the value of C_{cas} at the time. The lag or travel time is specified in the input data.

```
5290 ! *********************SALTS
5300 IF G=1 THEN 5540
5310 ! +++++++++++WORKINGS
5320 Z3=A(66)+A(32)-A(37)
5330 A(82)=FNQ(1) ! Ccascade
5340 A(79)=A(60)*(A(61)-A(82))^A
     (62) ! mg/l LEACHED
5350 A(83)=(Z3*(A(79  A(82))+A(3
     7)*A(59))/A(66)   Cworkings
```

7.4.2.2 Settlers - Salts (lines 5360-5370)

Complete mixing of the inflow is assumed and the concentration of salts in the settler A(85) equals the constant settler water volume A(8) divided by the total mass of salts in the settler A(55). The rate of increase of total salt mass in the settler F(8) equals the mass inflow rate A(66)*A(85) ($Q_{work}.C_{work}$ in 7.4.2.1 above), less the mass outflow rate A(66)*A(85). The settlers are designed to contain a fixed volume of water therefore the water volume outflow rate equals the inflow rate A(66).

```
5360 ! ++++++++++++SETTLER
5365 A(85)=A(8)/A(55)
5370 F(8) A(66)*(A(83)-A(85))
```

74.2.3 Main Pump Station Dam (MPS) - Salts (lines 5380-5390)

The average salt concentration of water in the dam A(86) equals the total salt mass A(9) divided by the volume A(1). The rate of increase of total salt mass F(9) equals the salt mass inflow rate from the settlers A(66)*A(85) less the mass outflow rate due to pumping A(67)*A(86)

```
5380 ! ++++++++++++MPS
5385 A(86)=A(9)/A(1)
5390 F(9)=A(66)*A(85)-A(67)*A(86
     )
```

74.2.4 Intermediate Pump Station (IPS) - Salts (lines 5400-5410)

As for the MPS above

```
5400 ! ++++++++++++IPS
5405 A(87)=A(10)/A(2)
5410 F(10)=A(67)*A(96)-A(68)*A(8
     7)
```

7.4.2.5 Sedimentation Tank-Salts (lines 5420-5430)

The average salt concentrationA(88)is the total salt divided by the water volume. The rate of increase of salt mass F(11) equals the salt mass inflow rate from the IPS A(68)*A(87) less the outflow rates to the Hotwell A(69)*A(88) and for mud displacement underground A(80)*A(88).

```
5420 ! +++++++++SED.TANK
5425 A(88)=A(11)/A(3)
5430 F(11)=A(68)*A(87)-(A(69)+A(
     80))*A(88)
```

7.4.2.6 Hotwell-Salts (lines 5440-5450)

The average salt concentration A(89) is calculated as before and the rate of increase of salt mass F(12) equals the mass inflow rates from the sedimentation tank and make-up water from RWB and supply dam, less the mass outflow rates to the refrigeration plant and due to possible overflow.

```
5440 ! +++++++++HOTWELL
5445 A(89)=A(12)/A(4)
5450 F(12)=A(69)*A(88)+A(74)*A(5
     7)+A(76)*A(58)-(A(70)+A(78)
     )*A(89)
```

7.4.2.7 Refrigeration Plant - Salts (lines 5460-5470)

If $Q_{hot} \cdot C_{hot}$ is the salt mass flow rate from the hotwell to the cooling tower and refrigeration plant, then the mass flow rate leaving will be

$$(Q_{hot} - Q_{windage})\, C_{hot}$$

Although evaporation and windage takes place and reduces the water flow rate to

$$Q_{hot} - Q_{windage} - Q_{evap} = Q_{plant},$$

only the windage loss reduces the mass flow rate. The mass flow rate from the refrigeration plant to the coldwell is therefore

$$Q_{plant}\, C_{plant} = (Q_{hot} - Q_{windage})\, C_{hot}$$

and $$C_{plant} = Q_{hot} C_{hot} \left(1 - \frac{Q_{windage}}{Q_{hot}}\right) / Q_{plant} \qquad (7\text{-}6)$$

Equation (7-6) appears on line 5470 below. The ratio of $Q_{windage}/Q_{hot}$ or A(34) in the program is a parameter specified by the user.

```
5460 ! +++++REFRIGERATION
5470 IF A(71)=0 THEN A(84)=0 ELS
     E A(84)=A(89)*A(70)*(1-A(34
     ))/A(71)
```

7.4.2.8 Coldwell - Salts (lines 5480-5490)

The average salt concentration A(90) is calculated as before and the rate of increase of salt mass F(13) equals the salt mass inflow rates from the refrigeration plant and make-up water from the RWB and supply dam , less the mass outflow rate of water to underground.

```
5480 ! +++++++++COLDWELL
5485 A(90)=A(13)/A(5)
5490 F(13)=A(71)*A(84)+A(75)*A(5
     7)+A(77)*A(58)-A(72)*A(90)
```

7.4.2.9 Underground Storage Dam - Salts (lines 5500-5510)

The average salt concentration A(91) is calculated as before, and the rate of increase of salt mass F(14) equals the inflow rate of water from the coldwell, less the outflow rate to the cascade dams.

```
5500 ! ++++++++U/G STORAGE DAM
5505 A(91)=A(14)/A(6)
5510 F(14)=A(72)*A(90)-A(65)*A(9
     1)
```

7.4.2.10 Cascade Dams - Salts (lines 5520-5530)

The cascade dams,one on each working level,are assumed to be grouped together to form one large dam of constant volume A(7). This has the effect of assuming that the salt concentration is the same in all the dams and that any inflow into the top dam, which has a different concentration, will be completely mixed in all the dams after one time increment (normally in excess of a few hours). Being of constant volume, the assumption is also made that the inflow equals the outflow. It is felt that the assumption of one dam is reasonable as the large volume and combined retention time of more than 12 hours of the coldwell and storage dam would ensure that only gradual changes would occur in the salt concentration of inflow to the cascade dams. It is felt that the assumption of a constant dam volume is also reasonable as the changes in storage volume of each of the 0,15Mℓ capacity dams occurs over periods much shorter than the simulation time steps used. A simulation of a detailed cascade model could be carried out as a separate study if required. The simple representation of the cascade system in this model should be adequate for studying the mine water system as a whole.

The average salt concentration of the cascade water A(92) equals the total salt mass stored A(15) divided by the constant volume A(7). The rate of increase of total salt mass stored equals the mass inflow rate A(65)xA(91) less the outflow rate to the workings A(65)xA(92).

```
5520 ! ++++++++CASCADE DAMS
5525 A(92)=A(15)/A(7)
5530 F(15)=A(65)*(A(91)-A(92))
5540 ! *********END OF SALTS
```

7.4.3 Cumulative Flow Volumes and Salt Masses (lines 5550-5660)

It is often useful to know, for example, what the total mass of salts leached or the total volume of make-up water draw in from a source is, since the start of the simulation. This value can be compared with recorded values for verification and calibration or used to determine important unrecorded values such as the overflow from the hotwell.

Depending on the solution algorithm used, the model is called a number of times during a single iteration time step (once for the Euler method, twice for Modified Euler and four times for the Fourth-Order Runge-Kutta). Each time the model is called, all the variables are calculated at the time T of the simulation, and then the derivatives of each of the 15 differential equations are calculated at the end of the model (lines 7960-7990). The simulation program then uses a weighted average of the derivate values in the simultaneous solution of the equations. In order to obtain the correct cumulative values for the water volume or salt mass, the same weighted average (as used in the algorithm) of the respective water or salt mass flowrates must be integrated over the time interval.

In this model, the water and salt-mass flowrates A(74) to A(80) are integrated to obtain the corresponding cumulative values A(95) to A(101). For example

$$A(100) = \int_{TO}^{T} A(79)\,dt$$

where TO is the simulation start time and T is the current time in the simulation.

Up to four values of each of the variables A(74) to A(8) are stored in matrix Z5 for each iteration (lines 5560 to 5580). The modified Euler and the Runge-Kutta modules set the time T to T1+H (iteration start time plus the iteration time step), for the last address of the model. In other words all the variable values are available for calculating the cumulative values when T=T1+H (see line 5590). The Euler method is the exception, only one variable value per differential equation is available for the iteration, therefore the cumulative value is calculated each time MODEL is called. A check is therefore made on line 5590 to ensure integration if the Euler method is used, i.e. when flag F2= 1.

Depending on the algorithm used (flag F2=1,2, or 3 for the Euler, Modified Euler and Runge-Kutta), the cumulative value is calculated on lines 5630 to 5650. The cumulative value of a variable at the end of the iteration equals the value at the start of the iteration plus the weighted average of the variable values stored in Z5, multiplied by the iteration time step H.

```
5550 ! ++INTEGRATE A(74)-A(80)
5560 FOR J3=74 TO 80
5570 Z5(J3-73,Z)=A(J3)
5580 NEXT J3
5590 IF F2#1 AND T<T1+H THEN 567
     0
5600 FOR J3=95 TO 101
5610 Z3=J3-94
5620 ON F2 GOTO 5630,5640,5650
5630 A(J3)=A(J3)+H*Z5(Z3,1) @ GO
     TO 5660
5640 A(J3)=A(J3)+H*(Z5(Z3,1)+Z5(
     Z3,2))/2 @ GOTO 5660
5650 A(J3)=A(J3)+H*(Z5(Z3,1)+2*(
     Z5(Z3,2)+Z5(Z3,3))+Z5(Z3,4)
     )/6
5660 NEXT J3
```

7.4.4 Total Water Volume and Salt Mass in the System at the Current Simulation Time (lines 5662-5667)

It is useful to see what the total water volume and total salt mass stored in the eight storage elements of the system is. At the end of each iteration, the total water volume A(93) and the total salt mass A(81) is calculated from the variables A(1) to A(15).

```
5662 ! ++++TOT.VOLUME&MASS
5663 A(93),A(81)=0
5664 FOR J3=1 TO 7
5665 A(93)=A(93)+A(J3) @ NEXT J3
5666 FOR J3=8 TO 15
5667 A(81)=A(81)+A(J3) @ NEXT J3
5670 ! ***********END OF MODEL
```

7.5 DISCUSSION OF THE MODULE - INITIAL (lines 8000-8999)

This module is called by the simulation program immediately before simulation begins and any instructions specified by the user are executed.

```
8000 ! ------------INITIAL
8010 ! SUBR DONE BEFORE SIMUL'N
8020 DISP "1)UNSTEADY FLOWS&NO S
     ALTS"
8021 DISP "2)UNSTEADY FLOWS&UNST
     EADYSALTS  (1or2)";@ INPUT
     G
8025 IF G#1 AND G#2 THEN 8020
8026 G1,G2=0 ! NO COLD-W MAKEUP
8030 DISP USING 8040 ; "UNISEL G
     OLD MINE","Detailed unstead
     y flow and","unsteady salt
     model."
8040 IMAGE /,3(32A),32("=")
8050 DISP "UNITS:Ml,d,kg.(NOTE:k
     g/Ml=mg/l)" .
8060 DISP @ DISP "CAPACITY FLOW
     RATES(Ml/d):"
8070 DISP " PUMPS:MPS,IPS,HOT-CO
     LDWELL    ";A(20);A(21);A(2
     2)

8080 DISP " MAKEUP WATER TO HOT&
     COLDWELLS:   RWB";A(23);A(2
     4)
8090 DISP "   SUP.DAM";A(25);A(26
     )
8100 DISP " COLDWELL TO INTERMED
     .DAM";A(27)
8110 DISP @ DISP "AVERAGE FLOWS
                 (Ml/d)"
8120 DISP USING 8130 ; " INFILTR
     ATION",A(37)," EVAPORATION"
     ,A(32)
8130 IMAGE 24A,X,K,/
8135 DISP TAB(26);"RATIOS"
8140 DISP USING 8130 ; " TOWER E
     VAP/INFLOW",A(33)," TOWER W
     INDAGE/INFLOW",A(34)
8150 DISP USING 8130 ; " ORELOSS
     /CASCADE FLOW",A(35)," MUD/
     SETTLER INFLOW",A(36)
```

```
8160 ! +++++++HYDROGRAPH FACTORS
8170 B1=A(38)/A(39)
8180 Z3=A(38)+7*(A(37)-A(32))-A(
     35)*A(38) ! WEEKLY SETTLER
     INFLW
8190 B2=Z3/A(40)
8195 DISP TAB(23);"(Ml/week)"
8200 DISP USING 8130 ; " MINE IN
     FLOW",A(38)," SETTLER INFLO
     W",Z3
8210 DISP @ DISP "VOLUMES(Ml):"
8220 DISP " START-STOP VOLUMES:"
8230 DISP "  MPS PUMP";;;A(42);A
     (43) @ DISP "  IPS PUMP";A(
     44);A(45)
8240 DISP " SWITCH ON VOLUMES FO
     R MAKEUP:"
8250 DISP "  HOTWELL-RWB&SUP.DAM
     ";A(47);A(48) @ DISP "  COL
     DWELL-RWB&SUP.DAM";A(50);A(
     51)
8260 DISP " FULL CAPACITY:"
8270 DISP "  SED.TANK,HOTWELL,CO
     LDWELL";A(46);A(49);A(52)
8280 DISP "  INTERMED&CASCADE DA
     MS,SETTLERS";A(53);A(54);A(
     55)
8290 IF G=1 THEN GOTO 8800
8300 DISP @ DISP "SALT CONCENTRA
     TION(kg/Ml=mg/l):"
8310 DISP " MAKEUP-RWB&SUP.DAM";
     A(57);A(58)
8320 DISP " INFILTRATION";A(59)
8330 DISP " LEACHING=X(Y-Ccascad
     e)^Z" @ DISP " X,Y,Z";A(60)
     ;A(61);A(62)
8800 ! DATA FOR 1)INIT VALS
8802 ! 1TO7
8804 DATA 1.3,.45,1.6,4,2,2,1.5
8806 ! 8TO19



8808 DATA 1260,910,315,1120,2800
     ,1400,1400,1050,0,0,0,0
8810 ! 20TO29
8812 DATA 14.7,14.7,11.5,.5,.5,5
     .4,5.4,26,0,0
8814 ! 30TO40
8816 DATA 0,0,.4,.05,.01,.2,.08,
     .5,70,2.619,1.478
8818 ! 41TO56
8820 DATA 0,2.9,1.6,1,.4,1.6,1.6
     ,1.6,4,.8,.8,2,2,1.5,1.8,0
8822 ! 57TO64
8824 DATA 700,12000,4000,.1,1600
     0,1,0,0
8826 ! 65TO73
8828 DATA 2,.1,0,0,0,0,0,2,0
8830 ! 74TO81
8832 DATA 0,0,0,0,0,1530,0,10250
```

```
8834 ! 82T084
8836 DATA 700,700,700
8838 ! 85T094
8840 DATA 700,700,700,700,700,70
     0,700,700,0,0
8842 ! 95T0101
8844 DATA 0,0,0,0,0,0,0
8850 ! DATA FOR 4)PRT LIST
8852 DATA 13,1,2,3,4,5,6,7,95,96
     ,97,98,99,101
8870 ! DATA FOR 6)PLOT INPT
8872 DATA 12,4000
8876 DATA 1
8878 DATA 1,0.4,1,2,0,1.5,1,3,1.
     5,1.7,.1,4,0,4.5,1,5,0,2.5,
     1,66,0,20,5
8880 DATA 67,0,20,5,69,0,20,5,70
     ,0,20,5,72,0,20,5,97,0,1,1,
     98,0,1,1
8900 ! DATA FOR 10)TABLES
8902 DATA 4
8904 DATA 11,6,2,4,11,5,2,6
8906 DATA 0,1,2,3,4,7,8,9,10,11,
     14
8908 DATA 5,12
8910 DATA 0,1,2,3,4,7,8,9,10,11,
     14
8912 DATA 5,12
8914 DATA 0,.2,.25,.2,.5,.8,.75,
     .5,.875,.7,1,.2
8916 DATA 0,.2,.25,.2,.5,.8,.75,
     0
8918 DATA 0,.01,.25,.01,.5,.5,.7
     5,.4,1,.01
8920 DATA 0,.01,.25,.01,.5,.5,.7
     5,.4,1,.01,2,.01
8970 ! DATA FOR 11)QUEUES
8972 DATA 1,92,.02,5,700
8999 RETURN
```

In this particular example, INITIAL is used to carry out the following:

1) Ask the user what type of simulation is required; unsteady flows and no salts or unsteady flows and unsteady salts? A flag G is set to 1 or 2 and is used in MODEL during simulation. See lines 8010 to 8025. If G equals 1 then the salt information is irrelevant (line 8290).

2) Set flags G1 and G2 to zero which means that no make-up water is being added when the simulation is started (line 8026).

3) Display the name of the model (line 8030)

4) Display the name and value of all the parameters used in the simulation (lines 8050 to 8330)

5) Use parameters to calculate other parameters The correction factors B1 and B2, for converting the graphs of relative cascade outflows and settler inflows to actual flows are calculated on lines 8160 to 8190. The graphs and detailed explanation are given earlier in section 7.4.1.1. B1 and B2 are used in MODEL.

