UNITED STATES

DEPARTMENT OF THE INTERIOR

GEOLOGICAL SURVEY

MULTI-EVENT URBAN RUNOFF QUALITY MODEL

By William M. Alley and Peter E. Smith

GEOLOGICAL SURVEY OPEN FILE REPORT 82-764

UNITED STATES DEPARTMENT OF THE INTERIOR

JAMES G. WATT, Secretary

GEOLOGICAL SURVEY

Dallas L. Peck, Director

For further information write to:

Chief Hydrologist
U.S. Geological Survey
410 National Center
Reston, Virginia 22092

CONTENTS

FIGURES

FIGURES

TABLE

GLOSSARY

distributed-parameter model. -- A model in which parameter values may vary spatially.

impervious areas, effective. -- Impervious areas from which runoff drains directly to channels without flowing over pervious surfaces.

impervious areas, noneffective. -- Impervious areas that drain to pervious areas, such as roofs that drain onto lawns.

inches of runoff. -- A volume of runoff expressed as a depth of water in inches over the area of the watershed. Unless otherwise noted, the area of the watershed used in computing the inches of runoff is the effective impervious area.

land-use type. -- A designation for an area on the watershed land surface that is identified with a unique set of impervious-area runoff-quality parameters (K_1-K_4).

lumped-parameter model. -- A model in which spatial variations in model parameters are ignored.

CONVERSION FACTORS

Multiply inch-pound unit	By	To obtain SI unit
inch	25.4	millimeter
inch per hour	25.4	millimeter per hour
foot (ft)	0.3048	meter
foot per hour (ft/hr)	0.3048	meter per hour
acre	0.4047	hectare
square mile	2.590	square kilometer
acre-foot (acre-ft)	0.001233	cubic hectometer
cubic foot per second (ft^3/s)	0.02832	cubic meter per second
pound (avoirdupois)	0.4536	kilogram
pound per acre	1.121	kilogram per hectare
pound per day	0.4536	kilogram per day
ton (short, 2000 pounds)	0.9072	metric ton

MULTI-EVENT URBAN RUNOFF QUALITY MODEL

by William M. Alley and Peter E. Smith

ABSTRACT

A computer model is presented for simulating the quality of surface runoff from urban watersheds. The model can simulate impervious area, pervious area, and precipitation contributions to runoff quality as well as the effects of street sweeping and (or) detention storage. Within-storm variations of runoff quality are simulated for user-specified storm-runoff periods. Between these storms, a daily accounting of the accumulation and washoff of water-quality constituents on effective impervious areas is maintained. The time step of the within-storm simulations can range from 1 to 60 minutes.

The model can be operated as a lumped-parameter model or as a distributed-parameter model. As a lumped-parameter model, no spatial variations in model parameters are accounted for, and input to the model requires flow hydrographs only at the outlet of the watershed. The outlet hydrographs can be either observed or simulated. As a distributed-parameter model, the model requires flow hydrographs at many points in the watershed, as defined by basin segmentation. These hydrographs will normally be simulated by the Distributed Routing Rainfall-Runoff Model.

This report includes a presentation of the theory and limitations of the model, as well as a program listing, instructions for running the program, and example simulations.

INTRODUCTION

The effects of urban runoff on water quality are a concern in many metropolitan areas of the United States. Many studies have recently highlighted a need for improved means of assessing urban nonpoint sources of pollution. As a tool to assist in meeting this need, a computer model has been developed to simulate the quality of rainfall runoff from urban areas. This model (referred to as DR_3M-QUAL) can be linked with the Distributed Routing Rainfall-Runoff Model (DR_3M) documented by Alley and Smith (1982). Major features of DR_3M-QUAL include:

1. Simulation of impervious area, pervious area, and precipitation contributions to runoff quality.

2. Detailed simulation of the water quality of user-specified storm-runoff periods and daily accounting of the accumulation and washoff of water-quality constituents on effective impervious surfaces between storms. The time step of detailed simulations can range from 1 to 60 minutes.

3. The model can be operated as a lumped-parameter model or as a distributed-parameter model. As a lumped-parameter model, no spatial variations in model parameters are accounted for and input to the model

requires flow hydrographs only at the outlet of the watershed. The outlet hydrographs can be either observed or simulated. As a distributed-parameter model, the model requires flow hydrographs at many points in the watershed as defined by basin segmentation. These hydrographs will normally be simulated by using DR_3M.

4. The model can be used to simulate the effects of street sweeping and (or) detention storage on runoff quality.

5. Measured runoff and runoff-quality data can be input to the model for calibration purposes. Model calibration is facilitated by graphical output.

The main part of this report will describe the theory behind DR_3M-QUAL as well as some of its limitations and potential applications. A program listing, instructions for using the program, and example simulations are included in the attachments.

ACKNOWLEDGMENTS

The authors are grateful to Andy Ward of the University of Kentucky for several fruitful discussions on detention storage and use of the DEPOSITS model as a prototype for the detention storage part of DR_3M-QUAL.

SOURCES OF WATER-QUALITY CONSTITUENTS

DR_3M-QUAL operates on two time intervals. The model provides detailed simulation of the quality of storm-runoff events during days for which short-time interval data are input to the program. These days are referred to as "unit days." Between unit days, the model uses daily precipitation data to provide a continuous accounting of impervious-area land-surface loads on a daily basis.

DR_3M-QUAL considers three sources of water-quality constituents: Impervious-area runoff, pervious-area runoff, and precipitation.

Impervious-Area Runoff Quality

The accounting of water-quality constituents on the impervious-areas includes constituent accumulation and washoff.

Constituent Accumulation

Several studies have suggested that the rate of accumulation of water-quality constituents on urban impervious surfaces is nonlinear and that there is a limit to the amount of constituents that can accumulate between storms, regardless of the length of dry period. Data collected by Sartor and Boyd (1972) and Pitt (1979) suggest that the accumulation rate is largest for several days after a period of street cleaning or rainfall, and then the rate decreases and approaches zero. Two possible explanations for this are that wind, either natural or due to vehicular traffic, may resuspend constituents and deposit them on pervious and noneffective impervious areas,[1/] or that the natural biological and chemical decay of constituents may limit buildup.

Constituent accumulation on impervious surfaces can be modeled as:

$$\frac{dL}{dt} = K - K_2 L \tag{1}$$

where L is the amount of the constituent on the effective impervious area, in pounds; K is a constant rate of constituent deposition, in pounds per day; K_2 is a rate constant for constituent removal, in day^{-1}; and t is time, in days. The parameter K_2 can theoretically be assumed to account for losses due to wind and also the biological and chemical decay of some constituents. Integration of equation (1) yields:

$$L = K_1 [1 - \exp(-K_2 T)] \tag{2}$$

where $K_1 = K/K_2$ is the maximum amount of the constituent which can accumulate on the effective impervious area, in pounds; and T is accumulation time, in days.

Traditionally, equation 2 has been derived with the assumption that effective impervious surfaces were completely clean at the end of the previous period of street sweeping or storm runoff. In order to eliminate this assumption, T is redefined as equivalent accumulation time and set equal to:

$$T = t + t_e \tag{3}$$

where t is the time since last period of street sweeping or storm runoff, and t_e is the time required for a constituent load on effective impervious surfaces to accumulate equal to that at the end of the last period of street sweeping or storm runoff, assuming initially clean urban impervious surfaces. The variable t_e is computed as:

$$t_e = -\frac{1}{K_2} \ln \left(1 - \frac{L_e}{K_1}\right) \tag{4}$$

where L_e is the amount of the constituent on the effective impervious surfaces in pounds at the end of the last period of street sweeping or storm runoff. Equation 4 is derived from equation 2 by substituting L_e for L and solving for $T = t_e$. Figure 1 illustrates the exponential constituent accumulation equations. The upper curve accounts for a residual amount of constituent remaining on effective impervious surfaces at the end of the last period of street sweeping or storm runoff, whereas the lower curve assumes no residual.