6) Entry of large amounts of input data. Certain lines in INITIAL are reserved for data statements from which the input data can be read. During input the user has the option of entering the data through the keyboard or asking the program to read it from data statements, or a mixture of both. The following lines contain the data values in the order described:

a) Lines 8800 to 8844: The initial values of variables A(1)toA(101)

b) Line 8850 to 8852: The values of 13 variables (1, 2, ------ 99,101) must be displayed as output. The frequency and simulation time interval are entered through the keyboard when the program is running.

c) Lines 8870 to 8880. Data for plotting graphs as follows: (8872) 12 graphs and file space for 2000 points on tape file; (8876) one day markings on the time axis or abscissa; (8878,8880) variable number, minimum and maximum ordinate values - repeated for each graph.

d) Lines 8900 to 8920; Table data for time dependent relationships which can best be entered as tables. (8902); a total of four tables are used for cascade outflow on weekdays and weekends and for settler inflow on weekdays and weekends. (8904); table 1 for weekday cascade outflow is used 11 times for a two week period starting on a Monday 0h00 and ending on a Monday 24h00. There are six data pairs in table 1 (see graph 7.1 in section 7.4.1.1). Table 2 for weekend cascade outflow is used 2 times in the 2 week period and contains 3 data pairs. Similarly for table 3 and 4. (8906-8912); the simulation time-step is in units of 1 day and the simulation starting time (T=0) is Monday 0h00. The starting times T, for table 1 are therefore 0,1,2,3,4,7, etc., and the starting time for the weekend cascade outflow table 2 is 5 (first Saturday 0h00) and 12. Similarly for tables 3 and 4. (8914-8920); the data pairs of each table in turn, that is, the time increment after the start times given above and the corresponding relative flow value from the relevant graph (see graphs 7.1 to 7.4 in section 7.4.1.1)

e) Lines 8970 and 8972; Queues are used to store the value of certain variables over a certain period of time. (8972); 1 queue is used to store up to 5 past values of variable 92 (the salt concentration of cascade outflow) so that values can be interpolated 0,02 days earlier in the simulation. At the start of the simulation all the queue values are set to 700 kg/Mℓ. The lag time of 0,02 days or 50 minutes is the average travel time assumed for water leaving the cascade dams and gravitating down to the settlers.

Five groups of data a) to e) have been described above. If, during input, the user elects to read predefined data from data statements rather than enter it through the keyboard then a pointer will be set to the first data statement following the comment lines at the start of the relevant data group. If, for example, the relevant data group is table data, then the pointer is set to the first data statement following the comment statement on line 8900, which is on line 8902. There the value 4 is the first value that will be read. The five comment lines at the start of each data group, on lines 8800, 8850, 8870, 8900 and 8970 are reserved for directing the pointer. The data can be entered on any line number below the relevant comment time using any number of data statements so long as the data values appear in the correct order and they are above the next reserved comment line.

The advantage of using prodefined input data in data statements is that values can quickly be re-entered whenever the values are destroyed. All the variable values become undefined whenever execution is terminated as a result of a programming error, large numbers from instability, division by zero etc.

7.6 DISCUSSION OF THE MODULE - FINAL (lines 9000-9999)

This module is called by the simulation program immediately after the simulation ends and any user specified instructions in the module will be executed. In this particular example, FINAL is used to print out the step size and algorithm used, and the cumulative values of variables A(74) to A(80) (see section 7.4.3).

```
9000 ! ------------FINAL
9010 ! SUBR TO PERFORM ANY USER-
     DEFINED OPERATIONS AFTER A
     RUN
9020 DISP "TIME STEP";H;"DAYS"
9025 DISP "START&END TIME:";T0;T
     9
9029 DISP "ALGORITHM:";
9030 IF F2=1 THEN DISP "EULER"
9032 IF F2=2 THEN DISP "MOD.EULE
     R"
9034 IF F2=3 THEN DISP "4th ORDE
     R R-KUTTA"
9040 DISP @ DISP "CUMULATIVE OUT
     PUT(Ml)"
9050 DISP " MAKE UP WATER TO HOT
     &COLDWELLS"
9060 DISP "  RWB     ";A(95);A(96
     ) @ DISP "  SUP.DAM";A(97);
     A(98)
9070 DISP " HOTWELL OVERFLOW";A(
     99)
9080 DISP " MUD REMOVED";A(101)
9090 DISP " SALTS LEACHED(kg)";A
     (100)
9999 RETURN
```

CHAPTER EIGHT

8. RUNNING THE DETAILED UNISEL MODEL - MODEL OPERATION AND RESULTS

The steps involved in running the program are summarised in appendix E. After the RUN key is pressed, the heading and notes are displayed.

```
CONTINUOUS SIMULATION OF A
SYSTEM DESCRIBED BY ORDINARY
DIFFERENTIAL EQUATIONS WITH
INITIAL BOUNDARY CONDITIONS

N.C.HOLTON-WITS UNIVERSITY.
********************************
NOTE
1)Press key k1 WHILE EXECUTING
  to get back into input mode
2)Space is provided inF&K for up
  to 20 differential equations
3)Space is provided      A0&B
  for up to 105 var      s in the
  prgm
4)Space is provided in A1 for
  pr  ting up to 50 variables at
  each step
5)File space provided for<=4000
  graph points M has sp   for
  <=20graphs.
6)Space is provided in D1 for
  <=10 tables and in D2&D3 for a
  total of<=100start times and
  <=100data points
7)Space is  rovided in E1&E2for
  <=20queu   and a total of 100
  values
********************************
```

Fig. 8.1

8.1 Run No. 1: Water Quantity Simulation (no salts)

Although the complete model contains 15 differential equations, the first run only deals with water quantity aspects and therefore involves only the first 7 differential equations which describe water volume changes in components (fig. 8.2. typed in).

There are 101 program variables in the model and although many of them deal with water quality aspects, they are still required to be initialised in the program otherwise they will be undefined when encountered in the model for the first time. Water quality variables will not be affected during simulation because the salt section of the model will be skipped (fig 8.2. typed in).

```
DIFFERENTIAL EQUATIONS&VARIABLES
dA(J)/dT=F(J) J=1 to?
7

F(J)=func(T,A(1),.A( 7 ),.A(N1))
EXTRA VARIABLES ARE A( 8),.A(N1)
(IF NO EXTRAS,ENTER N1= 7) N1=?
?
101
```

Fig. 8.2

The initial values are listed in Fig. 8.3 (Predefined data)

```
A0( 1 ) 1.3
A0( 2 ) .45
A0( 3 ) 1.6
A0( 4 ) 4
A0( 5 ) 2
A0( 6 ) 2
A0( 7 ) 1.5
A0( 8 ) 1260
A0( 9 ) 910
A0( 10 ) 315
A0( 11 ) 1120
A0( 12 ) 2800
A0( 13 ) 1400
A0( 14 ) 1400
A0( 15 ) 1050
```

Fig. 8.3

```
A0( 16 ) 0
A0( 17 ) 0
A0( 18 ) 0
A0( 19 ) 0
A0( 20 ) 14.7
A0( 21 ) 14.7
A0( 22 ) 11.5
A0( 23 ) .5
A0( 24 ) .5
A0( 25 ) 5.4
A0( 26 ) 5.4
A0( 27 ) 26
A0( 28 ) 0
A0( 29 ) 0
A0( 30 ) 0
A0( 31 ) 0
A0( 32 ) .4
A0( 33 ) .05
A0( 34 ) .01
A0( 35 ) .2
A0( 36 ) .08
A0( 37 ) .5
A0( 38 ) 70
A0( 39 ) 2.619
A0( 40 ) 1.478
A0( 41 ) 0
A0( 42 ) 2.9
A0( 43 ) 1.6
A0( 44 ) 1
A0( 45 ) .4
A0( 46 ) 1.6
A0( 47 ) 1.6
A0( 48 ) 1.6
A0( 49 ) 4
A0( 50 ) .8
A0( 51 ) .8
A0( 52 ) 2
A0( 53 ) 2
A0( 54 ) 1.5
A0( 55 ) 1.3
A0( 56 ) 0
A0( 57 ) 700
A0( 58 ) 12000
A0( 59 ) 4000
A0( 60 ) .1
A0( 61 ) 16000
A0( 62 ) 1
A0( 63 ) 0
A0( 64 ) 0
A0( 65 ) 2
A0( 66 ) .1
A0( 67 ) 0
A0( 68 ) 0
A0( 69 ) 0
A0( 70 ) 0
A0( 71 ) 0
A0( 72 ) 2
```

Fig. 8.3 continued

```
A0( 73 ) 0
A0( 74 ) 0
A0( 75 ) 0
A0( 76 ) 0
A0( 77 ) 0
A0( 78 ) 0
A0( 79 ) 1530
A0( 80 ) 0
A0( 81 ) 10250
A0( 82 ) 700
A0( 83 ) 700
A0( 84 ) 700
A0( 85 ) 700
A0( 86 ) 700
A0( 87 ) 700
A0( 88 ) 700
A0( 89 ) 700
A0( 90 ) 700
A0( 91 ) 700
A0( 92 ) 700
A0( 93 ) 0
A0( 94 ) 0
A0( 95 ) 0
A0( 96 ) 0
A0( 97 ) 0
A0( 98 ) 0
A0( 99 ) 0
A0( 100 ) 0
A0( 101 ) 0
```

Fig. 8.3 continued

The value of thirteen variables are specified for printed output,(fig 8.4. pre-defined data). The variables are as follows, (see section 7.3.1 for complete list):

Description

Number	
1	Volumes: MPS
2	IPS
3	sedimentation tank
4	hotwell
5	coldwell
6	underground storage dam
7	cascade dams
95	cumulative make-up from RWB to hotwell
96	cumulative make-up from RWB to coldwell
97	cumulative make-up from supply dam to hotwell
98	cumulative make-up from supply dam to coldwell
99	cumulative overflow from hotwell
101	cumulative mud pumped from the mine.

```
HOW MANY VARIABLES
A(1)..A(101)MUST BE PRINTED
 13
ENTER THE 13 NUMBER(S) BETWEEN
1 AND 101 IN ANY ORDER.
(NO. 1 ) 1
(NO. 2 ) 2
(NO. 3 ) 3
(NO. 4 ) 4
(NO. 5 ) 5
(NO. 6 ) 6
(NO. 7 ) 7
(NO. 8 ) 95
(NO. 9 ) 96
(NO. 10 ) 97
(NO. 11 ) 98
(NO. 12 ) 99
(NO. 13 ) 101
```

Fig 8.4

The variables shown in Fig. 8.4 must be printed out from day 4 in the simulation to day 8. The variable values must be printed every 12 iteration time-steps (fig. 8.5. typed in)

```
PRINT OUTPUT FROM T=T2 TO T8
EVERY S0 TIME STEPS
(T2,T8,S0)?
4,8,12
```

Fig. 8.5

Twelve graphs are to be plotted. A file exists for storing up to 2000 coordinates for plotting at the end of the simulation. Tic-marks at 1-day spacing are required on the time axis. For each graph four values are entered; the variable number, the minimum and maximum values on the variable axis and the spacing of the tic-marks on the variable axis. (fig. 8.6. Predefined data).

```
NO.PLOTS,MAX NO.PTS.(See note 5)
 12  4000
UNIT SPACING ALONG T-AXIS 1
FOR EACH VARIABLE TO BE PLOTTED
ENTER:
V-VARIABLE NO.,MN-MIN.VALUE,MX-
MAX.VALUE,U-UNIT SPACING ALONG
VARIABLE AXIS
(V,MN,MX,U) 1  0  4  1
(V,MN,MX,U) 2  0  1.5  1
(V,MN,MX,U) 3  1.5  1.7  .1
(V,MN,MX,U) 4  0  4.5  1
(V,MN,MX,U) 5  0  2.5  1
(V,MN,MX,U) 66  0  20  5
(V,MN,MX,U) 67  0  20  5
(V,MN,MX,U) 69  0  20  5
(V,MN,MX,U) 70  0  20  5
(V,MN,MX,U) 72  0  20  5
(V,MN,MX,U) 97  0  1  1
(V,MN,MX,U) 98  0  1  1
```

Fig. 8.6

All the variables specified in fig. 8.6 must be plotted from day 4 to day 8 and about 100 points must be plotted for each graph (fig. 8.7 typed in).

```
PLOT SPECIFIED VARIABLES FROM
T=T3 TO T4,USING ABOUT C8 PTS.
(T3,T4,C8)?
4,8,100
```

Fig. 8.7

Execution must be from day 0 to day 8 and the time step increment must be 0,083' days or 2 hours (fig. 8.8 typed in). It should be noted that there is no output from day 0 to day 4 as this is considered a 4 day (4 cycles) 'warm-up' period to eliminate errors in the initial values (see dicussion in chapter 5).

```
EXECUTE FROM T=T0 TO T9 WITH TIM
E STEP H
(T0,T9,H)?
0,8, .08333333333
```

Fig. 8.8

The Fourth-Order Runge-Kutta algorithm is selected to solve the 7 differential equations simultaneously at each iteration of the simulation (fig. 8.9 typed in).

```
EULER,MOD.EULER OR 4TH ORDER
RUNGE-KUTTA(1,2or3)?
3
```

Fig. 8.9

Four external variables define the demand from the cascade dams and the settler inflow on weekdays and weekends. The relationships are stored as tables of data pairs. The data pairs and start times for each table are entered (fig. 8.10 predefined data. See section 8.4.1.1 and appendix E for a more detailed explanation.

Only one queue is required to store earlier values of variables A(92) in the simulation. Only values which are up to about 0,02 days old (about 1 hour) must be stored. Five values are stored in the queue. Since the simulation time step is 6 hours, only the first two values in the queue will be used to interpolate the value of A(92) at the lag time of 0,02 days. The extra three values in the queue would only be required if a much smaller time-step were specified. The execution is halted so that the user can note the allocation and make a note to change it later, if required. Changes cannot be made at this stage. All five values in the queue are initialised to a value of 700 mg/ℓ (fig.F11 predefined data).

```
HOW MANY QUEUES(See note 7) 1
FOR EACH QUEUE ENTER VARIABLE NO
,LAG TIME
Q NO. 1 (A(?),LAG) 92  .02
TOTAL SPACES FOR QUEUE VALUES
(See note 7) 5
SPACE FOR QUEUES 1 TO 1 :
 5 TOTAL= 5 .(see note 7)
IF ALLOCATIONS UNSATISFACTORY
CORRECT LATER.(press CONT key)
INITIALIZE EACH QUEUE TO 1 VALUE
Q NO  1 A( 92 ) 700
```

Fig. 8.11

```
HOW MANY TABLES(see note 5)
 4
FOR EACH TABLE ENTER 2 VALUES:
NO.OF TIMES TABLE WILL BE USED,
NO. OF DATA PAIRS IN TABLE(See
note 6)
(TABLE 1 ) 11  6
(TABLE 2 ) 2  4
(TABLE 3 ) 11  5
(TABLE 4 ) 2  6
TABLE 1
 1 (Tstart) 0
 2 (Tstart) 1
 3 (Tstart) 2
 4 (Tstart) 3
 5 (Tstart) 4
 6 (Tstart) 7
 7 (Tstart) 8
 8 (Tstart) 9
 9 (Tstart) 10
 10 (Tstart) 11
     start) 14

      rt) 5
     tart) 12

 1 (Tstart) 0
 2 (Tstart) 1
 3 (Tstart) 2
 4 (Tstart) 3
 5 (Tstart) 4
 6 (Tstart) 7
 7 (Tstart) 8
 8 (Tstart) 9
 9 (Tstart) 10
 10 (Tstart) 11
 11 (Tstart) 14
TABLE
 1 (Tstart) 5
 2 (Tstart) 12
ENTER T-INCREMENT AFTER TABLE
START VALUE,TABLE VALUE
TABLE 1
 1 (T-inc,TAB) 0       .2
 2 (T-inc,TAB) .25     .2
 3 (T-inc,TAB) .5      .8
 4 (T-inc,TAB) .75     .5
 5 (T-inc,TAB) .875    .7
 6 (T-inc,TAB) 1       .2
TABLE 2
 1 (T-inc,TAB) 0       .2
 2 (T-inc,TAB) .25     .2
 3 (T-inc,TAB) .5      .8
 4 (T-inc,TAB) .75     0
TABLE 3
 1 (T-inc,TAB) 0       .01
 2 (T-inc,TAB) .25     .01
 3 (T-inc,TAB) .5      .5
 4 (T-inc,TAB) 75      .4
 5 (T-inc,TAB) 1       .01
TABLE 4
 1 (T-inc,TAB) 0       .01
 2 (T-inc,TAB) .25     .01
 3 (T-inc,TAB) .5      .5
 4 (T-inc,TAB) .75     .4
 5 (T-inc,TAB) 1       01
 6 (T-inc,TAB) 2       .01
```

<u>Fig. 8.10</u>

After the above input data has been entered, 12 options are given (fig. 8.12). Since no mistakes have been made and no last minute changes are required, option 12 is selected to start the run. Only option 2 is not applicable because all variables are set to the initial values at this stage in any case.

```
1)CHANGE ALL INIT. VALUES
2)RE-USE INIT.VARIABLES&QUEUES
3)CHANGE SOME VARIABLE VALUES
4)CHANGE PRINT LIST
5)CHANGE PRINT TIMES
6)CHANGE PLOTTING DATA
7)CHANGE TIMES FOR PLOTTING
8)CHANGE EXECUTION TIMES
9)CHANGE ALGORITHM
10)CHANGE FUNCTION TABLES
11)CHANGE QUEUE PARAMETERS
12)EXECUTE PROGRAM
(LAST OPTION 12 )
ENTER NEW OPTION 1TO 12?
12
OUTPUT OF RESULTS-
Print or Display(PorD)?
P
```

Fig. 8.12

The module INITIAL is executed immediately before simulation begins (fig. 8.13). As only 7 differential equations were specified in fig. 8.2 above, option 1 must be selected as shown, otherwise the differential equations 8 to 15 will be encountered and execution will stop with an error message. A heading is printed and important initial values are listed with descriptions. Notice that the average settler inflow has been calculated and displayed although it is not specified in the input data.