Constituent Washoff

Constituent washoff from effective impervious areas can be simulated using an exponential washoff equation:

1/ Definitions of selected terms including effective and noneffective impervious area are included in the glossary.

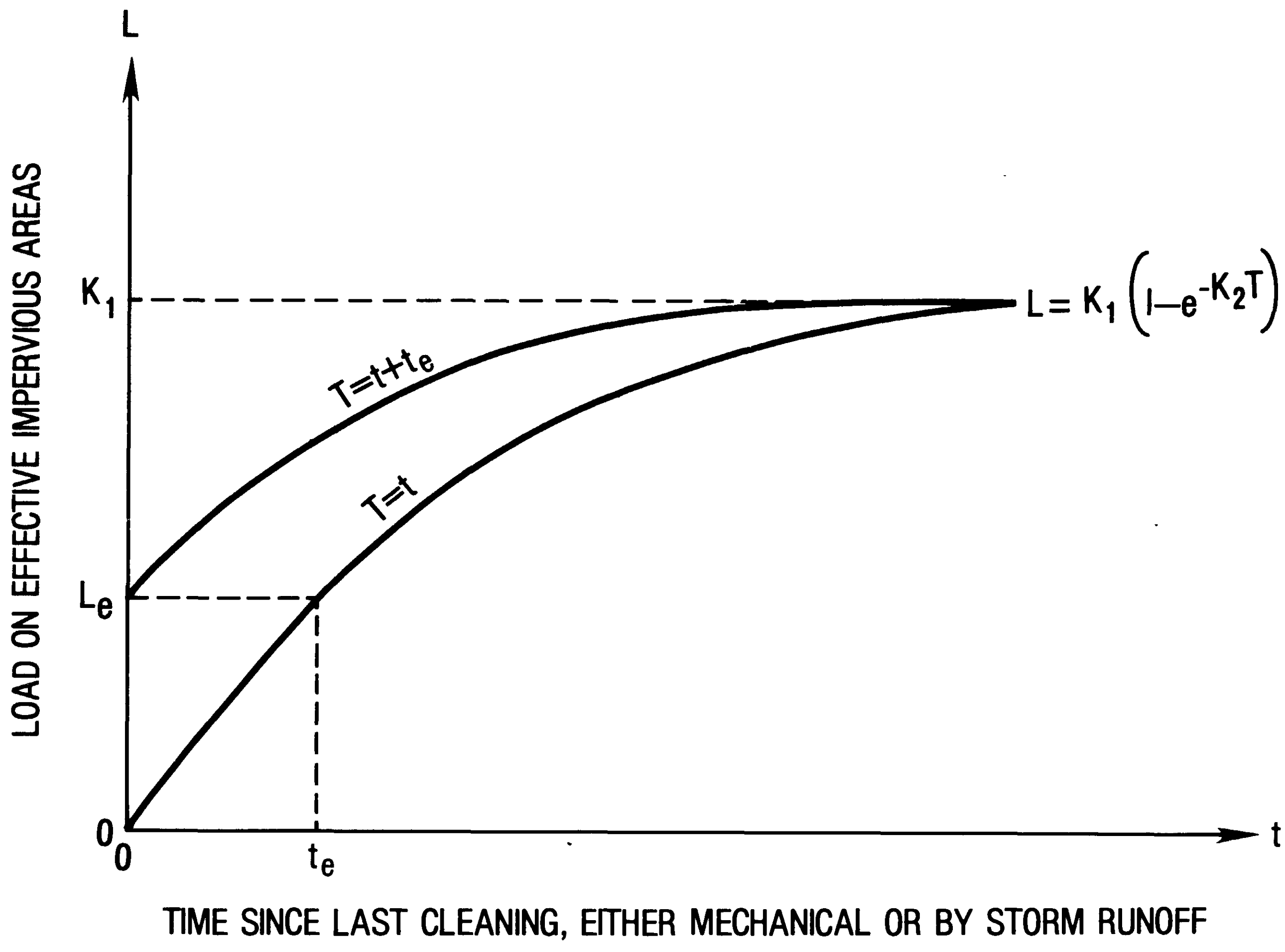


Figure 1.--Constituent accumulation on effective impervious areas.

$$W = L_o[1 - \exp(-K_3 R \Delta t)] \tag{5}$$

where W is the amount of constituent removed from effective impervious surfaces during a time step, in pounds; L_o is the amount of constituent on effective impervious surfaces at the beginning of the time step, in pounds; K_3 is the washoff coefficient in inches^{-1}; R is the runoff rate, in inches per hour;2/ and Δt is the time step, in hours. Constituent concentrations are determined by dividing W by $AR\Delta t$ where A is the effective impervious area.

Equation 5 is the same equation used to simulate constituent washoff in several well-known urban runoff models such as SWMM (Metcalf and Eddy, Inc., and others, 1971) and STORM (U.S. Army Corps of Engineers, 1976). The exponential washoff is derived by assuming that the rate of constituent washoff on effective impervious surfaces is proportional to the amount remaining. That is,

$$\frac{dL}{dt} = -CL \tag{6}$$

where C is the rate constant and is assumed to vary in direct proportion to the rate of runoff (that is, $C=K_3R$). Because washoff by the exponential washoff equation (equation 5) is a function of the product of runoff intensity (R) and duration (Δt), the amount of constituents washed off during a storm is a direct function of the total volume of storm runoff. The equation does not account for the biological and chemical decay of constituents, all constituents are assumed conservative. Since the response time of most small watersheds is short, this assumption should not limit the applicability of the model. During simulation on a daily basis, the washoff of constituents is estimated with equation 5 using a Δt of 1 day and impervious-area runoff equal to daily precipitation minus impervious retention.

It will be seen in the next section on parameter estimation that it is very helpful in model calibration to present the results of equation 5 in the form of curves of cumulative dimensionless load for a storm versus cumulative dimensionless runoff volume. These curves are called "load characteristic curves" because they identify the "character" of the load distribution for a storm. Figure 2 shows four types of load characteristic curves representing four different concentration distributions over a storm hydrograph. A load characteristic curve is intimately related to a plot of storm concentration versus time. The slope of a load characteristic curve is dL/dV, which represents mass per unit volume of runoff or concentration. Thus, if the slope of a load characteristic curve is decreasing, so also are constituent concentrations, and conversely, if the slope is increasing constituent concentrations are increasing. For the special case of a simultaneous load characteristic curve, concentrations through the storm will be constant because the slope of the curve is constant.

2/ Inches of runoff refers to the volume of runoff in inches distributed over the effective impervious area.