During simulation the specified variables are printed out every 12 iteration steps of 2 hours, i.e., once a day, from day 4 (Thursday 24h00 or Friday 00h00) to day 8 (end of Mond y 24h00). The sequence shows

```
1)UNSTEADY FLOWS&NO SALTS
2)UNSTEADY FLOWS&UNSTEADYSALTS
(1or2)?
1

UNISEL GOLD MINE
Detailed unsteady flow and
unsteady salt model.
===============================
UNITS:Ml,d,kg.(NOTE:kg/Ml=mg/l)

CAPACITY FLOW RATES(Ml/d):
 PUMPS:MPS,IPS,HOT-COLDWELL
 14.7  14.7  11.5
 MAKEUP WATER TO HOT&COLDWELLS:
  RWB .5  .5
  SUP.DAM 5.4  5.4
 COLDWELL TO INTERMED.DAM 26

AVERAGE FLOWS               (Ml/d)
 INFILTRATION               .5
 EVAPORATION                .4
                            RATIOS
 TOWER EVAP/INFLOW          .05
 TOWER WINDAGE/INFLOW       .01
 ORELOSS/CASCADE FLOW       .2
 MUD/SETTLER INFLOW         .08
                           (Ml/week)
 MINE INFLOW                70
 SETTLER INFLOW             56.7

VOLUMES(Ml):
 START-STOP VOLUMES:
  MPS PUMP 2.9  1.6
  IPS PUMP 1  .4
 SWITCH ON VOLUMES FOR MAKEUP:
  HOTWELL-RWB&SUP.DAM 1.6  1.6
  COLDWELL-RWB&SUP.DAM .8  .8
 FULL CAPACITY:
  SED.TANK,HOTWELL,COLDWELL 1.6
 4  2
  INTERMED&CASCADE DAMS,SETTLERS
 2  1.5  1.8
```

Fig. 8.13

what takes place over a typical weekday Friday, the weekend, and Monday. Variables A(1) to A(7) are the various dam volumes. It can be seen that the volumes at the end of Monday (T=8) are almost the same as those at the end of Friday (T=4); both are typical weekdays. This can also be seen on the graphs and shown later indicating that the system returns to its steady weekday cycle within 24 hours, even after a considerable disturbance such as the weekend. This statement applies to water quantity and not to water quality which is examined later.

```
***************************  ****
INITIAL VALUE(S);T= 0
A( 1 )= 1.3
A( 2 )= .45
A( 3 )= 1.6
A( 4 )= 4
A( 5 )= 2
A( 6 )= 2
A( 7 )= 1.5
A( 95 )= 0
A( 96 )= 0
A( 97 )= 0
A( 98 )= 0
A( 99 )= 0
A( 101 )= 0
T= 4
A( 1 )= 2.08947338745
A( 2 )= .858333333334
A( 3 )= 1.98230829952
A( 4 )= 2.12197827397
A( 5 )= .936939212076
A( 6 )= 2
A( 7 )= 1.5
A( 95 )= .71527777778
A( 96 )= .888888888894
A( 97 )= 7.725
A( 98 )= 9.6
A( 99 )= 0
A( 101 )= 2.82349120433
T= 4.99999999999
A( 1 )= 1.92955006764
A( 2 )= .858333333335
A( 3 )= 1.87290369872
A( 4 )= 2.26162118483
A( 5 )= 1.06027123726
A( 6 )= 2
A( 7 )= 1.5
A( 95 )= .909722222225
A( 96 )= 1.12500000001
A( 97 )= 9.825
A( 98 )= 12.15
A( 99 )= 0
A( 101 )= 3.52936400542
T= 5.99999999999
A( 1 )= 1.76962674783
A( 2 )= .858333333336
A( 3 )= 1.76349909792
A( 4 )= 4.03566190054
A( 5 )= 2.24110982069
A( 6 )= 2
A( 7 )= 1.5
A( 95 )= 1.11805555557
A( 96 )= 1.17361111112
A( 97 )= 12.075
A( 98 )= 12.675
A( 99 )= .605679972937
A( 101 )= 4.23523680651
```

Fig.8.14

```
T= 6.99999999999
A( 1 )= 1.94908660348
A( 2 )= .450000000003
A( 3 )= 1.7340877312
A( 4 )= 4.03586190054
A( 5 )= 2.24110982069
A( 6 )= 2
A( 7 )= 1.5
A( 95 )= 1.11805555557
A( 96 )= 1.17361111112
A( 97 )= 12.075
A( 98 )= 12.675
A( 99 )= 1.21690121786
A( 101 )= 4.26592692831
T= 7.99999999999
A( 1 )= 1.99332995034
A( 2 )= .858333333337
A( 3 )= 1.99607465047
A( 4 )= 2.74855773576
A( 5 )= 1.08138629031
A( 6 )= 2
A( 7 )= 1.5
A( 95 )= 1.16666666667
A( 96 )= 1.4027777778
A( 97 )= 12.6
A( 98 )= 15.15
A( 99 )= 1.21690121786
A( 101 )= 4.9717997294
********************************
```

Fig.8.14
Continued

All 12 graphs below are plotted for 4 days from the start of Friday (0h00) to the end of the first Monday (24h00). This is the same period as was selected for the printout. The variable name is written on the vertical axis and the unmarked horizontal axis is the time axis.

Fig. 8.15 is a plot of the main pump station dam volume over the 4 day period. The information printed to the right of the graph tells us that the minimum and maximum values on the abscissa are 4 and 8 (days) and the tic-marks are 1 day apart. The minimum and maximum values on the ordinate (not the graph curve) are 0 and 4 (megalites) and the tic-marks are 1 unit (megalitres) apart.

The level rise corresponds to the settler inflow (fig. 8.28 below). As the level in the sump reaches 2,9 Mℓ, the MPS pumps are switched on (indicated by a kink on the rising limb). On all three occasions that the pump is switched on, inflow A(66) exceeds the pump capacity for a certain period, resulting in a maximum dam volume of about 3,6Mℓ (the full capacity A(49) is only 4 Mℓ). After this point the inflow is less than the pumping rate and the level in each case is dropped to the switch off volume A(43) of 1,6 Mℓ after which the level again rises due to inflow.

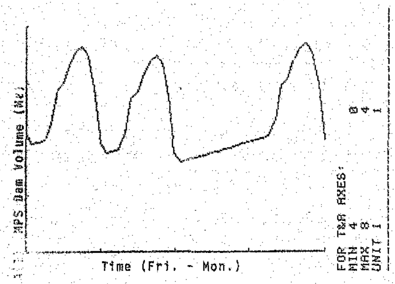


Fig. 8.15 Main Pump Station Dam Volume

Fig. 8.16 shows the fluctuations in the Intermediate Pump Station dam volume. The volume increases after the MPS pumps are switched on. When the volume reaches a certain point A(44), the IPS pumps are switched on. In this simulation the IPS and the MPS pumps have the same capacity A(20) and A(21), therefore the level emains constant and only drops when the MPS pump is switched off. The IPS pump is then able to reduce the volume until the switch-off volume is reached. It should be noted that the graph indicates that the IPS pump was started prematurely (before the start volume of 1Mℓ). This is due to the large 1 hour step size and the high rate of volume increase in the small IPS dam.

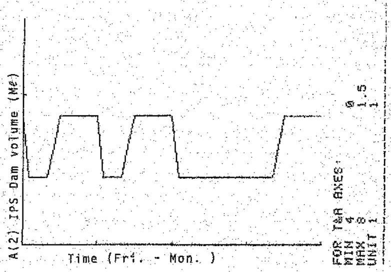


Fig. 8.16 Intermediate Pump Station Volume

Fig. 8.17 shows the fluctuations in the sedimentation tank. The dam volume is always in excess of its 1,6Mℓ maximum volume, again because of the 2 hour time-step and the rapid rate at which the dam fills when the IPS pumps are pumping (see the rising limb on the right-hand side). The excess above full volume is not as dramatic as it looks due to the scale of the graph and only represents an actual rise above the full water level of about 400mm. The way this component is described in the model only allows the excess water volume to reduce by the mud displacement flow from the tank (this could be changed). Fig.8.18 shows the same graph but from a different run where the step size was half an hour, and the water volume can be seen to remain much closer to the full volume of 1,6 Mℓ (note the scale is different).

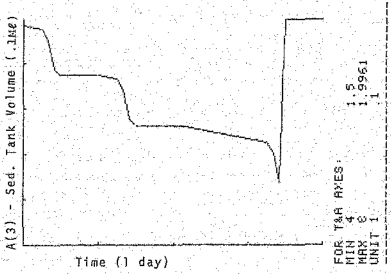


Fig. 8.17. Sedimentation Tank Volume (2 hour time-step)

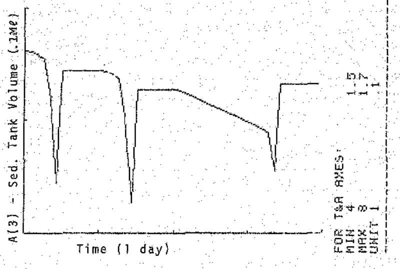


Fig. 8.18 Sedimentation Tank Volume (Half hour step from another run)

Fig. 8.19 shows the variation in the hotwell volume. The top and bottom of the ordinate scale are the full and empty volumes respectively. During the weekdays the volume, or level, fluctuates between a fifth and a half full in response to coldwell demand and make-up water inflow. At the end of the second day (Saturday), water pumped from underground fills the hotwell because the coldwell is full (see fig.8.20) and does not draw from the hotwell. The highest part of the graph is flat, indicating that the hotwell is full and that some overflow is likely, (the printed results show a 1,2 Mℓ overflow from the hotwell over Saturday and Sunday; see A(99) in fig. 8.14). The volume response on Monday is not quite the same as on Thursday and Friday.

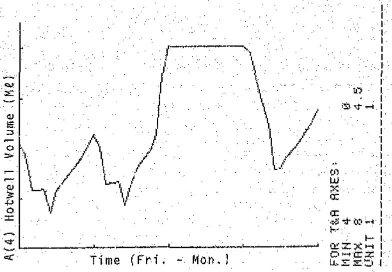


Fig. 8.19 Hotwell Volume

Fig. 8.20 shows the variation in the coldwell volume in response to demand from the mine and hotwell and make-up water inflow. The pattern agrees closely, as does that of the hotwell, with actual graphs of level recordings automatically recorded at the mine.

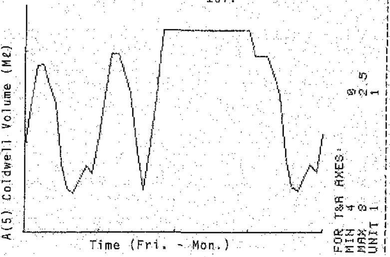


Fig. 8.20 Coldwell Volume

Fig. 8.21 shows the settler inflow rate (Mℓ/d) at the bottom of the mine. The relationship is specified by the user and the values are calculated from table values of the relative flows and a correction factor (see 7.4.1.1 for discussion). These values are easily checked. The ba' flow is due to constant groundwater infiltration. A similar graph is plotted for the cascade outflow (or demand) during the second run below.

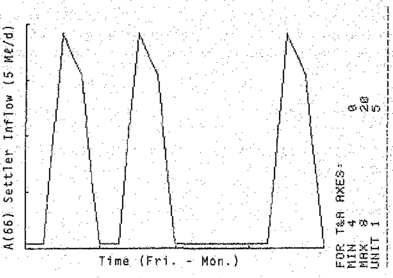


Fig. 8.21 Settler Inflow Rate

Fig. 8.22 shows the pumping flowrate from the MPS to the IPS. The graph shows sloping lines when the pump is switched on and off. This, again, is because of 1 hour time-steps; all points are joined by straight lines.

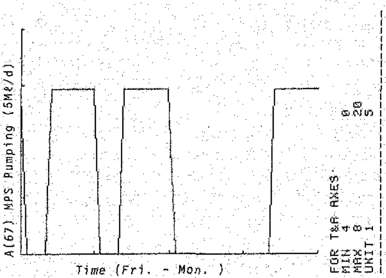


Fig.8.22 Flowrate from the Main Pump Station Mℓ/d

Fig. F23 shows the flowrate from the sedimentation tank to the hotwell. This only occurs during periods when flow is being pumped from the IPS. The flowrate to the hotwell during the pumping period increases due to the decrease in mud displacement water drawn from the tank.

Fig.8.24 shows pumping from the hotwell through the cooling tower and refrigeration plant to the coldwell. It can be seen from the graph of coldwell volume (fig. F20) that flow only ceases when the coldwell is full.

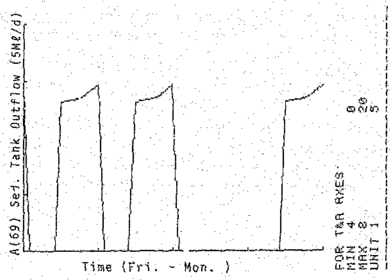


Fig. 8.23 Flowrate from sedimentation tank to Hotwell

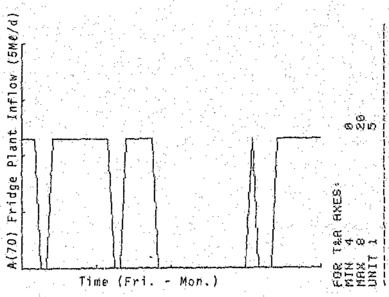


Fig. 8.24 Flowrate to Coldwell from Refrigeration Plant

Fig. 8.25 shows the flowrate from the coldwell to the underground storage dam. Due to the fact that the underground storage dam probably remains full, this pattern is exactly the same as the cascade outflow or demand relationship specified by the user in the tables. The underground storage dam remains full because in the simulation the coldwell, supplying the dam, never runs empty and the maximum flowrate is 21,4 Mℓ/d (see fig. 8.25) which is less than the capacity flowrate of 26 (parameter A(27)).

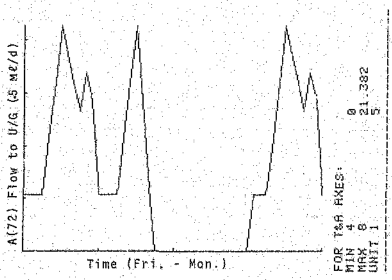


Fig. 8.25 Flowrate from Coldwell to underground (mℓ/d)

Figs. 8.26 and 8.27 give the cumulative volumes of make-up water from the surface supply dam to the hotwell and coldwell respectively. It can be seen from the graphs over which period inflow is being added. The coldwell has priority over make-up water when its volume is low. It can be seen from the two graphs that make-up is only to one well at a time. It is only during the weekend that no make-up is added because the hot and coldwells are full; during the rest of the time the hotwell is almost constantly below the switch-on volume for make-up water (parameter A(48)). Similar graphs can be plotted of make-up water supplied from Rand Water Board.

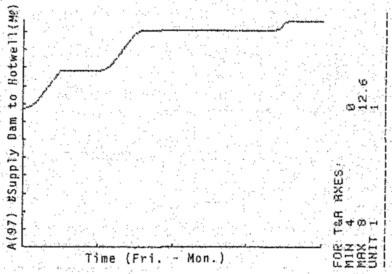


Fig. 8.26 Cumulative volume: Supply Dam to Hotwell

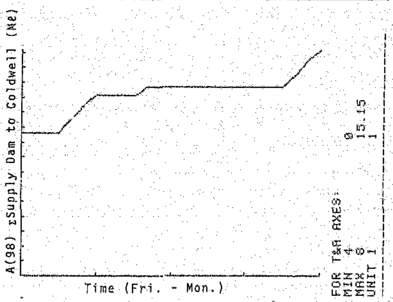


Fig. 8.27 Cumulative volume: Supply Dam to Coldwell

Figure 8.29 is the output given after all the graphs have been plotted. Only 49 points were plotted on each graph even though 100 were specified. This is because only 49 points are available over 4 days using 2-hour time increments.

The rest of the output in Fig.8.29 results from instructions programmed into FINAL. The simulation program then asks the user if another run is required.

```
 49 POINTS PLOTTED
TIME STEP 8.33333333333E-2 DAYS
START&END TIME: 0  8
ALGORITHM:4th ORDER R-KUTTA

CUMULATIVE OUTPUT:
 MAKE UP WATER TO HOT&COLDWELLS
  SOURCE1 1.16666666667
 1.4027777778
  SOURCE2 12.6  15.15
 HOTWELL OVERFLOW 1.21690121786
 MUD REMOVED 4.9717997294
 SALTS LEACHED(kg) 12240
ANOTHER RUN(YorN)?
```

Fig.8.29 Final Output

8.2 Second Run - Transient Effects of Salt Build-Up

The complete model was used in this run to simulate both water quality and quantity aspects. The complete model involves all 15 differential equations and not only the first 7 as in the first run. The program was therefore re-run (fig. 8.30). All the predefined data was re-entered as in the first run above, except for the printing and plotting data. Different graphs to those specified in the predefined data statements were required for this run. The relevant input data was therefore entered through the keyboard as shown in fig. 8.31 No printed out output was specified.

```
DIFFERENTIAL EQUATIONS&VARIABLES

dA(J)/dT=F(J).J=1 to?
15

F(J)=func(T,A(1)..A( 15
)..A(N1))
EXTRA VARIABLES ARE A(16)..A(N1)
(IF NO EXTRAS,ENTER N1=15) N1=?
?
101

A0( 1 ) 1.3 etc. as in the first run.        Fig. 8.30
A0( 2 ) .45


NO.PLOTS,MAX NO.PTS.(See note 5)
?
13
Error 43 on line 1590 : NUMERIC
INPUT
?
13,4000
UNIT SPACING ALONG T-AXIS?
1
FOR EACH VARIABLE TO BE PLOTTED
ENTER:
V-VARIABLE NO.,MN-MIN.VALUE,MX-
MAX.VALUE,U-UNIT SPACING ALONG
VARIABLE AXIS
(V,MN,MX,U)?
65,0,30,5
(V,MN,MX,U)?
79,1000,0,100
(V,MN,MX,U)?
100,30000,5000,1000
(V,MN,MX,U)?
81,15000,10000,500
(V,MN,MX,U)?
93,15,10,1
(V,MN,MX,U)?
15000,5000,500
Error 43 on line 1660 : NUMERIC
INPUT
?
83,15000,5000,500
(V,MN,MX,U)?
85 15000,5000,500
(V,MN,MX,U)?
86,15000,5000,500
(V,MN,MX,U)?
87,15000,5000,500
(V,MN,MX,U)?
88 15000,5000,500
(V,MN,MX,U)?
89,15000,5000,500
(V,MN,MX,U)?
90,15000,5000,500
(V,MN,MX,U)?
92,15000,5000,500

PLOT SPECIFIED VARIABLES FROM                 Fig. 8.31
T=T3 TO T4,USING ABOUT C8 PTS.
(T3,T4,C8)?
4,8,50
```

In fig.8.31 the minimum values exceed the maximum values specified for the graphs of variables 79 to 92. This has resulted in automatic selection of scales in figures 8.36 to 8.47 which ensures that the maximum and minimum values plotted are at the top and bottom of the respective graphs for the best resolution.

At the beginning of this run all storage elements of the system contained good quality water (700 mg/ℓ). The output in the form of graphs (fig.8.35 to 8.47) shows the transient response of the system as the salt concentrations rapidly increase due to the effects of leaching, evaporation, poor quality make-up etc. The third run in section 8.7.3 below, shows the system response under steady state conditions, i.e. when the salt concentrations have reached 'equilibrium' at approximately 13500 mg/ℓ.

This run uses the same initial conditions as the first. Graphs are also plotted from day 4 to day 8 (Fri.0h00 to Mon.24h00). It can be seen from the results of the first run that a 'warm-up' period of about one day would have been sufficient to eliminate discrepancies from incorrect initial values. Execution therefore started on day 3 (Thurs. 0h00) instead of day 0 as before. The time-step size is 0,0416' or 1 hour (fig.8.32).

```
EXECUTE FROM T=T0 TO T9 WITH TIM
E STEP H
(T0,T9,H)?
3,8.001,.0416666666666666

EULER,MOD.EULER OR 4TH ORDER
RUNGE-KUTTA(1,2or3)?
3
```

Fig.8.32

Simulation of the complete model is selected in fig.8.33 because the simulation program expects to find all 15 differential equations in the model (see fig. 8.30 above).

```
1:UNSTEADY FLOWS&NO SALTS
2:UNSTEADY FLOWS&UNSTEADY SALTS
(1or2)?
2
```

Fig. 8.33

The printout shown in fig.8.34 (in response to instructions in INITIAL) includes a list of salt parameters at the end which was not included in the first run because it dealt with quantity aspects only.

```
UNISEL GOLD MINE
Detailed unsteady flow and
unsteady salt model.
================================
UNITS:Ml,d,kg.(NOTE:kg/Ml=mg/l)

CAPACITY FLOW RATES(Ml/d):
 PUMPS:MPS,IPS,HOT-COLDWELL
 14.7  14.7  11.5
 MAKEUP WATER TO HOT&COLDWELLS:
  RWB .5  .5
  SUP.DAM 5.4  5.4
 COLDWELL TO INTERMED.DAM 26

AVERAGE FLOWS              (Ml/d)
 INFILTRATION              .5
 EVAPORATION               .4
                          RATIOS
 TOWER EVAP/INFLOW         .05
 TOWER WINDAGE/INFLOW      .01
 ORELOSS/CASCADE FLOW      .2
 MUD/SETTLER INFLOW        .08
                         (Ml/week)
 MINE INFLOW               70
 SETTLER INFLOW            56.7

VOLUMES(Ml):
 START-STOP VOLUMES:
  MPS PUMP 2.9  1.6
  IPS PUMP 1  .4
 SWITCH ON VOLUMES FOR MAKEUP:
  HOTWELL-RWB&SUP.DAM 1.6  1.6
  COLDWELL-RWB&SUP.DAM .8  .8
 FULL CAPACITY:
  SED.TANK,HOTWELL,COLDWELL 1.6
 4  2
  INTERMED&CASCADE DAMS,SETTLERS
 2  1.5  1.8

SALT CONCENTRATION(kg/Ml=mg/l):
 MAKEUP-RWB&SUP.DAM 700  12000
 INFILTRATION 4000
 LEACHING-X(Y-Ccascade)^2
 X,Y,Z .1  16000  1
```

Fig. 8.34

Fig.8.35 gives the cascade outflow over the 4 days. The relative flows were specified as input and the actual flows plotted are calculated using the correction factors B1 and B2. A similar graph of settler inflow A(66) was plotted in the first run.

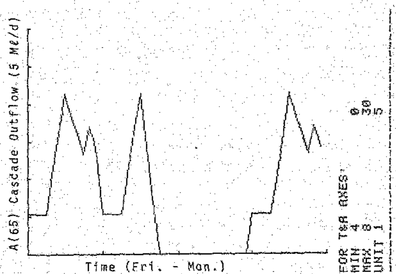


Fig. 8.35 Cascade Outflow

Fig.8.36 shows the decrease with time of the salt concentration added to the MSW from the cascade dams due to leaching. The leaching equation on line 5340 of MODEL describes a decrease in the leaching rate as the salt concentration of MSW tends towards saturation concentration (parameter A(61). The graph only shows the concentration added to MSW not the mass added. The mass added depends on the MSW flowrate as well as the concentration added.

Fig.8.37 shows the cumulative mass of salts leached from the ore which entered the mine water over the four days shown. The curve is slightly concave up and indicates a reduction with time of the salt mass inflow rate.

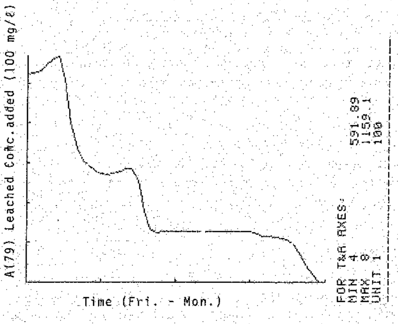


Fig. 8.36 Concentration Added to Used MSW Due to Leaching

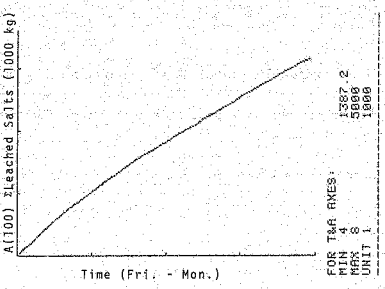


Fig. 8.37 Cumulative Mass of Leached Salts.

Fig.8.38 shows the increase in total salt mass in all the storage components of the system. This is again consistent with the above discussion on figures 8.36 and 8.37. The solid black line along the ordinate axis resulted from a choice of unit spacing marks which are too close together.

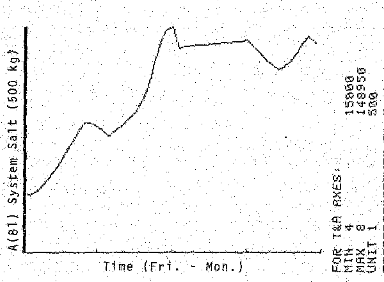


Fig. 8.38 Total Salt Mass in the System (kg)

Fig.8.39 shows the fluctuation with time of the total volume of water stored in components of the system. It can be seen that the system is never in true equilibrium (steady state) as regards quantity aspects. Water is continually being added and removed from the system at different rates. The maximum storage occurs over weekends when mining actively ceases and all free water in the workings gravitates to the settlers from where it is pumped to fill all available surface storage and eventually overflow from the hotwell. The rapid reduction after the maximum point in the graph is due to continual pumping by the main and intermediate pumps and results in overflow because surface dams are full. Continued pumping is carried out to provide safe volume capacities underground for emergency purposes.

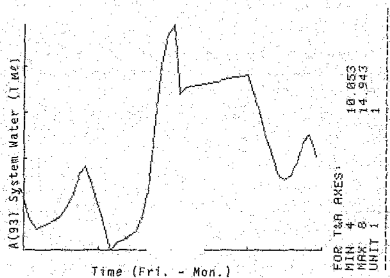


Fig. 8.39 Total Water Volume in the System (Mℓ)

Fig. 8.40 shows the variation of the salt concentration of the settler inflow with time. In practice the fluctuations in the concentration are probably not as great as shown in the graph. Section 7.4.2.1 describes the processes and assumptions used in the derivation of the equation describing the inflow concentration. It may be necessary to modify the model so that the processes which occur in the mine workings are described in a better way.

Figures 8.41 to 8.47 show how the salt concentration increases with time in all of the storage components. The increase or decrease in the salt concentration of any of the components is solely due to the concentration of the inflows to that component. For example, the drop in concentration in the hotwell and coldwell (figures 8.45 and 8.46) is due to the diluting effect of make-up water added.

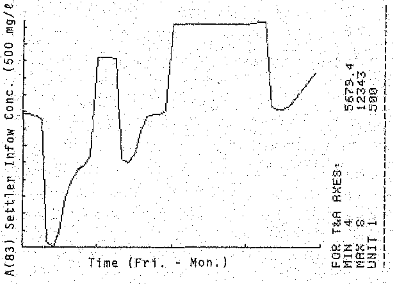


Fig. 8.40 Salt Concentration of Settler Inflow

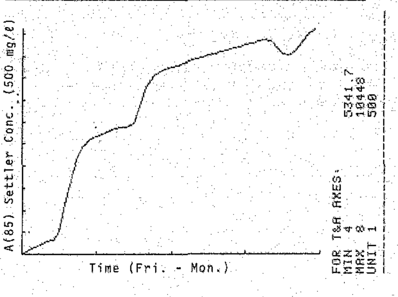


Fig. 8.41 Salt Concentration in the Settler

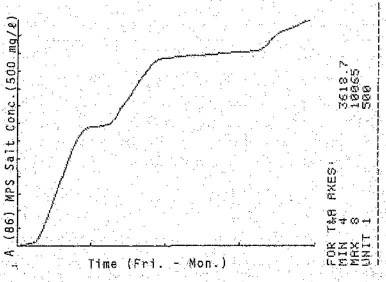


Fig. 8.42 Salt Concentration in the MPS

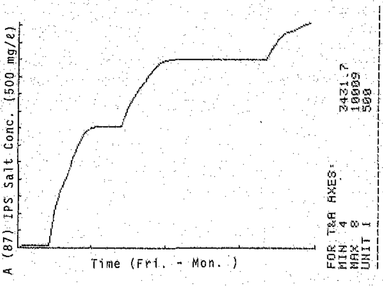


Fig. 8.43 Salt Concentration in the IPS

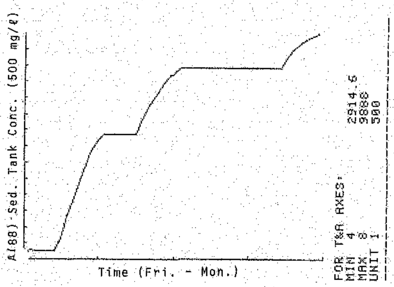


Fig. 8.44 Salt Concentration in the Sedimentation Tank

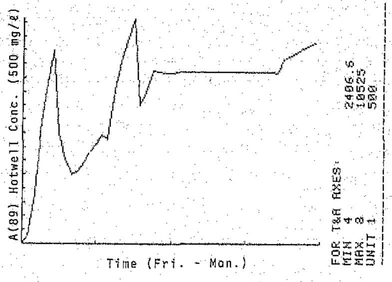


Fig. 8.45 Salt Concentration in the Hotwell

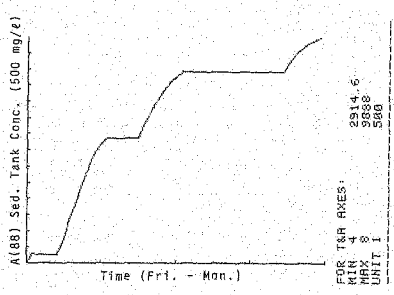


Fig. 8.44 Salt Concentration in the Sedimentation Tank

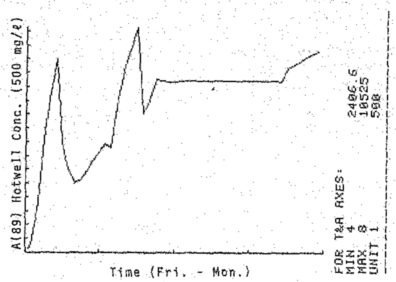


Fig. 8.45 Salt Concentration in the Hotwell

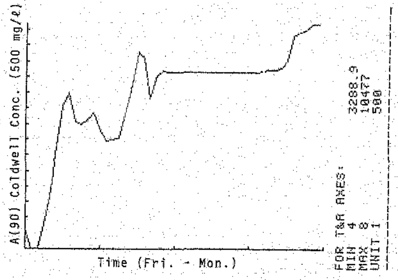


Fig. 8.46 Salt Concentration in the Coldwell

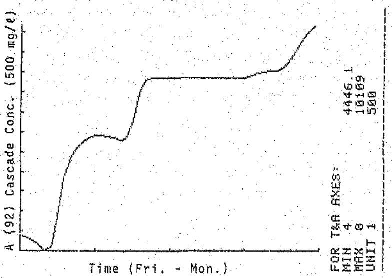


Fig. 8.47 Salt Concentration in the Cascade Dams

The run ended with output (fig.8.48) similar to that of the previous run.

```
 48 POINTS PLOTTED
TIME STEP 4.16666666667E-2 DAYS
START&END TIME: 3  8.001
ALGORITHM 4th ORDER R-KUTTA

CUMULATIVE OUTPUT(M1)
 MAKE UP WATER TO HOT&COLDWELLS
  RWB      .57291666665
 .670138888881
  SUP DAM 6.1875  7.2375
 HOTWELL OVERFLOW 1.42479758683
 MUD REMOVED 2.85418132599
 SALTS LEACHED(kg)
 4600.88854164
```

Fig. 8.48

8.3 Third Run - The Variation of Salt Masses and Concentrations during Steady-State Conditions.

The previous run was used to illustrate the transient effects of salt build-up in the system by starting the run with good quality water in the system. This run was also started with good quality water. The model was run for a number of weeks until the total salt mass in the system reached a maximum and no significant changes in the salt mass and concentrations could be detected from week to week. A number of graphs plotted show the variation with time of the salt concentration in each storage element under steady-state conditions.

Fig.8.49 shows the total salt mass in the system at the end of each day over a period of 14 days. At the end of the period (day 14 or Sun. 24h00) the total mass of 206512 kg is significantly greater than the mass on the previous Sunday (day 7). Further simulation was therefore required to reach steady state conditions. Fig. 8.50 shows the results of simulating for a further 8 days with an increased time-step to speed up computation. At the end of day 7 the total salt mass of 209760 kg is only 1,5% greater than the value for the corresponding day a week earlier (day 0). After 22 days of simulation the system was therefore considered to be close to steady state conditions, i.e. the pattern of salt mass and concentration is repeated from week to week.