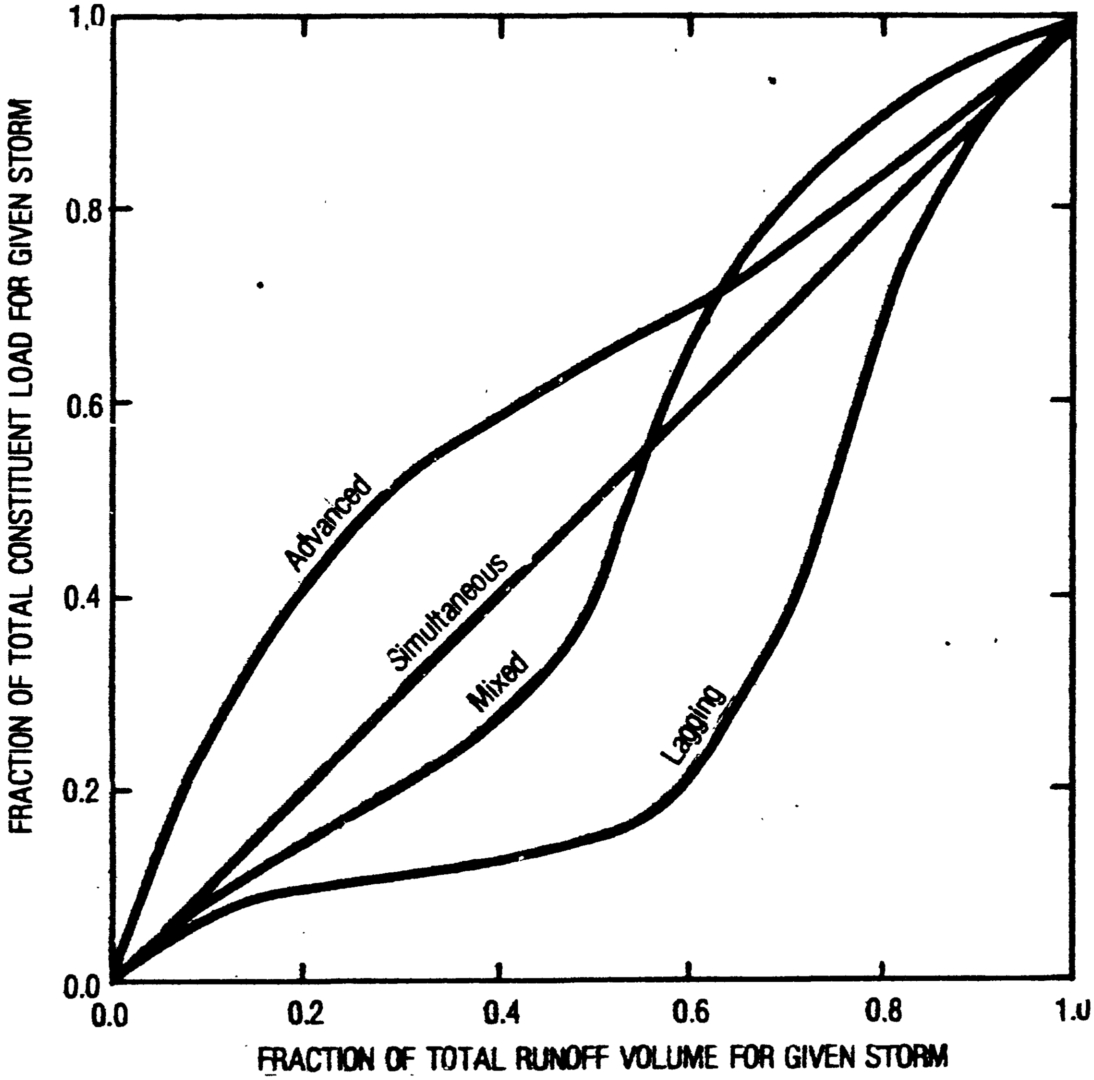


Figure 2.--Load characteristic curves for typical advanced, simultaneous, lagging, and mixed types of concentration distributions.

Load characteristic curves can be useful in determining storm-water management strategies. For example, collection of the first 20 percent of runoff volume from a storm would result in a 40 percent capture of the constituent load for the advanced-type characteristic curve shown in figure 2 but less than 10 percent capture of the constituent load for the lagging-type characteristic curve of figure 2. Ellis and Sutherland (1979) demonstrated that application of the exponential washoff equation on a lumped-parameter basis resulted in the following equation for load characteristic curves:

$$Y_F = \frac{[1-\exp(-K_3V_F)]}{[1-\exp(-K_3V_T)]} \tag{7}$$

where Y_F is the fraction of the total constituent load for a given storm (Y-axis in figure 2), V_F is the cumulative runoff volume (in inches) for the point on the load characteristic curve of interest, and V_T is the total storm-runoff volume (in inches).

Assuming V_T = 1.0 inch and substituting various K_3 values into equation 7 results in the family of curves shown in figure 3. Note that all curves in figure 3 are advanced-type curves with the slope always decreasing through the storm. Because the washoff equation is exponential, curves of this shape will be predicted. As a result the equation will always predict decreasing concentrations with increasing time through a storm. If the washoff equation is applied on a distributed basis, this same shape of curve will be observed from each overland flow plane, but the routing of constituents through a channel network may result in curves of any of the types shown in figure 2.

It should be further noted from figure 3 that the curvature of each load characteristic curve increases as the value of K_3 increases. Conversely, as the value of K_3 approaches zero, the load characteristic curve approaches a straight line. In a similar manner it can be shown that for a given value of K_3 the curvature of load characteristic curves increases as the total storm-runoff volume increases (Alley, 1981).

One important limitation of the washoff equation is that it does not account for the effects of runoff intensity on the amount of constituents available for transport from effective impervious surfaces. Experience has shown that at low runoff intensities it may be necessary to limit the amount available for transport of certain constituents, such as sediments and constituents associated with sediments. For this purpose the model contains an optional availability factor defined as:

$$AVAIL = K_4R \tag{8}$$

where R is runoff intensity in inches/hour and K_4 is an empirical coefficient which in general must be calibrated for a given constituent and land-use type using measured runoff-quality data. Constituent washoff is then simulated by multiplying equation 5 by the lesser of AVAIL or 1.0. Equation 8 is analogous to a similar set of equations contained in SWMM and STORM. The advantage of equation 8 lies in its simplicity. Its limitations include the fact that constituent availability relationships would be expected to change from storm to storm and within storms due to changes in particle-size distributions of constituents. The availability factor can prove useful in modeling,