The simulation was then continued for a further 7 days. Graphs, showing the variation with time of the salt concentration of water in storage elements, were plotted for the 7 day period and are shown in figures 8.51 to 8.58. It can be seen that the concentrations tend to increase slightly over the weekends and reduce during the week. The concentration fluctuations are generally of the order of a few hundred mg/ℓ. The hotwell and coldwell experience larger fluctuations due to periodic inflow of better quality make-up water.

```
******************************
INITIAL VALUE(S):T= 0
A( 81 )= 10250
T= .999999999998
A( 81 )= 50856.6221472
T= 1.99999999999
A( 81 )= 88733.4252514
T= 2.99999999999
A( 81 )= 115295.0677
T= 3.99999999999
A( 81 )= 133158.349698
T= 4.99999999999
A( 81 )= 144467.154001
T= 5.99999999999
A( 81 )= 192305.204989
T= 6.99999999999
A( 81 )= 187042.092337
T= 7.99999999998
A( 81 )= 170346.979037
T= 8.99999999998
A( 81 )= 166625.469689
T= 9.99999999998
A( 81 )= 165824.768067
T= 10.9999999999
A( 81 )= 165661.998431
T= 11.9999999999
A( 81 )= 165542.187642
T= 12.9999999999
A( 81 )= 212607.232233
T= 13.9999999999
A( 81 )= 208512.468859
*********************************
```

Fig. 8.49

```
TIME STEP 2.08333333333E-2 DAYS
START&END TIME: 0  14.001
ALGORITHM:4th ORDER R-KUTTA

CUMULATIVE OUTPUT(M1)
 MAKE UP WATER TO HOT&COLDWELLS
  RWB      2.44444444485
 2.03472222253
  SUP.DAM 26.4  21.975
 HOTWELL OVERFLOW 2.64377678167
 MUD REMOVED 8.53185385701
 SALTS LEACHED(kg)
 7499.86002098
```

Fig. 8.49 continued

```
*******************************
INITIAL VALUE(S):T= 0
A( 01 )= 206512.468859
T= 1
A( 01 )= 180581.104387
T= 2
A( 01 )= 174629.499010
T= 3.00000000001
A( 01 )= 168638.429282
T= 4.00000000001
A( 01 )= 168979.706243
T= 5.00000000001
A( 01 )= 169239.905545
T= 6.00000000001
A( 01 )= 223316.794827
T= 7.00000000001
A( 01 )= 209761.604796
T= 8.00000000001
A( 01 )= 182117.965232
*******************************
TIME STEP 4.16666666667E-2 DAYS
START&END TIME: 0  8.01
ALGORITHM:4th ORDER R-KUTTA

CUMULATIVE OUTPUT(M1)
 MAKE UP WATER TO HOT&COLDWELLS
  RWB      3.59375000025
 3.32986111126
  SUP.DAM 38.8125  35.9625
 HOTWELL OVERFLOW 4.06857436851
 MUD REMOVED 13.5036535874
 SALTS LEACHED(kg)
 9774.33352094
```

Fig. 8.50

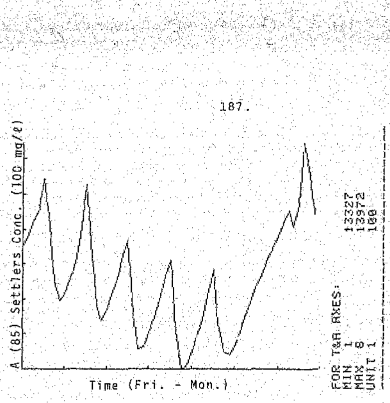


Fig. 8.51 Salt Concentration in the Settlers

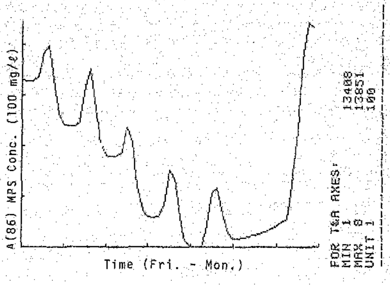


Fig. 8.52 Salt Concentrations in MPS Dams

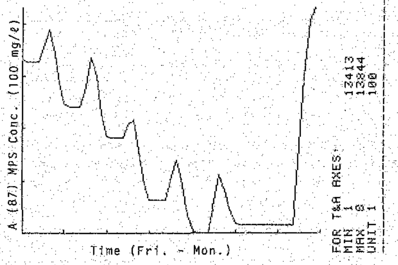


Fig. 8.53 Salt Concentration in IPS Dams

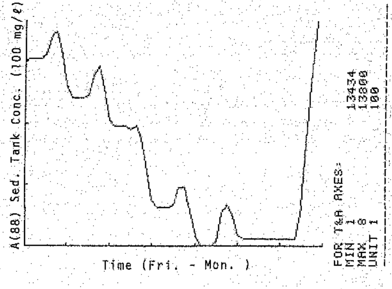


Fig. 8.54 Sedimentation Tank Salt Concentration

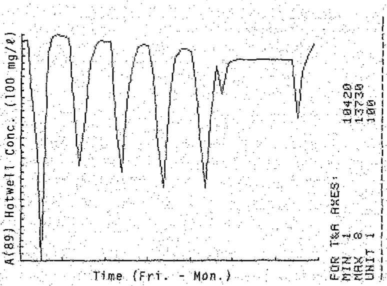


Fig. 8.55 Hotwell Salt Concentration

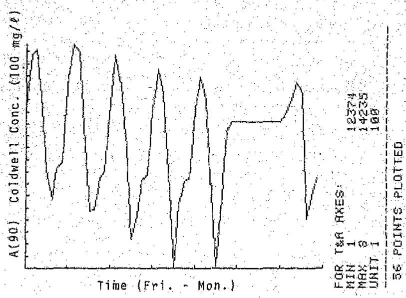


Fig. 8.56 Coldwell Salt Concentration

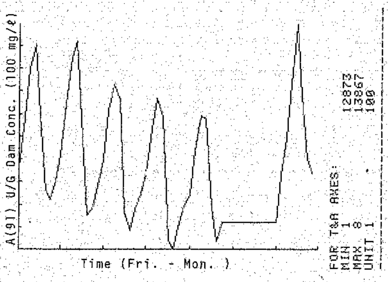


Fig. 8.57 Underground Storage Dam Salt Concentration

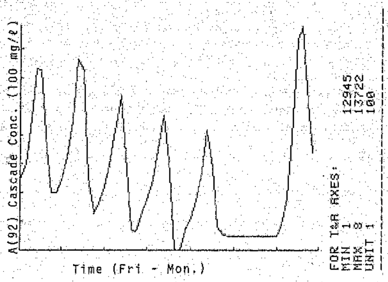


Fig. 8.58 Salt Concentration in the Cascade Dams

```
 56 POINTS PLOTTED
TIME STEP 4.16666666667E-2 DAYS
START&END TIME: 1  8.01
ALGORITHM:4th ORDER R-KUTTA

CUMULATIVE OUTPUT(Ml)
 MAKE UP WATER TO HOT&COLDWELLS
  RWB     4.68402777787
 4.40277777778
  SUP DAM 50.5875  47.55
 HOTWELL OVERFLOW 4.88215071042
 MUD REMOVED 17.769580517
 SALTS LEACHED(kg)
 11732.6161884
```

CHAPTER NINE

9. CONCLUSIONS

The pumping of saline water from South African gold mines contributes significantly towards the mineral pollution of the water resources of the Vaal Basin.

The decreasing quality and increasing demand for water supplied from the Vaal Basin to various consumers has resulted in the already high and increasing cost of good quality raw water supplied to gold mines and stricter regulations for discharge from the mines.The quality of recycled mine service water (MSW) is generally poor which limits surface discharge, which in turn causes recycling and higher build-up of dissolved salts in the water.

The quality of MSW is of prime importance, since poor quality water can have a detrimental effect on the performance and life of all equipment exposed to it. High chloride concentrations cause corrosion of metal pipework and equipment. Sulphates and carbonates give rise to scaling and blockages in heat exchange equipment found in refrigeration plants. New hydraulic machines using oil-water-emulsions are being introduced underground as part of a conversion programme. Water hardness can cause a breakdown in the emulsions. There is also the problem of potability to be considered.Although the extent of such problems is largely unknown at present, the cost implications are probably considerable.

The increasing costs of raw water, and supply quantity and discharge quality constraints, limit the dilution of mine water as a possible solution to water quality problems. Other means of dealing with pollution are prevention at the source, desalination, neutralising or protective chemicals and prevention of pollution. The probable solution may lie in some combination of these techniques.

Solution to mine water quality problems involve expensive techniques and flowrates which can be in excess of 100 megalitres per day. Because of the high cost of solutions, optimisation should ideally be used to minimise costs (capital, power, labour, material etc.).

Unfortunately direct optimisation techniques, applied to complex systems such as mine water reticulation, invariably require gross over-simplification of the systems before the techniques can be applied. The optimal solutions obtained in this way cannot be guaranteed.

Design changes in one part of a water system may have far-reaching repercussions which are difficult to trace manually. The small improvement in water quality during peak demand period, which results from a desalination plant, for example, may not justify the costs. Mine water systems are complex and dynamic and solutions require an understanding of the system. Simulation of mathematical models is often the only practical means of studying such systems.

Simulation models of existing and proposed water systems can be experimented with without the cost, risk and expenditure of time involved in experimenting with the real system. Short and long term effects on water quality of design and policy changes can be simulated and the results used for forecasting. Important processes and sensitive parameters and variables can be identified and exploited to bring about improvements. Simulation can be used as a design tool on site in order to study the effect of a proposed underground dam or an extra pump for example.It can be used as a management tool for operating a system and may even form part of a monitor and control system designed to ensure optimum operating conditions. The pumping for example could be sequenced to minimise peak power demand for the mine. The communication of simulation results is generally easier and more convincing than with other methods.

The dynamic behaviour of mine water systems (the variation with time of salt masses, concentrations, volumes, flows etc.)can generally be described by a system of linear ordinary differential equations. Because the equations are linear, the systems are linear and the principle of superposition can be applied.Complicated solutions to a system of linear differential equations of any order can be expressed as the sum of simple solutions. Fortunately, mine water models can be classed as initial-value problems because the state of the system is generally known or can be estimated at certain times.

Various single-step and multistep numerical methods for solving single and systems of differential equations are discussed. Three single-step methods were selected for inclusion in the simulation program; the Euler, the Modified Euler and the Fourth-Order Runge-Kutta methods. An important objective in writing the simulation program was simplicity, readability and ease of maintenance. The more powerful multistep methods were therefore excluded because it was felt that the complex programming required did not justify the increase in efficiency. The Fourth-Order Runge-Kutta algorithm included, is one of the most powerful and simple single-step methods to program. For a given time-step size the more powerful the algorithm the more accurate the results but the longer the execution (or simulation) time. The less efficient Euler and Modified Euler algorithms were therefore included in the simulation program because it has been found necessary at times to use a smaller time-step size for reasons other than accuracy.

The general simulation program written can be used to simulate mathematical mine water models or any other models described by similar equations. The program is fully interactive with convenient facilities for the input, simulation and output stages of simulation. The program will accept graphical input (coordinate points) and can produce output in the form of graphs. The program and models are written in BASIC language for use on a Hewlett Packard HP-85 desktop computer.

Two models of Unisel Gold Mine were conceived and simulated. Complete data concerning flows and operating relationships at the mine as well as chemical results from leaching tests conducted at the University were not available. The best available data was used in the model simulations but the results cannot be guaranteed to represent the real system response. Although the Unisel model simulations are used to provide examples of the model construction and simulation procedures suggested, the models are at a stage where they can be used practically in analysis and design situations given the required data. Certain deductions based on simulations of the present stage models can nonetheless be made and are given below.

The first model of Unisel Gold Mine is a very simple representation of the system for which a general analytical solution is derived. The model assumes constant water volume and flows and describes the change of salt concentration with time. The analytical model can be applied to any mine to obtain approximate answers to the following: the time it will take for the average salt concentration of the system to change from one value to another; the final concentration after a finite or infinite time lapse given the initial concentration. The use of the simulation program and the application of step-size and convergence criteria are illustrated in simulation examples using the simple model representation.

The second model of Unisel Gold Mine uses 15 differential equations to desribe the dynamic behaviour of the system storage components (changes in salt masses and water volumes with time). Over 10^ variables and parameters are used in the model to enable the user to monitor and manipulate the system during experimentation.

Three experimental runs were performed on the detailed model.In the first run graphs were plotted showing volumes of water in the storage components and different flowrates. The results are for a four day simulation period which includes a weekend and illustrates the dynamic effects, for example, of pumps switching on and off and make-up water additions when levels are too low. During weekends most underground water is pumped into dams on the surface, which eventually fill and overflow. This phenomenon is depicted graphically and the overflow volume is printed out at different times. The overflow is an example of a problem situation where different pumping policies, capacities or times, or different dam volumes could be tried to eliminate overflow. The first run dealt only with water quantity aspects.

The second run of the detailed model simulated the first four days (including a weekend) of rapid salt build-up in the system after starting with good quality regional water board water in the system. The salt build-up is due to phenomena such as leaching of salts from ore, poor quality make-up additions and the concentrating effect of evaporation. The complex effects of the dynamic water quantity changes on the salt build--up can be seen. Disturbances of the system, for example using only good quality make-up water, will cause the salt concentrations to change with time until a new salt balance is achieved. It may be useful to simulate the transient effects of design changes or disturbances in order to predict the system response (quality and quantity changes)at different times and the final outcome.

The last run of the detailed Unisel model also simulated the salt build-up in the various system components starting with good quality water in the system. A few weeks simulating the salt build-up were required before the salt concentrations began to repeat themselves from week to week, thus signifying steady-state conditions. Detailed output was obtained during a further week of simulation under steady-state conditions. The average steady- state salt concentration of the system was about 13500 milligrams per litre (mg/ℓ) of total dissolved solids. The graphs show that even under these conditions the salt concentration of water in storage elements can fluctuate by several hundred mg/ℓ from day to day and from week to week. Fluctuations of a few thousand mg/ℓ are predicted for surface dams receiving make-up water. The experiment illustrates the errors that could result if, for example, a desalination plant is sized assuming average flows and concentrations.

The detailed model of Unisel, once properly verified, can be used to predict the variation in quality with different operating conditions. Increased dilution with good quality water may be the only cost effective means of improving the quality of Unisel's mine service water. Make-up water to replace water losses such as evaporation, is generally added to the surface dams during the day when demand from the mine is high and surface dam levels are low. During the night and over weekends the surface dams are generally full and make-up water is not added. It appears that very little good quality water board, make-up water is added to the mine water during the day as this water is used extensively for other surface requirements such as compounds and the refrigeration plant. Large quantities of poor quality make-up water from an adjacent surface pan is added instead. A solution may be to add water board water to the surface dams during the night and weekends when the supply pressure is at its greatest. This would ideally require spare surface dam capacity (before the chilling process) during these periods.

Spare surface storage capacity can be provided in a number of ways: larger surface and subsurface dams; remove a certain amount of the worst quality water pumped directly from underground; store more water in the underground pump sump dams during low demand periods. Pumping from underground normally continues until the underground dam levels are lowered to the pump switch-off levels. This is done regardless of whether the surface dams are full and overflowing. The provision of spare surface capacity at night and during weekends may make it possible to dilute the poor quality water at Unisel with better quality water board water to a greater extent than done presently.

It may be useful to modify the simulation program to store the input data on different tape files. Input data could then be read from tape instead of the data statements as done presently. This would be simpler for the user and more space would become available for larger models. Should simulation be used for optimisation then a facility could be built into the program to execute repeated runs with parameter changes. Changes could be made using sampling and search procedures discussed to find the best set of parameters for calibration or the set of decision variables for an optimum design.

APPENDIX A.

MODULAR STRUCTURE OF THE SIMULATION PROGRAM

As described in section 4.3.2, structured programming embraces a collection of methods which makes programs more reliable and easier to understand and maintain than programs written in the normal way. Fig. A1 shows the modules and their names, which make up the simulation program. Starting with the first module at the top, MASTER, there are five levels in the program, going from the top to the bottom. Each module performs a basic function, as the module name suggests.Each module is described in detail in appendix B. Modules at a higher level call the modules at the level below as shown by the lines in Fig. A1 connecting the modules. Modules at a lower level generally deal with more detailed aspects which are required by the modules in the levels above. In this way details are relegated to minor modules at lower levels and all modules are easier to follow because they are designed to occupy a page or less of source code. The source code of the program is listed in appendix C. Table A1 lists the modules in the order in which they appear in the module descriptions in appendix B and the source code listing in appendix C. The program line number for the first line of each module is also given in table A2.

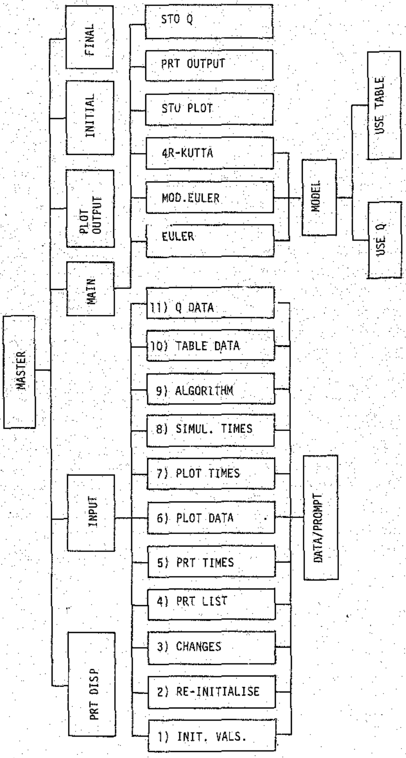


FIG.A1 PROGRAM STRUCTURE

Table A2 Sequence of Program Modules

Module	Line No.
MASTER	5
PRT/DISP	610
INPUT	700
1) INIT VALS	1030
2) RE-INITIALISE	1110
3) CHANGES	1180
4) PRT-LIST	1280
5) PRT TIMES	1450
6) PLOT DATA	1500
7) PLOT TIMES	1690
8) SIMUL TIMES	1740
9) ALGORITHM	1790
10) TABLE DATA	1830
11) Q DATA	1280
DATA/PROMPT	2921
MAIN	2930
EULER	3175
MOD.EULER	3190
4R-KUTTA	3340
STO PLOT	3560
PRT OUTPUT	3670
STO Q	3740
USE Q	3840
USE TABLE	4020
PLOT OUTPUT	4350
MODEL	4630
INITIAL	8000
FINAL	9000

APPENDIX B

DESCRIPTION OF THE SIMULATION PROGRAM MODULES

Yourdan (1975) suggests the use of HIPPO (Heirarchical-Input-Processing-Output) daigrams to describe each module of a modular structured program (see section 4.3.2). The method was developed by IBM and is used to describe separately the inputs to, the function of, and the output from each module. HIPPO diagrams consist of three blocks, one each for the input processing and output, within which is listed the relevant module functions. Instead of HIPPO diagrams, the input, processing and output functions of each module is described in this section in three separate paragraphs, including examples where necessary. Input in the context used here is information, data or instructions from keyboard interaction with the user, from data statements or from tape files. Output is printed or displayed results in tabular or graphical form. All modules referred to in the discussion appear in capital letters (MASTER,3) CHANGES, MODEL etc.). Numbers preceded by L, for example L1027, refer to the line number in the program where the statement appears.

The modules are described in the same order as they appear in the simulation program. The arrangement of the modules (program structure) is described in fig.A1 in appendix A. The module names are listed in table A2 in the order in which they appear in the simulation program, along with the first program line number for each. The first program line number of each module is given next to the module name below.

MASTER (L5)

Input The number of differential equations and variables used in the model.

Processing Dimension all variables, arrays and matrices as either integer, short or real precision or as a string variable.

INPUT is called to obtain input which is essential for the first run (L290-304). A flag Z2 tells INPUT which type of input is required each time (e.g. initial values printout list etc.). Flag Z1 is set to zero and tells INPUT not to give the user any options, but to request only the input specified by Z2. At the end (L304) Z1 is set to 1 and INPUT again called so that the user can select any of the 12 input options to make changes or corrections. Call module INITIAL to execute any user specified instructions before simulation begins (L310). The simulation clock is set to the starting time TO. Checks are made to ensure that all the start and completion times specified for plotting (L380), printing (L493) and execution (L502) are consistent with the sign of the step size (increasing or decreasing time). Any inconsistencies result in an error message and transfer to INPUT for corrections. A variable PO is defined (L400) so that plotting data can be stored every PO time-steps and to ensure that the resolution of the graphical output (number of points on graph) approximates the resultion specified by variable C8, specified in PLOT TIMES. A tape file called P-DATA is opened for data to be plotted and P is used as a counter to storing data on file. Similarly output is stored in queues every EO steps using E as a counter (L460-480). Call module STO PLOT at the start of the simulation to store initial data if plotting data is required from the start of the simulation (L410-430). Similarly, STO Q is called if any queues exist (L470). After the above initialisation, MAIN, is called to execute the simulation (L506). If the flag F1 indicates that plotting of output is specified, then PLOT OUTPUT is called (L510). FINAL is called last to carry out any user-specified instructions after the simulation. If no more simulation runs are required then the execution will terminate at the end of this module.

Output. The program heading and a list of notes giving the current size of arrays and matrices dimensioned, is printed. A list of the initial values used for the simulation is given (L498). PRT/DISP is called at various places in the module to determine whether output is to be printed on paper or displayed on the screen.

PRT/DISP (L610)

Input. The user specifies whether the input, output, interactive conversation etc., must be printed on paper or displayed on the screen.

Processing. Only DISP statements are used for output in the program. If the NORMAL statement was executed last then output only appears on the screen. If the PRINTALL statement was executed last then output is printed on paper and displayed on the screen.

Ouput. None.

INPUT (L700)

Input. Input options (1 to 12) are entered one at a time. Each option represents a different change or correction which can be made to the input (initial values, plotting times etc). After option 12 has been selected the user specifies whether the output is to be printed or displayed.

Processing. If flag Z1 equals 0 (L710) then one of the input modules (1) INITIAL VALS to 11) Q DATA) is called according to the value of Z2 (L1000) which is defined in MASTER (L292). Control is then returned to MASTER (L1005). If Z1 equals 1 (set on L304) then the list of 12 options is displayed on the screen and the user selects one, the relevant modular is called, and the changes are made. The option list is repeatedly displayed until option 12 is selected and the simulation run is begun. Control is returned to MASTER after a run.

Output. A list of the various changes which can be made to the existing input data stored.

1) INIT. VALS. (L1030)

Input. The N1 initial values of all the variables used in the model can either be entered through the keyboard or read from data statements defined in INITIAL (L1090).

Processing. The initial values of all variables, required to start the simulation, are stored in the array AO. The values are not affected during simulation, and should a re-simulation be required from the same initial conditions then the module RE-INITIALISE is called. The next module discussed, RE-INITIALISE, is part of this module and will be executed every time INIT VALS is called.

Output. None

2) RE-INITIALISE (L1110)

Input and output. None

Processing. During simulation, the values of all the variables are continuously being modified. The arrays A and B are used to store the latest values and the values from the last iteration respectively. This module initialises A and B to the same values as AO at the beginning of the first run (see INIT. VALS. above), or whenever the user wishes to start or continue a simulation with the original initial values. If queues are being used (F3=1) then all queues will also be initialised (L1130).

3) CHANGES (L1180)

Input. The number of changes, the variable numbers and their new initial values.

Processing. This module performs the same function as INIT. VALS. above, but allows the user to easily make a few changes to the initial values instead of entering all the values in each time as required in INIT. VALS. Both the permanent record of initial variables (array AO) and the latest values (arrays A and B) are modified.

Output. None.

4) PRT.LIST (L1280)

Input. If the user requests printed output (L1310) then the number of variables and their numbers are entered. The variable list can be entered through the keyboard, or read from data statements in INITIAL.

Processing. The values of certain variables are required during simulation. This module enters the required list of variables into an array A1, in the order in which they must be printed or displayed. The next module PRT TIMES determines when the output is displayed (and printed).

Output. None

5) PRT TIMES (L1450)

Input. The start and finish times, and the frequency for displaying (or printing) the variables specified in PRT LIST.

Processing and Output. None.

6) PLOT DATA (L1500)

Input. The number of graphs to be plotted, the maximum number of points available for each graph and the units along the time axis. For each variable to be plotted the variable number, the minimum and maximum values, and the units along the variable axis are required. The data is entered via the keyboard or data statements.

Processing. Flag F1 is set to tell MAIN whether there is any plotting, and if so to store the relevant plotting data on tape.

Outputs. None

7) PLOT TIMES (L1690)

Input. Start and finish times for plotting and the approximate number of points that should be used.

Processing and Output. None

8) SIMUL TIMES (L1740)

Input. The start and finish times for simulation of the model and the time-step size to be used

Processing and Output. None.

9) ALGORITHM (L1790)

Input. The solution algorithm to be used in the simulation; Simple Euler, Modified Euler or the Fourth-Order

Runge-Kutta.

Processing and Output. None.

10) TABLE DATA (L1830)

Input. The number of tables D9 used by the model. For each table: the number of times the table is used during the simulation; the number of data pairs in the table; the simulation times when use of the table is to start; the data pairs in the table (time increment after table start, table value). Input can either be entered through the keyboard or read from data statements in INITIAL.

Processing. Certain model relationships are best presented in table form. This module only defines the arrays and matrices required to store the data. If the model requires the use of tables then MODEL defines the table number and time and calls USE TABLE to interpolate a table value. The notes at the beginning of the program list the max. number of tables, start times etc.

Output. Error messages are given (L2160 and L2330) if the table start-values and the time-increment-after-start-values do not increase.

The example which follows illustrates how the arrays and matrices are used to store table information. The example, comments in the program and the list of symbols in appendix D should be sufficient to explain how the module operates.

Example

Suppose the observations made in fig. B1, e.g. water flow rate versus time, are to be used in the model.

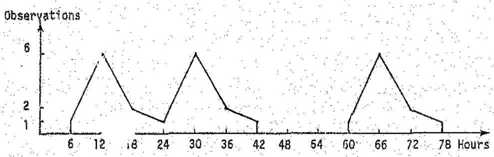


Fig. 81 Graph of Observations at Different Times

The data can be represented as one table used three times during simulation. The three starting times for the table are 6, 24 and 60. Each table consists of 4 data pairs:

Time increment after start	0	6	12	18
Table value	1	6	2	1

Assuming that this is the second graph of observations being represented by a table, i.e., table No. 2, then the above data would be stored in D1, D2 and D3 as shown.

Matrix D1

Row No. = table No.		No.of Times Used	1st Row of Array D2	No. of Data Pairs	1st Row of Matrix D3
	1	-----	-----	-----	-----
	2	3	6	4	5
	3	------	-----	-----	-----

Array D2

	Start Values	Row
	-	1
Assume Table 1 is used 5 times	-	2
	-	3
	-	4
	-	5
Table 2 starts at row No.6	6	6
	24	7
	60	8
Table 3	-	

Matrix D3

	Time Increment After Start Value	The Table Value
	-	-
Assume Table 1 has 4 data pairs	-	-
	-	-
	-	-
Table 2 starts at row No.5	0	1
	6	6
	12	2
	18	1
Table 3	-	-

This example is continued with USE TABLE below, where it is shown how a value is interpolated, given the time and table number.

11) Q-DATA (L2380)

Input. Yes or no to the use of queues. The number of queues and the variable number and lag time corresponding to each queue. The total spaces available for all the queues. Yes or no to the space allocation for each queue. An initial value for each queue. The module is broken down into a number of sections so that changes can be made to particular inputs without having to re-enter all input data. Input can either be entered through the keyboard or read from data statements. Reference is made to the notes at the beginning of the program (L50 to L100) which state the maximum number of queues, lag times etc., and which arrays and matrices have been dimensioned for this.

Processing. Queues are required when it is necessary to know the value of certain variables at some time earlier than the current time in the simulation (see 4.2.1d). A flag F3 is set to tell MASTER and MAIN whether queues are being used or not. The processing involved is best described by the example below.

Output. The space allocation for each queue (column two of matrix E1) in the example below is displayed for approval or re-allocation.

Example. Assume two queues, 1 and 2, are required to store past values of variables, 7 and 16 respectively. Queues 1 and 2 store 2 and 4 values respectively, as shown in array E2. The length of the queues is automatically proportioned and displayed. (L2670 to L2730) according to the queue-lag times and the total number of spaces specified - E8.

MATRIX E1

variable no.	no. of values stores in queue	1st row no. of array E2
7	2	1
16	4	3

MATRIX E3

Row No = Queue No.	initial values	lag times
1	66	6
2	92	12
3		

ARRAY E2 (initial values in queues)

Row		
1	66	2 values of variable No. 7 stored in queue No. 1
2	66	
3	92	4 values of variable No. 16 stored in queue No. 2
4	92	
5	92	
6	92	
7		Queue 3 (if required)

The module has two entry points, the first at the beginning of the module and the second at line 2860. There is only one exit at the end of the module. The portion of the module after the second entry point is used to initialise all the queues (array E2) to the initial values stored in matrix E3. This is done before the first simulation, and for a re-simulation if the RE-INITIALISE option is selected. In the example, the queues 1 and 2 are initialised to 66 and 92 respectively. The initial values are stored in matrix E3 and are not altered during simulation. Array E2 is used to store the queue values generated during simulation. The diagram shows the initial values stored in array E2 before the commencement of simulation.

The example is continued with STO Q and USE Q where it is explained how the queue array E2 is updated with the latest variable values, and how variable values at particular lag times are interpolated from E2.

DATA/PROMPT (L2921)

Input. Choose whether input data is to be entered from

ARRAY E2 (initial values in queues)

Row		
1	66	2 values of variable No. 7 stored in queue No. 1
2	66	
3	92	4 values of variable No. 16 stored in queue No. 2
4	92	
5	92	
6	92	
7		Queue 3 (if required)

The module has two entry points, the first at the beginning of the module and the second at line 2860. There is only one exit at the end of the module. The portion of the module after the second entry point is used to initialise all the queues (array E2) to the initial values stored in matrix E3. This is done before the first simulation, and for a re-simulation if the RE-INITIALISE option is selected. In the example, the queues 1 and 2 are initialised to 66 and 92 respectively. The initial values are stored in matrix E3 and are not altered during simulation. Array E2 is used to store the queue values generated during simulation. The diagram shows the initial values stored in array E2 before the commencement of simulation.

The example is continued with STO Q and USE Q where it is explained how the queue array E2 is updated with the latest variable values, and how variable values at particular lag times are interpolated from E2.

DATA/PROMPT (L2921)

Input. Choose whether input data is to be entered from data statements or through the keyboard by the user responding to prompts which appear on the screen.

Processing. A flag F4 is set according to the choice made and is used by the module, which called this one, to carry out the selected method of data entry.

Output. None

MAIN (L2930)

Input and Output. None

Processing. This module is called by MASTER and is resposible for the execution, displaying (and printing if specified) and storage of plotting and queue data. A do-loop with counter C is responsible for carrying out the iterations. T1 is the time at the start of the iterations and T, which is used as the current time in the model, is defined by the solution algorithm discussed below. Depending on the flag F3 and the counter E, STO-Q is called to modify the queue values. Depending on the flag F1, the time range specified for plotting, and the counter P1, STO-PLOT is called to store plotting data. If the number of values stored C1 is greater than the number of spaces available per plot C9 then no more data will be stored Depending on the flag F0, the specified time range for displaying tabular output (and printing if specified) and the counter S, PRT OUTPUT is called for output.

EULER (L3175)

Input and Output. None

Processing. The simple Euler Algorithm is the simplest, least accurate of the three algorithms available in the program to numerically solv the differential equations describing the model. MODEL is called once to get K-values for each differential-equation-variable used in the model. The latest variable values are then calculated and stored in array A at the end of the iteration. Section 3.2 describes the mathematics of the different numerical methods used in the program.

MOD. EULER (L3190)

Input and Output. None.

Processing. The Modified Euler algorithm is used i· e same way as in Euler above except that MODEL is cal. twice per iteration instead of once and it is more accurate.

4R-KUTTA (L3340)

Input and Output. None

Processing. The Fourth-Order Runge-Kutta algorithm is the most accurate of the three algorithms available in the program. It is used in the same way as in Euler above, except that MODEL is called four times per iteration instead of once.

(STO-PLOT L3560)

Input and Output. None

Processing. This module is called by MAIN to store the specified variable values and the corresponding time, T, on the tape file called P-DATA using buffer number 1 (see user manual). After the simulation is complete, PLOT OUTPUT will use the file data to plot the variable values against time. Column 1 of matrix M contains the E9 variable numbers which are to be plotted. If, for example, variables 1, 4 and 9 are to be plotted at times T_0, T_1, T_2 then the corresponding variable and time values will be recorded on tape during simulation as follows:

T_0	V_0^1	V_0^4	V_0^9	T_1	V_1^1	V_1^4	V_1^9	t_2	V_2^1	V_2^4	V_2^9	T_3	V_3^1

This module will modify the minimum and maximum scale values of a graph if a variable value stored on tape is beyond the range specified in PLOT DATA above.

PRT [illegible] [illegible]670)

Input.

Processing. A do-loop is used to generate the output.

Output. A list of the current variable values from the completed iteration, and the corresponding simulation time T will be displayed and printed if specified (only those variables specified in array A1).

STO Q (L3740)

Input and Output. None

Processing. Each time this module is called, the value at the bottom of each queue is lost as the queue is shifted one position to make space for the latest variable value at the top of the queue. The queue values are stored in array E2 and matrix E1 is used to store the variable number, the number of values in the queue and the position of queue values in E2 (see Q-DATA for more details).

The example given in Q-DATA is continued here. If variables 7 and 16 (queue numbers 1 and 2 respectively) have the values shown in table C1 at the different simulation times when STO Q is called, then the queue values are changed as shown in array E2.

Table C1 - values of variables 7 and 16 at different simulation times.

time	T_0	T_1	T_2	T_3	T_4
variable 7	66	67	69	75	81
variable 16	92	85	80	77	75

Note
the values at T_0 are the initial values shown in Q DATA

Array E2

	T_0	T_1	T_2	T_3	T_4
Queue No.1	66	67	69	75	81
variable 7)	66	66	67	69	75
	92	85	80	77	75
	92	92	85	80	77
Queue No.2	92	92	92	85	80
(variable 16)	92	92	92	92	85

Time T_0 T_1 T_2 T_3 T_4

This example is continued with USE Q below where the interpolation of variable values from E2 at particular lag times is explained:

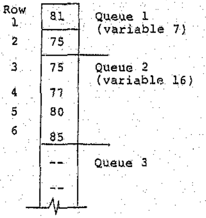
USE Q (L3840)

Input. None

Processing. This module is called by MODEL whenever the value of a particular variable is required at some time earlier on in the simulation.