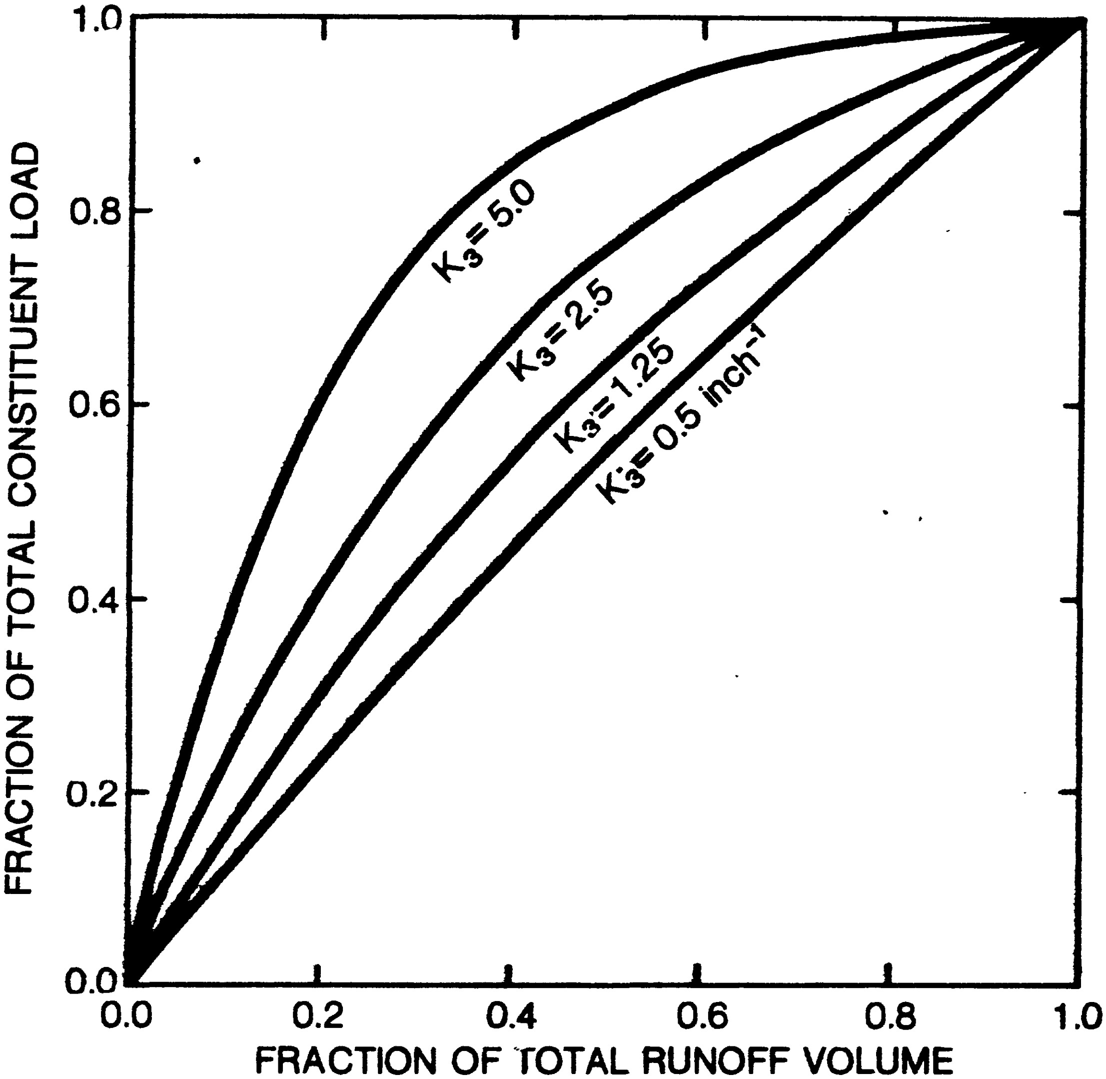


Figure 3.--Predicted load characteristic curves for various values of K_3 (V_T = 1.0 inch).

because when it is used to adjust equation 5 it becomes possible to predict mixed load characteristic curves.

Alternatively, the effects of runoff intensity on constituent washoff could be accounted for without using the availability factor by assuming the rate constant, C, in equation 6 is proportional to the square of runoff intensity. In this case the exponential washoff equation becomes

$$W = L_o[1 - \exp(K_{3*} R^2 \Delta t)] \quad (9)$$

Use of equation 9 as an alternative to equation 5 is included as an option in the model. However, note that K_{3*} is a negative number. This is to signify to the model that equation 9 rather than equation 5 is used for washoff. The units of K_{3*} in equation 9 are hr/inch^2.

Parameter Estimation for Lumped-Parameter Simulations

Estimates of K_1 to K_3 (and optionally, K_4) are required in order to apply DR$_3$M-QUAL. The parameters K_1 and K_2 could be determined from data obtained in a street-surface sampling program (Sarter and Boyd, 1972; Pitt, 1979). However, the values of K_1 and K_2 determined in this manner would not account for constituent loads on other effective impervious surfaces such as roofs and driveways, and the sampled material removed by sweeping, vacuuming, or flushing techniques may not be related to the amount of constituent available for transport by storm runoff. An alternate approach is to estimate these parameters using runoff-quality data collected at the watershed outlet. The following discussion pertains to estimating values of K_1 to K_4 for lumped-parameter simulations using measured runoff-quality data. Care should be exercised in using this approach because the outlet data represent the combined effects of accumulation, washoff, routing, and contributions from not only effective impervious areas but pervious areas and rainfall as well. In addition, effective impervious areas may include different land-use types with different accumulation rates. The procedures described herein are therefore intended only for application to small watersheds of uniform land use and for storm events with little pervious-area runoff. When these assumptions are not met, best-fit parameter values may differ substantially from "true" values. Note that in the following discussion it is assumed that equation 5 rather than equation 9 is used for washoff.

Inspection of equation 7 reveals that simulated load characteristic curves (for the assumption of exponential washoff and lumped-parameter simulations) are independent of the amount of a constituent (L_s) on the effective impervious areas at the start of the storm. Thus, simulated load characteristic curves are not affected by assumed values of the accumulation parameters K_1 and K_2. For this reason, Alley (1981) presented an optimization scheme for determining the best-fit value of K_3, the washoff coefficient, for a given constituent and set of storm events. This optimization scheme was based on fitting measured and simulated load characteristic curves. One can use this scheme to determine best-fit values of K_3 or a value for K_3 can be estimated from visual inspection of simulated and measured load characteristic curves. These curves can be obtained as part of the output from

DR_3M-QUAL. Increasing the value of K_3 will increase the curvature of simulated load characteristic curves and decreasing the value of K_3 will decrease their curvature. A value of 4.6 $inches^{-1}$ is a reasonable initial estimate of K_3.

A characteristic of storm-water quality sampling programs is that field or automatic sampling logistics result in the first sample of storm runoff being collected sometime after the start of the runoff and the last sample collected before the end of the runoff period. Often a considerable part of the total storm runoff may not be included in runoff occurring between the first and last sample. Thus, measured runoff loads often should be based only on the runoff occurring between the first and last sample. If measured concentrations are input to DR_3M-QUAL, then measured and simulated load characteristic curves are based on the sampled interval of the runoff.

Having selected a value of K_3 for a particular constituent, and assuming for the moment that the availability factor (equation 8) is not used, values for K_1 and K_2 can be determined using a technique presented by Alley and Smith (1981). This technique solves the following unconstrained optimization problem:

$$\min \quad Z = \sum_{i=1}^{N} \left[K_1[1-\exp\left(-K_2T(i)\right)] - L_s(i)\right]^2 \qquad (10)$$

where N is the number of storms included in the objective function, T(i) is the simulated equivalent accumulation time for the i^{th} storm in days, and $L_s(i)$ is the amount of constituent on effective impervious areas at the start of the i^{th} storm, in pounds. The value of $L_s(i)$ for each storm can be computed (Alley and Smith, 1981) as:

$$L_s(i) = \frac{\Delta L}{\exp(-K_3V_a)[1-\exp\left(-K_3(V_b-V_a)\right)]} \qquad (10a)$$

where ΔL is the measured runoff load occurring between the first and last sample in pounds, V_a is the cumulative storm runoff at the time of the first sample in inches over effective impervious area, and V_b is the cumulative storm runoff at the time of the last sample in inches over the effective impervious area. From the above definition, $L_s(i)$ can be interpreted as the amount of constituent on effective impervious areas at the start of the i^{th} storm that will yield a simulated washoff load equal to the measured load, when the given K_3 is used in the exponential washoff equation (equation 5). Values for $L_s(i)$ and T(i) are computed by DR_3M-QUAL and included as part of the model output for lumped-parameter simulations, if equation 5 is used without the availability factor.

Solving equation 10 is analogous to the problem of finding ultimate BOD (biochemical oxygen demand) and the first-order rate coefficient for the BOD reaction. In this analogy K_1 is equivalent to the ultimate BOD, K_2 is equivalent to the first-order rate coefficient, and $L_s(i)$ is equivalent to the amount of oxygen consumed after the equivalent of T(i) days. Thus, a number of programs are available for solving equation 10. These include a user's manual documented by Jennings and Bauer (1976) and a program described by Barnwell (1980).

In order to solve equation 10, it is first necessary to make an initial estimate of the values of K_1 and K_2. DR_3M-QUAL should then be run using these estimates of K_1 and K_2 to obtain estimates of T(i), i = 1, . . . , N. Usually, this step will be performed as part of the estimation of K_3. The user can then plot the values of $L_s(i)$ versus T(i), i = 1, . . . , N, and either "eyeball" an exponential accumulation curve through the points or use one of the programs previously cited. If the "eyeball" approach is used then K_1 is the asymptote of the curve. The parameter K_2 can be estimated using

$$K_2 = 2.303/T_{90} \tag{11}$$

where T_{90} is the equivalent accumulation time, in days, at which the land-surface load equals 90 percent of K_1 and is read from the eyeballed curve.

DR_3M-QUAL can then be run using the new estimates for K_1 and K_2 to generate new estimates for equivalent accumulation times [T(i), i = 1, . . . , N]. This procedure can be repeated until the changes in equivalent accumulation times from model run to model run have negligible effect on the simulation results. Experience thus far with the model indicates that usually no more than two iterations with the above procedure are required.