The <u>example</u> in STO Q explains how values of the variable chosen in Q DATA are entered into queues. MODEL specifies the queue number J and a check is made (L3900) to ensure that the queue exists. If not, execution is terminated. The lag-time (the difference between current simulation time and the time of interest) is specified for each queue in Q DATA. The lag-time can be changed with each call by MODEL to USE Q by specifying a new lag-time L = 0.6, for example. A value of L=0 is interpreted by USE Q as an instruction to use the original lag-time specified in Q DATA. A check is made (L3920) to ensure that any new lag-time specified, L, is less than or equal to the original lag-time specified in Q DATA. This check is to ensure that errors will not result from excessive extrapolation of values stored in the queue. Before exit from the module L is set equal to 0 (L4005) to ensure that the original lag-time will be used when USE Q is called again, unless L is changed by MODEL. The interpolation/extrapolation of values from a queue for a given lag-time is illustrated using the last array of queue values, E2, shown in the example given in STO Q.

<u>Array E2</u>



Assume that queue values are entered into E2 every 6 iterations, i.e. E0=6. The iteration time-step H=0,1 hours and 5 iterations (E+1=5) hi e elapsed since 81 and 75 were stored in queues 1 and 2 respectively. Since successive row numbers in E2 above represent a lag-time difference of EO.H=0,6 hours, values can be interpolated/ extrapolated from E2 by calculating a row number R2 equivalent to the specified lag-time L. The row number R2 is not necessarily an integer and the two queue values (V6 and V7) in the rows R1 and R3 closest to R2 can be used to interpolate/extrapolate the queue value at L. For example:

1) What is the approximate value of variable 16 (queue 2) at a lag-time L=1,16 hours earlier in the simulation?

The value required is at the row number.

R2=3+1,6/0,6-5/6 = 4,833

of array E2 above. R2 (L3940) equals the start row no. of queue 2 plus the time-lag, i.t.o. the number of rows, less the elapsed time, i.t.o. rows, since the last value was stored in E2. The queue values at row 4 (V6=77&R1=4) and row 5(V7=80&R3=5) can now be used to <u>interpolate</u> the value at row 4,833.

The interpolated value (L4000)

FNQ = V6+(V7-V6)(R2-R1)/(R3-R1)
= 77+(80-77)(4,833-4)/(5-4)
= 79,50

is returned to MODEL.

2) As for 1), but the lag-time L=0,2 hours?

R2=3+0,2/0,6-5/6 = 2,50

which is less than the first row number of queue number 2 and therefore rules out the possibility of interpolation. The first two queue values in rows 3 and 4 are used to extrapolate a value for row hours and 5 iterations (E+1=5) have elapsed since 81 and 75 were stored in queues 1 and 2 respectively. Since successive row numbers in E2 above represent a lag-time difference of EO.H=0,6 hours, values can be interpolated/extrapolated from E2 by calculating a row number R2 equivalent to the specified lag-time L. The row number R2 is not necessarily an integer and the two queue values (V6 and V7) in the rows R1 and R3 closest to R2 can be used to interpolate/extrapolate the queue value at L. For example:

1) What is the approximate value of variable 16 (queue 2) at a lag-time L=1,16 hours earlier in the simulation?

The value required is at the row number.

R2=3+1,6/0,6-5/6 = 4,833

of array E2 above. R2 (L3940) equals the start row no. of queue 2 plus the time-lag, i.t.o. the number of rows, less the elapsed time, i.t.o. rows, since the last value was stored in E2. The queue values at row 4 (V6=77&R1=4) and row 5(V7=80&R3=5) can now be used to interpolate the value at row 4,833.

The interpolated value (L4000)

FNQ = V6+(V7-V6)(R2-R1)/(R3-R1)
= 77+(80-77)(4,833-4)/(5-4)
= 79,50

is returned to MODEL.

2) As for 1), but the lag-time L=0,2 hours?

R2=3+0,2/0,6-5/6 = 2,50

which is less than the first row number of queue number 2 and therefore rules out the possibility of interpolation. The first two queue values in rows 3 and 4 are used to extrapolate a value for row

2,50 (R1, R2, V6 and V7 equal 3, 4, 75 and 77 respectively). The value required

FNQ=75+(77-75)(2,5-3)/(4-3)=74,0

Output. An error message will result if the queue number specified does not exist or the lag-time is excessive.

USE TABLE (L4020)

Input and Output. None

Processing. This module is called by MODEL to interpolate a value from a table T5 for a given time X. The time is assumed to be the current simulation time T. The table interpolation time can be changed by MODEL if a time difference DO is specified. If, for example, T=15,2 hours and DO=-1,1 then a value will be interpolated from table T5 at time X=14,1(L4050). On exist from this module DO is set equal to 0(L4340) therefore MODEL must specify DO each time USE TABLE is called if a time other than the simulation time T is to be used for interpolation.

The process of interpolation from a table is illustrated by using the input data from the example in TABLE DATA. Fig.B2 shows the plot of observed values repeated for convenience. The matrices D1 and D3, and array D2 are not repeated here. The table number T5 equals 2.

R1, R2 and R3 are calculated (L4060 to 4100 and shown on the graph. If the time X is beyond the range of the table values (less than R1 or greater than R2+R3) then a zero table value FND is returned (L4110).

The next section of program (L4120 to 4170) determines the closest starting time S1 which is less than or equal to time X. If X equals any time between 24 and 59,99 hours, for example, then S1=24 hours (see fig.B2).

2,50 (R1, R2, V6 and V7 equal 3, 4, 75 and 77 respectively). The value required

FNQ=75+(77-75)(2,5-3)/(4-3)=74,0

Output. An error message will result if the queue number specified does not exist or the lag-time is excessive.

USE TABLE (L4020)

Input and Output. *None*

Processing. This module is called by MODEL to interpolate a value from a table T5 for a given time X. The time is assumed to be the current simulation time T. The table interpolation time can be changed by MODEL if a time difference DO is specified. If, for example, T=15,2 hours and DO=-1 then a value will be interpolated from table T5 at time X=14,1(L4050). On exist from this module DO is set equal to 0(L4340) therefore MODEL must specify DO each time USE TABLE is called if a time other than the simulation time T is to be used for interpolation.

The process of interpolation from a table is illustrated by using the input data from the example in TABLE DATA. Fig.B2 shows the plot of observed values repeated for convenience. The matrices D1 and D3, and array D2 are not repeated here. The table number T5 equals 2.

R1, R2 and R3 are calculated (L4060 to 4100 and shown on the graph. If the time X is beyond the range of the table values (less than R1 or greater than R2+R3) then a zero table value FND is returned (L4110).

The next section of program (L4120 to 4170) determines the closest starting time S1 which is less than or equal to time X. If X equals any time between 24 and 59,99 hours, for example, then S1=24 hours (see fig.B2).

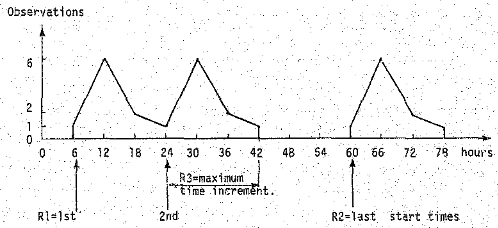


FIG.B2 Graph of Observation at Different Times

If the start time (S1=24) plus the maximum time increment (R3=18) is less than the time X=43 hours, say, then a zero table value FND is returned (L4180). If the time X is within the range of the table, X=37 hours, say, then L4190 to 4300 will find the times T6=36 and T7=42 hours which are on either side of X and for which there are corresponding table values, V6=2 and V7=1. The table value FND=1,75 is interpolated from V6 and V7 (L4310).

PLOT OUTPUT (L4350)

Input. Data stored on tape by STO PLOT is read into memory.

Processing. After simulation this module will use the variable values, stored on tape file P-DATA by STO PLOT, to plot separate graphs against time, of the variables specified in PLOT DATA. All graphs are plotted for the same time interval specified in PLOT TIMES. During simulation the time T, followed by all the values V of the specified variable (numbers 1, 4 and 9, say,) is stored on tape as shown:

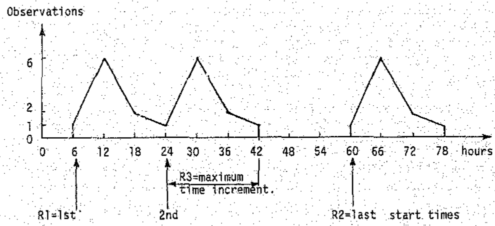


FIG.B2 Graph of Observation at Different Times

If the start time (S1=24) plus the maximum time increment (R3=18) is less than the time X=43 hours, say, then a zero table value FND is returned (L4180). If the time X is within the range of the table, X=37 hours, say, then L4190 to 4300 will find the times T6=36 and T7=42 hours which are on either side of X and for which there are corresponding table values, V6=2 and V7=1. The table value FND=1,75 is interpolated from V6 and V7 (L4310).

PLOT OUTPUT (L4350)

Input. Data stored on tape by STO PLOT is read into memory.

Processing. After simulation this module will use the variable values, stored on tape file P-DATA by STO PLOT, to plot separate graphs against time, of the variables specified in PLOT DATA. All graphs are plotted for the same time interval specified in PLOT TIMES. During simulation the time T, followed by all the values V of the specified variable (numbers 1, 4 and 9, say,) is stored on tape as shown:

T_0	V_0^1	V_0^4	V_0^9	T_1	V_1^1	V_1^4	V_1^9	T_2	//	T_n	V_n^1	V_n^4	V_n^9

STO PLOT enters values into a buffer which stores its contents onto tape whenever it is full. L4410 will empty the buffer onto tape to ensure that all values are on tape. If the simulation is done in reverse time sequence (negative time-steps), then TO will be greater than T1 in the above diagram. L4435 to 4460 defines the axes scales, defined in PLOT DATA and possibly modified in STO PLOT, so that time increases from left to right in all graphs.

Each graph number *J* is plotted in turn and involves starting at the beginning of the tape each time and reading every value into the buffer but selecting only the time T and the variable value for the particular graph. In the above example, L4465 to L4505 will search for the following points and plot them f. all the graphs in the order shown:

$(T_0,V_0^1),(T_1,V_1^1),(T_2,V_2^1),$ ----------- graph 1 (variable 1)

$(T_0,V_0^4),(T_1,V_1^4),(T_2,V_2^4)$ ----------- graph 2 (variable 4)

$(T_0,V_0^9),(T_1,V_1^9),(T_2,V_2^9)$ ----------- graph 3 (variable 9)

<u>Output</u>. If there are less than two data points then a message to this effect will be displayed instead of a graph (L4380,4400). If the graph consists of points with a constant variable value then the value will be displayed instead of the graph. The variable number, the variable and the time axes and the plotted points are first displayed on the screen and then copied onto paper. The minimum and maximum axes values and the unit spacing of the tic marks on the axes are listed below each graph. The total number of points is given after all graphs have been plotted.

MODEL (L4630)

Input. The user enters the model between L4640 and L7960 using only the symbols A(1), A(2), A(3) ----- for variables and F(1), F(2), F(3) ----- for differential equations.

Processing. This module is called by the particular solutions algorithm module selected in ALGORITHM. Each time MODEL is called, the values of certain of the variables (A(1), A(2) ------ etc.) will change and this, in turn, determines the values of the derivatives (F(1), F(2)---). For example,

F(6) = A(1)+F(4)*A(7)/(T-A(17)

where T is time. L7960 to L7990 is a permanent part of MODEL and will calculate a K-value (K(J,Z) for each of the NO differential equations)each time MODEL is called by the algorithm module. Depending on the algorithm, MODEL is called a number of times during each iteration of the simulation (Z= 1, 2 or 4 for EULER, MOD. EULER and 4R-KUTTA respectively).

Output. None

INITIAL (L8000)

Input, Processing and Output. This module is called by MASTER (L310) before the simulation begins. The user can therefore enter any program instructions into INITIAL to perform input, processing or output required before the simulation of the model. INITIAL is used only for initial input, processing and output for which the simulation program has no facilities such as variable names, simulation heading, calculation of parameters etc. The return-statement at the end of the module and the comment statements at the beginning are permanent features of the module.

FINAL (L9000)

Input, Processing and Output. This module has the same purpose as INITIAL above, but is called by MASTER (L515) after the simulation has been performed and the results printed.

APPENDIX C

SOURCE CODE OF THE SIMULATION PROGRAM

```
  5 ! --------MASTER
 10 CLEAR @ DISP "HEADING&NOTES-
    ",@ GOSUB 610 ! (PD)
 20 DISP "CONTINUOUS SIMULATION
    OF A      SYSTEM DESCRIBED B
    Y ORDINARY"
 30 DISP "DIFFERENTIAL EQUATIONS
     WITH      INITIAL BOUNDARY C
    ONDITIONS."
 35 DISP @ DISP "M.C.HOLTON-WITS
     UNIVERSITY."
 40 DISP "*********************
    **********"
 50 DISP "NOTE"
 60 DISP "1)Press key k1 WHILE E
    XECUTING     to get back into
     input mode"
 70 DISP "2)Space is provided in
    F&K for up  to 20 differenti
    al equations"
 80 DISP "3)Space is provided in
     A,A0&B      for up to 105 va
    riables in the  prgm"
 90 DISP "4)Space is provided in
     A1 for      printing up to 5
    0 variables at  each step"
 91 DISP "5)File space provided
    for<=2000   graph points,M h
    as space for    <=20graphs."
 92 DISP "6)Space is provided in
     D1 for      <=10 tables and
    in D2&D3 for a"
 93 DISP "  total of<=100start t
    imes and    <=100data points
    "
 94 DISP "7)Space is provided in
     E1&E2for    <=20queues and a
     total of 100   values"
 99 DISP "*********************
    **********"
100 ON KEY# 1 GOTO 580
110 REM DIM/DEF.ARRAYS&VARBLS
115 OPTION BASE 1
120 REAL A(105),B(105),K(20,4),F
    (20)
130 SHORT A0(105),D2(100),D3(100
    ,2),E2(100),E3(20,2),M(20,4)
140 INTEGER A1(50),C,C1,C9,D1(10
    ,4),D9,E,E0,E1(20,3),E9,F0,F
    1,F2,F3,J,J1,J2,N0,N1,N2,N3
150 INTEGER P0,P,P9,S,S0,T5,V
170 DISP "DIFFERENTIAL EQUATIONS
    &VARIABLES" @ GOSUB 610 ! (P
    D)
200 DISP "dA(J)/dT=F(J).J=1 to";
220 INPUT N0
230 DISP
240 DISP "F(J)=func(T,A(1)..A(";
    N0;")..A(N1))"
250 DISP USING 260 ; N0+1,N0
```

```
260 IMAGE "EXTRA VARIABLES ARE A
    (",DD,"),.A(N1)(IF NO EXTRAS
    ,ENTER N1=",DD,") N1=?"
280 INPUT N1
290 NORMAL @ Z1,F0,F1,F2,F3=0
292 FOR Z2=1 TO 11
294 IF Z2=2 OR Z2=3 THEN 302
296 IF Z2=5 AND F0=0 THEN 302
298 IF Z2=7 AND F1=0 THEN 302
300 DISP @ GOSUB 700
302 NEXT Z2
304 Z1=1 @ GOSUB 700
310 GOSUB 8000
315 ! INITIALIZE COUNTERS
320 T=T0 ! INITIALIZE CLOCK
360 IF F1=0 THEN 450 ! NO PLOT
370 C1=1 ! for STO PLOT
380 IF (T4-T3)*H>0 THEN 400
390 DISP "ERROR-Plot from";T3;"t
    o";T4;"but step size=";H @ G
    OTO 580
400 P0=INT((T4-T3)/((C8-1)*H)+.9
    5) @ P0=MAX(P0,1)
405 ASSIGN# 1 TO "P-DATA"
410 IF T3=T0 THEN 430
420 P=P0-1 @ GOTO 450
430 P=0 @ GOSUB 3560
440 C1=2
450 IF F3=0 THEN 490 ! NO QUEUES
460 E0=INT(E7/(E8-E9)/ABS(H)+.99
    ) @ E0=MAX(E0,1)
470 GOSUB 3740
480 E=0
490 IF F0=0 THEN 502 ! noPRT OUT
    PUT
493 IF (T8-T2)*H>0 THEN 496
494 DISP "ERROR-Print from";T2;"
    to";T8;"but step size=";H @
    GOTO 580
496 DISP USING 497
497 IMAGE 32("*")
498 DISP "INITIAL VALUE(S):";@ G
    OSUB 3670
500 IF T2=T0 THEN S=0 ELSE S=S0-
    1 ! OUTPT FRM START ELSE NOT
502 IF (T9-T0)*H>0 THEN 506
504 DISP "ERROR-Execute from";T0
    ;"to";T9;"but step size=";H
    @ GOTO 580
506 GOSUB 2930
508 DISP USING 497
510 IF F1=1 THEN GOSUB 4350
515 GOSUB 9000
520 NORMAL @ DISP "ANO[illegible] [illegible]UN(Y
    orN)";
540 INPUT I$
550 IF I$="N" THEN [illegible]
580 NORMAL @ GOSUB [illegible]
590 GOTO 310 ! RUN AG[illegible]
600 END
```

```
610 ! ----------PRT/DISP
630 NORMAL
640 DISP "Print or Display(PorD)
    ";
660 INPUT I$
670 IF I$="D" THEN 690
675 IF I$#"P" THEN 640
680 PRINT ALL
690 DISP @ RETURN
700 ! ----------INPUT
710 IF Z1=0 THEN 1000
790 DISP @ DISP "1)CHANGE ALL IN
    IT. VALUES"
800 DISP "2)RE-USE INIT.VARIABLE
    S&QUEUES"
820 DISP "3)CHANGE SOME VARIABLE
     VALUES"
830 DISP "4)CHANGE PRINT LIST"
840 DISP "5)CHANGE PRINT TIMES"
850 DISP "6)CHANGE PLOTTING DATA
    "
860 DISP "7)CHANGE TIMES FOR PLO
    TTING"
870 DISP "8)CHANGE EXECUTION TIM
    ES"
880 DISP "9)CHANGE ALGORITHM"
890 DISP "10)CHANGE FUNCTION TAB
    LES"
900 DISP "11)CHANGE QUEUE PARAME
    TERS"
910 DISP "12)EXECUTE PROGRAM"
960 DISP "(LAST OPTION";Z2;")" @
     DISP "ENTER NEW OPTION 1TO
    12";@ INPUT Z2
990 IF Z2>0 AND Z2<12 THEN 1000
993 IF Z2=12 THEN 1010
996 BEEP 100,100 @ GOTO 960
1000 ON Z2 GOSUB 1030,1110,1180,
     1280,1450,1600,1690,1740,17
     90,1830,2380
1005 NORMAL @ IF Z1>0 THEN 1020
     ELSE 790
1010 DISP "OUTPUT OF RESULTS-" @
      GOSUB 610
1020 RETURN
1030 ! ----------1)INIT VALS
1050 DISP "ENTER THE INITIAL VAL
     UE FOR EACH VARIABLE"
1055 GOSUB 2921 @ IF F4=1 THEN R
     ESTORE 8800
1056 GOSUB 610
1060 FOR J=1 TO N1
1070 DISP "A0(";J;")";
1090 IF F4=2 THEN INPUT A0(J) EL
     SE READ A0(J)@ DISP A0(J)
1100 NEXT J @ GOTO 1130
1110 ! ----------2)RE-INITIALISE
1120 ! NOTE-2) IS PART OF1)
1124 IF Z1=0 THEN 1140
1125 GOSUB 610 @ DISP "VARIABLES
      SET TO INITIAL VALUES."
```

```
1130 IF F3=1 THEN GOSUB 2860
1140 FOR J=1 TO N1
1150 A(J),B(J)=A0(J)
1160 NEXT J
1170 RETURN
1180 ! -----------3)CHANGES
1200 DISP "HOW MANY CHANGES";
1210 INPUT N3
1220 DISP "ENTER VARIABLE No.AND
      ITS VALUE."
1225 GOSUB 610
1230 FOR J=1 TO N3
1240 DISP "(No.,VAL.)";
1250 INPUT A3,A(A3)
1255 A0(A3),B(A3)=A(A3)
1260 NEXT J
1270 RETURN
1280 ! -----------4)PRT LIST
1300 F0=0 ! FLAG-NO(PRT)
1310 DISP "PRINT ANYTHING(YorN)"
     ;
1320 INPUT I$
1330 IF I$="N" T)      140
1340 F0=1
1343 GOSUB 2921        F4=1 THEN R
     ESTORE 88
1345 GOSUB 610
1350 DISP USING 1360 ; N1
1360 IMAGE "HOW MANY VARIABLES",
     /,"A(1)..A(",K,")MUST BE PR
     INTED"
1370 IF F4=2 THEN INPUT N2 ELSE
     READ N2@ DISP N2
1380 DISP USING 1390 ; N2,N1
1390 IMAGE "ENTER THE ",DD," NUM
     BER(S) BETWEEN",/,"1 AND ",
     K," IN ANY ORDER."
140  FOR J=1 TO N2
1410 DISP "(NO.";J;")";
1420 IF F4=2 THEN INPUT A1(J) EL
     SE READ A1(J)@ DISP A1(J)
1430 NEXT J
1440 RETURN
1450 ! -----------5)PRT TIMES
1455 DISP "OUTPUT TIMES-" @ GOSU
     B 610
1460 DISP "PRINT OUTPUT FROM T=T
     2 TO T8    EVERY S0 TIME ST
     EPS"
1470 DISP "(T2,T8,S0)";
1480 INPUT T2,T8,S0
1490 RETURN
1500 ! -----------6)PLOT DATA
1520 F1=0 ! FLAG-NO(PLT)
1530 DISP "PLOT ANYTHING(YorN)";
1540 INPUT I$
1550 IF I$="N" THEN 1680 ! RETUR
     N
1560 F1=1 ! FLAG ON
1565 GOSUB 2921 @ IF F4=1 THEN R
     ESTORE 8870
```

```
1570 GOSUB 610
1580 DISP "NO.PLOTS,MAX NO.PTS.(
     See note 5)";
1590 IF F4=2 THEN INPUT P9,C9 EL
     SE READ P9,C9@ DISP P9;C9
1600 DISP "UNIT SPACING ALONG T-
     AXIS";
1610 IF F4=2 THEN INPUT X1 ELSE
     READ X1@ DISP X1
1620 DISP "FOR EACH VARIABLE TO
     BE PLOTTED ENTER:"
1630 DISP "V-VARIABLE NO.,MN-MIN
     .VALUE,MX- MAX.VALUE,U-UNIT
      SPACING ALONG  VARIABLE AX
     IS"
1640 FOR J=1 TO P9
1650 DISP "(V,MN,MX,U)";
1660 IF F4=2 THEN INPUT A2,A3,A4
     ,A5 ELSE READ A2,A3,A4,A5@
     DISP A2;A3;A4;A5
1661 M(J,1)=A2 @ M(J,2)=A3 @ M(J
     ,3)=A4 @ M(J,4)=A5
1670 NEXT J
1680 RETURN
1690 ! ------------7)PLOT TIMES
1695 DISP "PLOT TIMES-" @ GOSUB
     610
1700 DISP "PLOT SPECIFIED VARIAB
     LES FROM   T=T3 TO T4,USING
      ABOUT C8 PTS."
1710 DISP "(T3,T4,C8)";
1720 INPUT T3,T4,C8
1730 RETURN
1740 ! ----------8)SIMUL.TIMES
1745 DISP "EXECUTION TIMES-" @ G
     OSUB 610
1750 DISP "EXECUTE FROM T=T0 TO
     T9 WITH TIME STEP H"
1760 DISP "(T0,T9,H)";
1770 INPUT T0,T9,H
1780 RETURN
1790 ! ------------9)ALGORITHM
1795 DISP "ALGORITHM-" @ GOSUB 6
     10
1800 DISP "EULER,MOD.EULER OR 4T
     H ORDER    RUNGE-KUTTA(1,2o
     r3)";
1810 INPUT F2@ IF F2<1 OR F2>3 T
     HEN 1800
1820 RETURN
1830 ! ---------10)TABLE DATA
1835 D0=0 ! forUSE TABLE
1840 DISP "USE TABLES(YorN)";@ I
     NPUT I$
1843 IF I$="N" THEN 2370
1845 IF I$#"Y" THEN 1840
1846 RESTORE 8900 @ GOSUB 2921 @
      GOSUB 610
1847 IF F4=1 OR Z1=0 THEN 1890
1850 DISP "ENTER/CHANGE NO.OF TA
     BLES,NO.OF TIMES EACH USED,
     OR NO.OF DATA   PAIRS(YorN)
     ";
```

```
1860 INPUT I$
1870 IF I$="N" THEN 2040
1890 DISP "HOW MANY TABLES(see n
     ote 6)"
1900 IF F4=2 THEN INPUT D9 ELSE
     READ D9@ DISP D9
1920 DISP "FOR EACH TABLE ENTER
     2 VALUES:"
1930 DISP "NO.OF TIMES TABLE WIL
     L BE USED, NO. OF DATA PAIR
     S IN TABLE(See  note 6)"
1940 FOR J=1 TO D9
1950 DISP "(TABLE";J;")";
1960 IF F4=2 THEN INPUT A4,A5 EL
     SE READ A4,A5@ DISP A4;A5
1961 D1(J,1)=A4 @ D1(J,3)=A5
1970 NEXT J
1980 D1(1,2),D1(1,4)=1
1990 FOR J=1 TO D9-1
2000 ! DEF START ROW NO'S OF  D2
     &D3 IN D1
2010 D1(J+1,2)=D1(J,2)+D1(J,1)
2020 D1(J+1,4)=D1(J,4)+D1(J,3)
2030 NEXT J
2035 IF F4=1 OR Z1=0 THEN 2070
2040 DISP "ENTER/CHANGE T-VALUES
     WHEN TABLES ARE TO BE START
     ED(YorN)";
2050 INPUT I$
2060 IF I$="N" THEN 2200
2070 J1=0
2080 FOR J=1 TO D9
2090 DISP "TABLE";J
2100 A3=-1 ! DUMMY VAL.
2110 FOR J2=1 TO D1(J,1)
2120 J1=J1+1 ! ROW OF D2
2130 DISP J2;"(Tstart)";
2140 IF F4=2 THEN INPUT D2(J1) E
     LSE READ D2(J1)@ DISP D2(J1
     )
2150 IF D2(J1)>A3 THEN 2170 ! LA
     TEST START TIME>PREV.
2160 F4=2 @ DISP "ERROR-start va
     lues must increase" @ GOTO
     2040
2170 A3=D2(J1) ! NEW PREV.
2180 NEXT J2
2190 NEXT J
2195 IF F4=1 OR Z1=0 THEN 2230
2200 DISP "ENTER/CHANGE DATA PAI
     RS(YorN)";
2210 INPUT I$
2220 IF I$="N" THEN 2370
2230 DISP "ENTER T-INCREMENT AFT
     ER TABLE   START VALUE,TABL
     E VALUE"
2240 J1=0
2250 FOR J=1 TO D9
2260 DISP "TABLE";J
2270 A3=-1 ! DUMMY VAL.
```

```
2280 FOR J2=1 TO D1(J,3)
2290 J1=J1+1
2300 DISP J2;"(T-inc,TAB)";
2310 IF F4=2 THEN INPUT A4,A5 EL
     SE READ A4,A5@ DISP A4,A5
2311 D3(J1,1)=A4 @ D3(J1,2)=A5
2320 IF D3(J1,1)>A3 THEN 2340 !
     LATEST INCR.>PREV.
2330 F4=2 @ DISP "ERROR-Tincreme
     nts must increase" @ GOTO 2
     230
2340 A3=D3(J1,1) ! NEW PREV.
2350 NEXT J2
2360 NEXT J
2370 RETURN
2380 ! -----------11)Q DATA
2395 L=0 ! forUSE Q
2400 F3=0 ! FLAG-NO(I/Q)
2410 DISP "USE QUEUES(YorN)";
2420 INPUT I$
2430 IF I$="N" THEN 2920 ! RETRN
2435 IF I$#"Y" THEN 2410
2440 F3=1
2445 GOSUB 2921 @ IF F4=1 THEN R
     ESTORE 8970
2447 GOSUB 610 @ IF F4=1 OR Z1=0
      THEN 2490
2450 DISP "ENTER/CHANGE NO.OF QU
     EUES,VAR-  IABLE NO'S,OR LA
     G TIMES(YorN)";
2460 INPUT I$
2470 IF I$="N" THEN 2580
2490 DISP "HOW MANY QUEUES(See n
     ote 7)";
2500 IF F4=2 THEN INPUT E9 ELSE
     READ E9@ DISP E9
2510 DISP "FOR EACH QUEUE ENTER
     VARIABLE NO,LAG TIME"
2520 E7=0
2530 FOR J=1 TO E9
2540 DISP "Q NO.";J;"(A(?),LAG)"
     ;
2550 IF F4=2 THEN INPUT A4,A5 EL
     SE READ A4,A5@ DISP A4;A5
2551 E1(J,1)=A4 @ E3(J,2)=A5
2560 E7=E7+E3(J,2) ! TOT.LAG-T
2570 NEXT J
2575 IF F4=1 OR Z1=0 THEN 2620
2580 DISP "ENTER/CHANGE SPACE AL
     LOCATION   FOR QUEUES(YorN)
     ";
2590 INPUT I$
2600 IF I$="N" THEN 2780
2620 DISP "TOTAL SPACES FOR QUEU
     E VALUES    (See note 7)";
2630 IF F4=2 THEN INPUT E8 ELSE
     READ E8@ DISP E8
2640 REM 1ST ROW NO.OF E2 FOR EA
     CH Q IS STO'D IN E1.NO.Q-VA
     LS IS PROP.TO LAG-T'S &STO'
     D IN E1
```

```
2650 E1(1,3)=1
2660 FOR J=1 TO E9 ! EACH Q
2670 E1(J,2)=INT(E3(J,2)*E8/E7+.
     5)
2675 IF E1(J,2)=1 THEN E1(J,2)=2
     ! NEED AT LEAST 2 PTS.
2680 IF J<E9 THEN E1(J+1,3)=E1(J
     ,3)+E1(J,2)
2690 NEXT J
2700 DISP "SPACE FOR QUEUES 1 TO
     ";E9;":"
2710 FOR J=1 TO E9
2720 DISP ;E1(J,2);
2730 NEXT J
2740 E8=E1(E9,3)+E1(E9,2)-1
2750 DISP "TOTAL=";E8;".(see not
     e 7)"
2755 IF F4=2 THEN 2760
2756 DISP "IF ALLOCATIONS UNSATI
     SFACTORY  CORRECT LATER.(p
     ress CONT key)" @ PAUSE @ G
     OTO 2810
2760 DISP "ALLOCATIONS OK(YorN)"
     ;@ INPUT I$
2770 IF I$="N" THEN 2620
2780 DISP "ENTER/CHANGE INITIAL
     VALUES TO  BE STORED IN EAC
     H QUEUE(YorN)";
2790 INPUT I$
2800 IF I$="N" THEN 2920
2810 DISP "INITIALIZE EACH QUEUE
      TO 1 VALUE"
2820 FOR J=1 TO E9
2830 DISP "Q NO.";J;"A(";E1(J,1)
     ;")";
2840 IF F4=2 THEN INPUT E3(J,1)
     ELSE READ E3(J,1)@ DISP E3(
     J,1)
2850 NEXT J
2860 ! ENTRY FROM 2)TO RE-USE IN
     IT Q-VALS.
2870 FOR J=1 TO E9
2880 FOR J1=E1(J,3) TO E1(J,3)+E
     1(J,2)-1
2890 E2(J1)=E3(J,1)
2900 NEXT J1
2910 NEXT J
2920 RETURN
2921 ! -----------DATA/PROMPT
2923 DISP "1)USE PREDEFINED DATA
     " @ DISP "2)KEYBOARD INPUT"
      @ DISP "(1or2)";@ INPUT F4
2924 IF F4#1 AND F4#2 THEN 2923
2925 RETURN
2930 ! ------------MAIN
2950 FOR C=0 TO (T9-T0)/H-1 ! NO
     .OF TIME INCREMENTS
2960 T1=T0+C*H ! Tn
2970 ON F2 GOSUB 3175,3199,3340
     ! ALGORITHMS
```

```
2980 ! STO Q
2981 IF F3=0 THEN 3030
2990 E=E+1
3000 IF E<E0 THEN 3030
3010 GOSUB 3740
3020 E=0
3030 ! STO PLOT
3031 IF F1=0 THEN 3110
3040 IF (T3-T)*SGN(H)>0 OR (T-T4
     )*SGN(H)>0 THEN 3110
3050 P=P+1
3060 IF P<P0 THEN 3110
3070 IF C1>C9 THEN 3110
3080 GOSUB 3560
3090 P=0
3100 C1=C1+1 ! PLOT COUNTER
3110 ! PRT OUTPUT
3111 IF F0=0 THEN 3170
3120 IF (T2-T)*SGN(H)>0 OR (T-T8
     )*SGN(H)>0 THEN 3170
3130 S=S+1 ! PRT COUNTER
3140 IF S<S0 THEN 3170
3150 GOSUB 3670
3160 S=0
3170 NEXT C
3171 RETURN
3175 ! --------------EULER
3180 T=T1 ! time Tn
3181 Z=1
3182 GOSUB 4630 ! K(J,1)
3184 FOR J=1 TO N0
3186 A(J),B(J)=B(J)+K(J,1)
3188 NEXT J
3189 T=T+H
3190 RETURN
3199 ! -----------MOD.EUL
3210 T=T1 ! act.time Tn
3220 Z=1 ! FOR K(J,Z)
3230 GOSUB 4630 ! K(J,1)
3240 T=T+H ! Tn+H
3250 Z=2
3260 FOR J=1 TO N0
3270 A(J)=B(J)+K(J,1)
3280 NEXT J
3290 GOSUB 4630 ! K(J,2)
3300 FOR J=1 TO N0
3310 A(J),B(J)=B(J)+(K(J,1)+K(J,
     2))/2 ! A(J)n+1
3320 NEXT J
3330 RETURN
3340 ! -----------4R-KUTTA
3360 T=T1 ! timeTn
3370 Z=1
3380 GOSUB 4630 ! K(J,1)
3390 T=T+H/2 ! Tn+H/2
3400 FOR Z=2 TO 3
3410 FOR J=1 TO N0
3420 A(J)=B(J)+K(J,Z-1)/2
3430 NEXT J
3440 GOSUB 4630 ! K(J,Z)
```