Three different situations can arise in performing the preceding analysis. Examples of these situations are shown in figure 4. If a plot of $L_s(i)$ versus T(i), i = 1, . . . , N appears to follow an exponential accumulation curve as in Figure 4A, then the above procedure can be used to estimate K_1 and K_2. If they appear to follow a linear accumulation curve as in figure 4B, then simple linear regression analysis can be used to fit a relationship of the form

$$L_s(i) = \hat{\beta} \cdot T(i) \tag{12}$$

where $\hat{\beta}$ is the regression coefficient. The value of K_2 should then be set at a very small number, say 0.001 and $K_1 = \hat{\beta}/K_2$. Alley and Smith (1981) present the details of this type of analysis.

Finally, if the plot of $L_s(i)$ versus T(i) appears as in figure 4C, then DR_3M-QUAL will not work satisfactorily on a lumped-parameter basis using equation 5 and the model should be applied on a distributed basis or an alternative model selected.

The availability factor (equation 8) has been left out of the above analysis and should be used with caution. However, it has been found that the exponential washoff equation can seriously overestimate constituent concentrations during the initial parts of storm-runoff periods when runoff intensities are very low. This situation can be remedied to a certain extent by judicious application of the availability factor. An availability factor is defined in

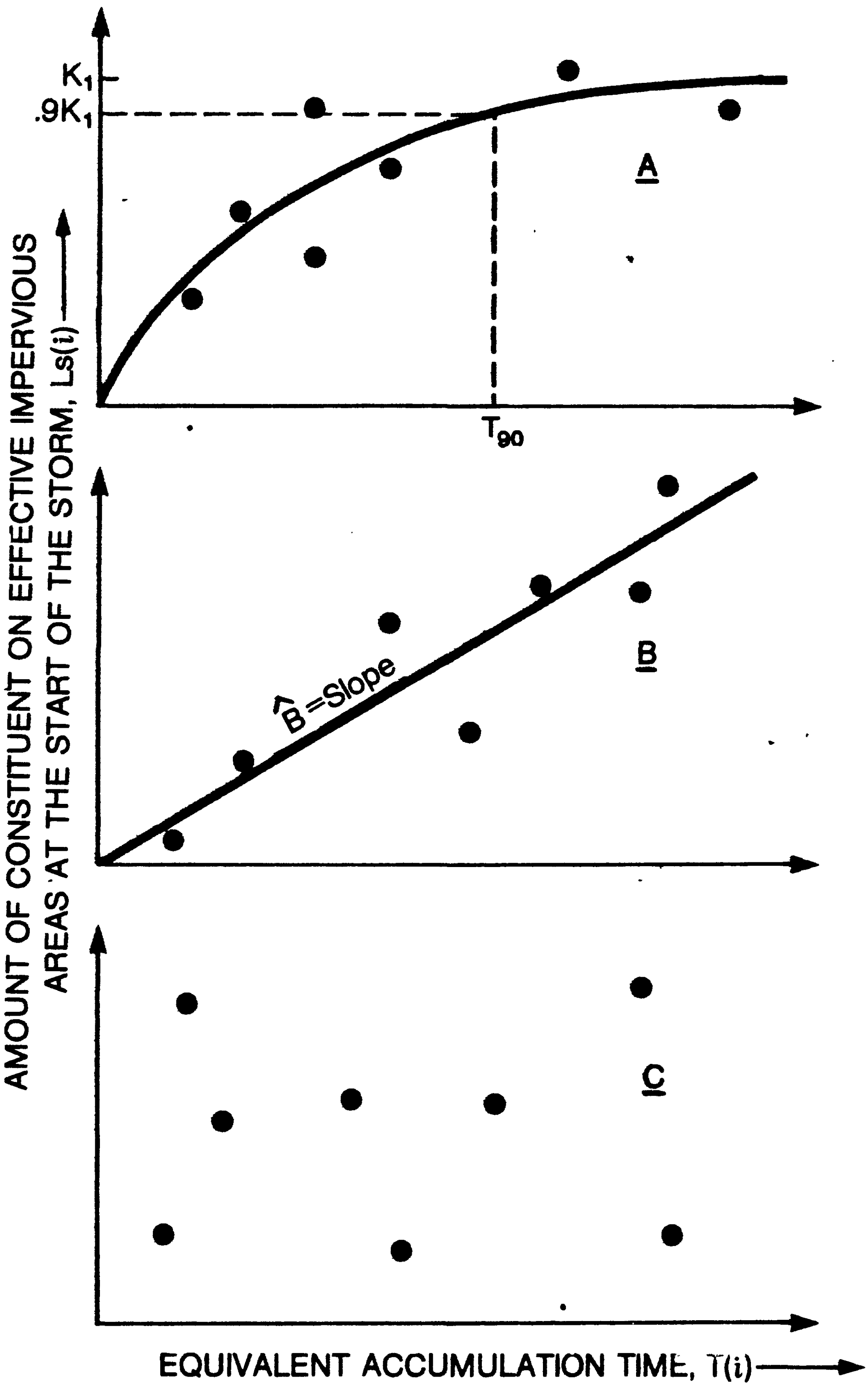


Figure 4.--Accumulation relationships displayed by plots of $L_s(i)$ versus T(i) including A--exponential accumulation, B--linear accumulation, and C--no apparent relationship.

the model by assigning a value for K_4 in the input. An estimate for K_4 can be made by dividing 1.0 by RMAX, where RMAX is the estimated runoff intensity above which no adjustment to exponential washoff is necessary. It should be noted, however, that if an availability factor is used it may change significantly the optimum values for K_1, K_2, and K_3, and these values may have to be re-adjusted if they have been determined without including the availability factor.

In summary, the following procedure can often be used for calibrating DR_3M-QUAL as a lumped-parameter model:

(1) Answer the question: Do water-quality constituent concentrations tend to decrease with time within a given storm?

(2) If the answer to question 1 is yes, estimate initial values of K_1, K_2, and K_3.

(3) Run model and fit parameter K_3 using measured and simulated load characteristic curves.

(4) Estimate K_1 and K_2 using values of $L_s(i)$ and $T(i)$, $i = 1, \ldots, N$ output by model.

(5) Run model with new estimates of K_1 and K_2 and obtain new estimates of $T(i)$, $i = 1, \ldots, N$.

(6) Repeat steps 4 and 5 until changes in $T(i)$, $i = 1, \ldots, N$ have negligible impact on the simulation results.

(7) If necessary adjust initial simulated concentrations during storms using K_4. This may require readjustment of the values estimated for $K_1 - K_3$.

The above procedure pertains to use of equation 5 as the washoff equation. If equation 9 is used instead, then the simulated load characteristic curves are still independent of the values of K_1 and K_2 and the load characteristic curves can still be used to fit K_{3*}. However, the present version of the model does not output values of $L_s(i)$ and $T(i)$, which could be used to estimate K_1 and K_2. Runs of the model using equation 5 and the above procedure may provide some guidance for estimating K_1 and K_2 using equation 9 for washoff. Values of K_{3*} will likely be much different from K_3.

Street Sweeping

A schematic illustrating DR_3M-QUAL's conceptualization of constituent accumulation on, and removal from, effective impervious areas is shown on figure 5. Removal of constituents can be by rainfall runoff or by street sweepers.

Street sweeping can be simulated according to a fixed schedule or actual days during which street sweeping occurred can be input to the model. The model assumes that street sweeping operations are suspended on unit days or on daily accounting days when rainfall exceeds impervious retention. If street sweeping is suspended because of rainfall, the model assumes that the streets are not swept until the next day on which street sweeping was scheduled.

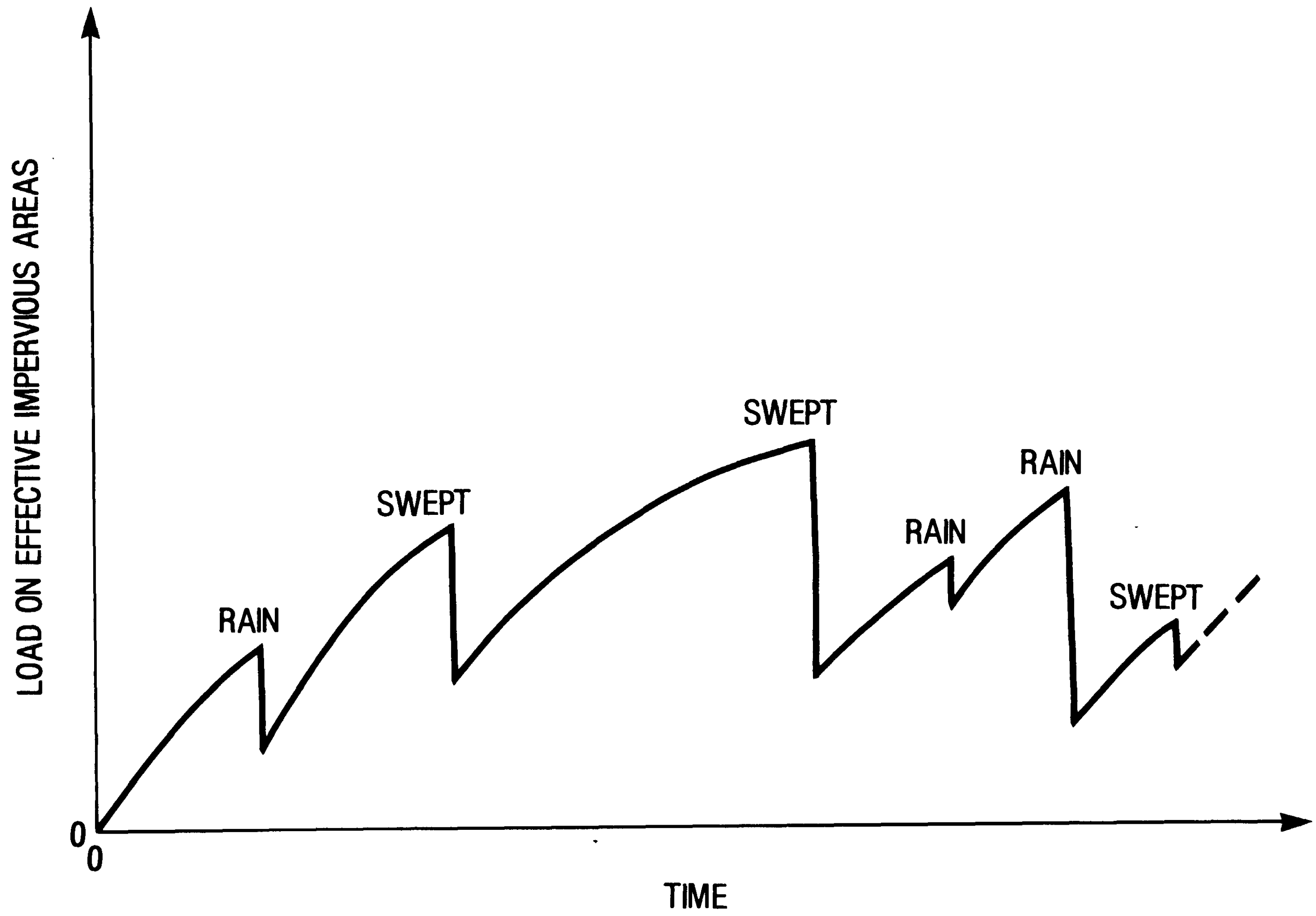


Figure 5.--Constituent accumulation on effective impervious areas and removal by rainfall and street sweeping.

This assumption conforms to common street sweeping practices of public works departments.

DR_3M-QUAL simulates removal of constituents by street sweeping as:

$$L_b = \begin{cases} L_a - E(L_a - L_r); & \text{if } L_a > L_r \\ L_a ; & \text{if } L_a \leq L_r \end{cases} \qquad (13)$$

where L_b is the amount of a constituent remaining after street sweeping, L_a is the amount of a constituent before street sweeping, L_r is the base residual load which cannot be removed by street sweepers, and E is the efficiency (ranging from 0 to 1.0) of street sweepers at removing the load of a constituent in excess of the base residual load.

Only part of the effective impervious area of a watershed (lumped-parameter simulation) or overland-flow segment (distributed simulation) may be swept by street sweepers. This is accounted for in the model as:

$$L_c = L_b \cdot (SSAREA) + L_a(1.0 - SSAREA) \qquad (14)$$

where L_c is the adjusted amount of the constituent remaining on effective impervious surfaces after street sweeping and SSAREA is the fraction of the effective impervious area that is swept.

Pervious-Area Runoff Quality

Little information is available on the contributions of pervious areas to urban runoff quality. A study by Barkdoll, Overton, and Betson (1977) suggests that "semipervious and pervious areas are highly significant in their contribution to urban water pollution." Theoretical studies by the originators of SWMM (Metcalf and Eddy, Inc., and others, 1971) indicated that very large rates of runoff would be required to remove dust and dirt from grass plots, and that unless erosion takes place from ungrassed areas, the contribution of pervious surfaces to suspended solids content is minor. However, they noted that runoff from pervious surfaces may contain significant amounts of dissolved solids.

Several indicators of the relative contributions of pervious and impervious areas to runoff loads can be used. These include the percentage of runoff that is from pervious areas, empirical knowledge of sources, results of regression analyses, distributions of constituent concentrations and loads over storm hydrographs, results of street-surface sampling, and the effectiveness of management strategies such as street sweeping.

Empirical knowledge of sources can provide an indication of the relative contributions of pervious and impervious areas to runoff loads for certain constituents. For example, one might expect pesticides to originate primarily from pervious areas and lead which is associated with automobile exhaust to originate primarily from impervious areas. Other constituents such as nitrogen species which are associated with organic pollution, fertilizers, and automobile sources may originate primarily from either pervious or impervious areas.

Regression analyses of storm-runoff loads with parameters such as average daily traffic or pavement condition may provide an indication of sources by identifying significant variables.

The distribution of constituent concentrations and loads over storm hydrographs may provide some indication of the relative contributions of pervious and impervious areas to runoff loads. In addition, the relative proportions of pervious- and impervious-area contributions to runoff, as determined from a rainfall-runoff model, might be compared to trends in constituent concentrations and loads.

Sampling of street-surface solids in conjunction with runoff-quality studies may provide an indication of the potential street-surface contributions to storm-runoff loads. These can be compared to storm-runoff loads. However, samples collected from street surfaces may not be typical of land-surface loads on other types of effective impervious areas. Also, the material removed by the sampling method may not be representative of the amount of constituent available for storm-water washoff.

Finally, the effectiveness of management strategies such as street sweeping may provide an indication of the relative importance of pervious and impervious areas as sources of storm-runoff loads. For example, effectiveness of street sweeping as measured by removal of constituents from streets could be compared to its effectiveness in reducing constituent loads in storm runoff.

Due to rapid response times of most urban watersheds, usually only the surface runoff and quick-return flow (i.e., interflow with shallow penetration of the soil) are of primary interest. Chemical constituents added to solution would normally be those characteristic of surficial soils. Techniques for predicting the interactions between these soils and the runoff are not far advanced and are often based on equilibrium chemistry. Unfortunately, the chemical extraction processes at the soil-runoff interface during the brief time of a runoff event are not likely to reach equilibrium (Bruce and others, 1975). Also, the amount and distribution of a constituent in the soils of a watershed cannot be determined precisely without intensive sampling and analysis.

Given the state-of-the-art, simple equations for predicting pervious-area contributions to runoff quality are included in the model. Conceptually, constituents in pervious-area runoff are classified according to two sources: (1) that part dissolved or desorbed from the soil matrix into the flowing water without soil erosion, and (2) that part associated with soil erosion. Mathematically,

$$CP(J) = a + b \cdot SED(J) \quad (15)$$

where CP(J) is the concentration of a constituent in pervious-area runoff at time step J, a is a constant representing the "non-erosion" contribution, b is a constant representing the ratio of the constituent concentration to the sediment concentration, and SED(J) is the concentration of sediment in the pervious-area runoff at time step J.

Williams (1975) has determined that the total sediment yield resulting from a period of storm runoff can be estimated by:

$$TMASS = PFAC1(V_R Q_P)^{PFAC2} \qquad (16)$$

where TMASS is the total sediment yield in tons, V_R is the volume of runoff in acre-ft, Q_P is the peak flow rate in ft^3/s; and PFAC1 and PFAC2 are constants.

Based on studies on 18 watersheds in Nebraska and Texas, Williams (1975) found that PFAC2 could be estimated as 0.56 and PFAC1 as a function of parameters in the Universal Soil Loss Equation (Wischmeier and Smith, 1965):

$$PFAC1 = 95 \cdot K \cdot LS \cdot C \cdot P \qquad (17)$$

where K is the soil erodibility factor, LS is the length-slope gradient ratio, C is the cropping management factor, and P is the erosion-control practice factor. Use of equation 17 and setting PFAC2=0.56 explained about 92 percent of the variation in sediment yield from the 16 Nebraska and 2 Texas watersheds (Williams, 1975). The transferability of the above relationships for PFAC1 and PFAC2 is undocumented and use of locally collected data is recommended for estimating these parameter values.

Instantaneous sediment concentrations (i.e., SED(J)) can be estimated from TMASS, using the assumption that sediment concentrations are proportional to flow rate. Based on studies by Rendon-Herrero (1974)., Kuo (1975), and Williams (1978), this appears to be a reasonable assumption. The equation used by the model is:

$$SED(J) = VOL(J) \cdot TERM \qquad (18)$$

where

$$TERM = \frac{735.48\ TMASS}{\sum_{I=1}^{ICT} VOL(I)^2} \qquad (19)$$

and VOL (J) is the incremental flow at time step J, in acre-ft, ICT is the number of time steps, and 735.48 is used to convert concentration in tons/acre-ft to a concentration in mg/L.

Precipitation Quality

Urban runoff can be viewed as chemically modified rainwater. Accounting for the original chemical composition of this rainwater may be important in terms of both model reliability and evaluation of management practices. For example, precipitation contributions to runoff loads should be largely unaffected by street sweeping practices.

DR_3M-QUAL accounts for precipitation contributions to runoff quality by adding a concentration representing precipitation quality to the concentrations predicted from constituent washoff. This adjustment can be a constant for all storms, can vary monthly, or can vary on a storm-by-storm basis. Although it is well known that the chemistry of precipitation can vary during a storm, few data exist to quantify this variation. Thus, assigning a constant concentration to precipitation on a storm basis is the maximum level of detail currently included in the model.

If a term is included in the model to account for precipitation quality, then the magnitude of this term will affect the best-fit values of the accumulation and washoff parameters. However, even for the parameter estimation techniques described for lumped-parameter runs, this does not present a large problem. The procedure for fitting K_3 using the plots of simulated and measured load characteristic curves is unchanged. However, in fitting the accumulation parameters the runoff load contributed by the precipitation should be accounted for. Because substances in precipitation are assumed to be conservative, and precipitation quality is assumed constant throughout a given storm, precipitation quality can be accounted for by replacing ΔL by $(\Delta L)_{adj}$ in equation 10a where

$$(\Delta L)_{adj} = \Delta L - 0.277PA(V_a - V_b) \qquad (20)$$

where $(\Delta L)_{adj}$ is the measured runoff load between the first and last sample minus the assumed precipitation load, in pounds, P is the precipitation quality in mg/L, A is the effective impervious area in acres, and 0.277 is a conversion factor to convert to pounds. Values of $L_s(i)$ computed by the model are based on equation 20, if a precipitation quality term is included in the model.

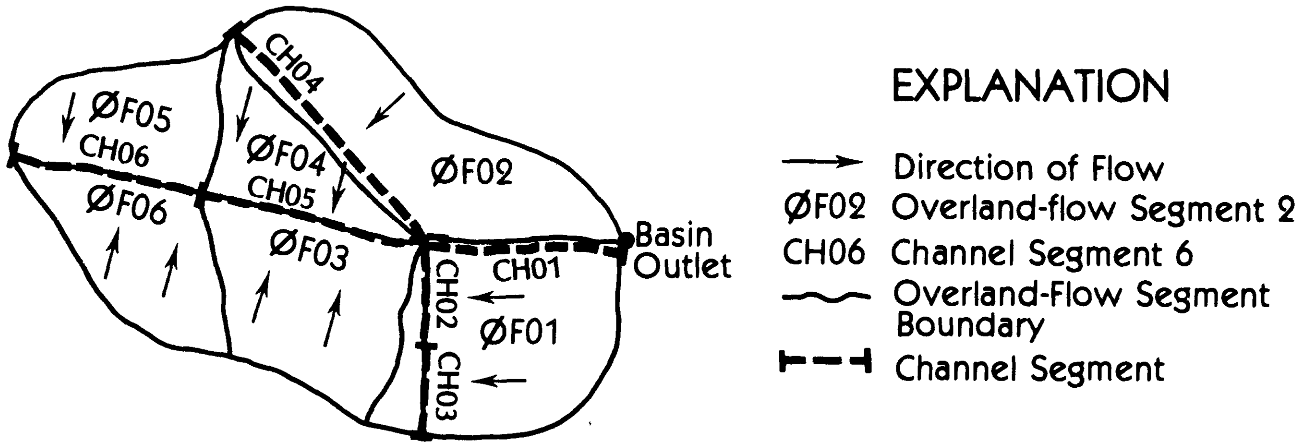
CONSTITUENT TRANSPORT

For applications of DR_3M-QUAL as a distributed-parameter model, DR_3M-QUAL uses segment flow data generated by DR_3M and stored on disk files. A drainage basin is represented as a set of segments which jointly describe the drainage features of the basin. The same types and designations of segments apply for DR_3M-QUAL as for DR_3M. These include:

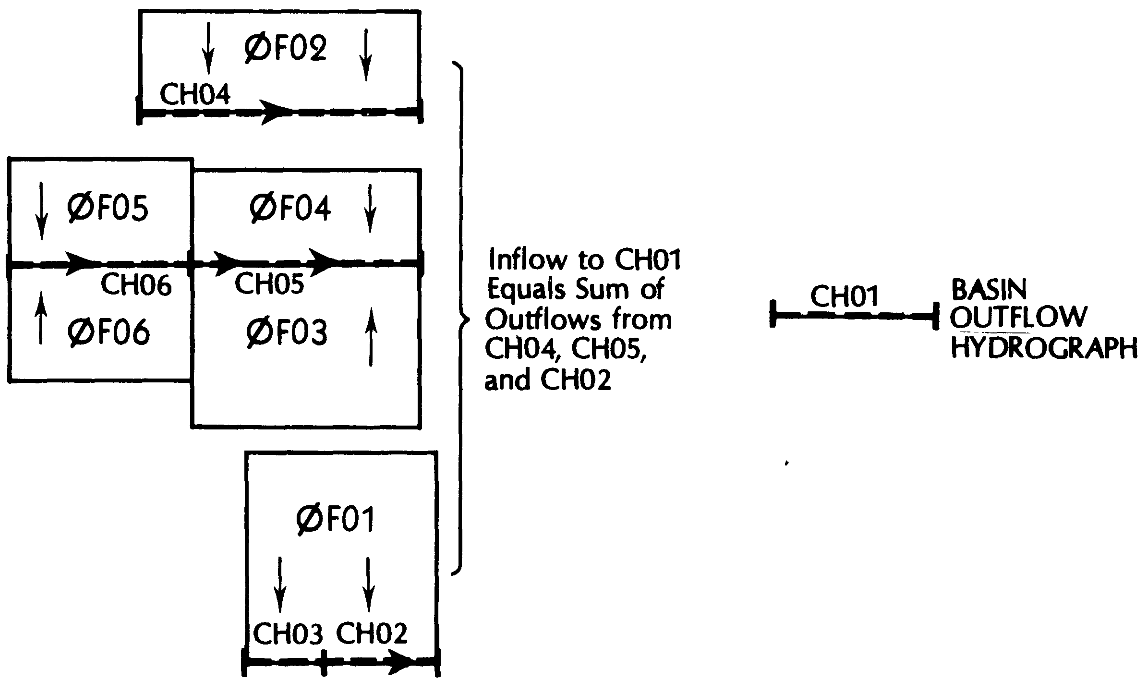
1. overland-flow segments
2. channel segments
3. reservoir segments
4. nodal segments

Channel and Overland-Flow Segments

Figure 6 illustrates the relationships between channel and overland-flow segments in DR_3M. It is important to understand the manner in which spatial variations in impervious-area and pervious-area runoff-quality parameters are taken into account in distributed runs of the DR_3M-QUAL. For impervious-area runoff-quality parameters DR_3M-QUAL provides for as many as four different "land-use types." The model user can assign a different set of values for the impervious-area runoff-quality parameters (K_1 - K_4)



(a) PLAN VIEW OF DRAINAGE BASIN



(b) SCHEMATIC REPRESENTATIONS OF MODEL SEGMENTS

Segment	Inflow to Segment	
	Lateral Inflow	Upstream Inflow
ØF01	Rainfall Excess	—
ØF02	" "	—
ØF03	" "	—
ØF04	" "	—
ØF05	" "	—
CH01	—	CH02, CH04, CH05
CH02	ØF01	CH03
CH03	ØF01	—
CH04	ØF02	—
CH05	ØF03, ØF04	CH06
CH06	ØF05, ØF06	—

(c) SEGMENT INTERRELATIONSHIPS

Figure 6.--Segmentation of an urban watershed into channel and overland flow segments.

to each of the land-use types. However, rather than assigning land-use types to overland-flow segments, land-use type is referenced to the channel network. Thus, the same overland-flow segment can be used several times, as documented in the DR_3M user's manual (Alley and Smith, 1982), but with a different land-use type associated with its different locations in the watershed. All overland-flow segments draining to a particular channel are assumed to have the same land use.

Of the pervious-area runoff-quality parameters, the parameters, PFAC1 and PFAC2 (see equation 16) are also referenced to channel segments and can vary between overland-flow segments draining to different channels but cannot vary between overland-flow segments draining to the same channel. On the other hand, the parameters a and b (see equation 15) are assumed to be constant for a given constituent throughout the watershed.

Constituent loads from overland-flow segments are determined using the discharge from these segments and the algorithms for impervious-area, pervious-area and precipitation quality previously described. These loads are contributed laterally to channel segments. Runoff from an overland-flow segment is assumed to correspond to impervious-area runoff unless an overland-flow plane is represented as two segments; one representing pervious-area runoff and the other impervious-area runoff. In this case, the pervious-area runoff-quality algorithms are used with the flows from the overland-flow segment representing pervious-area runoff.

The transport of constituents through channels is performed using a Lagrangian method that is precise and conceptually simple. The method is designed to solve the simplest transport problem, the plug flow (dispersionless) transport of a conservative substance. The basic Lagrangian method has been modified to account for constituents that enter channel flows by lateral inflow.

In a Lagrangian method, one conceptually follows an individual fluid parcel while keeping track of all factors which tend to change its concentration. If dispersion (mixing between parcels) is assumed negligible, and the pollutant being modeled is considered conservative, then the only factor that will tend to change the concentration of a parcel as it moves downstream is the addition of lateral flow of a different concentration.

The Lagrangian method in DR_3M-QUAL operates on one channel segment at a time starting at the upstream end of the basin and proceeding in the downstream direction. Computations for all time steps are computed for one channel segment, before proceeding to the next channel segment. Knowing the time-varying concentrations of the upstream and lateral flows entering a segment, the method is used to determine the time-varying concentrations of the flow leaving the segment. The method is no more than a bookkeeping that keeps track of the volume of water and the constituent concentrations in all fluid parcels that are in a segment during a time step.

Considering a channel segment, the method operates on a time-step basis. During each time step, a parcel enters the segment and is assigned the volume and constituent concentrations of the flow entering from the upstream end.

At the beginning of a time step there are anywhere from one to many parcels in the segment, and the number of parcels increases by one when the new parcel enters. The lateral inflow that enters the segment during a time step is distributed among all parcels in proportion to parcel volumes. Volume and constituent concentrations in each parcel are adjusted to account for the addition of the lateral inflow. At the downstream end of the segment, part of one or more parcels leaves the segment depending on the volume of flow that leaves the segment during the time step. At the end of a time step the constituent concentrations in the parcel at the downstream end of the segment are saved in an array and the number of parcels is updated. After completing all time steps, this array of concentrations is used as input to the next downstream segment.

Reservoir Segments

DR3M-QUAL transports water-quality constituents through reservoir segments (detention basins) using plug-flow concepts. Plug flow assumes no mixing between plugs and routes the flow on a first in, first out basis. The model assumes that a detention basin has one outlet and that all inflow occurs from channel(s) located at the most upstream end of the basin.

Particulate material entering a reservoir is assumed to settle according to Stokes' Law, and particles are considered trapped as soon as they reach the reservoir bed. Resuspension or movement along the bed of particles is not accounted for. Because of the short detention times that generally occur in most urban detention basins, chemical reactions are not considered by the model. The detention storage component consists of two main parts: plug routing and constituent settling.

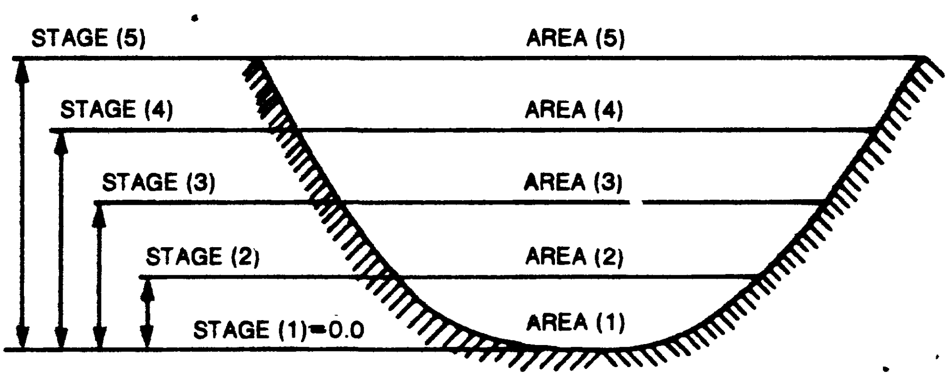
Plug Routing

Stokes' Law has generally been applied to problems involving estimation of the removal of discrete particles in settling tanks for water treatment facilities (Fair and others, 1968). However, urban detention basins, unlike water treatment facilities, are generally not well-defined hydraulic structures. Rather, these basins are often irregular in shape.

Basin geometry in the model is determined by the input of a stage-area curve. This curve is input as a set of points defining the surface area of the basin at selected stages. Stage is defined in the model as the depth of water above the lowest bed elevation. The capacity or volume of the basin below any given stage point (N) is computed by the trapezoidal method:

$$CAPAC(N) = \sum_{J=2}^{N} (Area(J) + AREA(J-1)) \cdot (STAGE(J) - STAGE(J-1))/2 \qquad (21)$$

where CAPAC(N) is the reservoir capacity in acre-ft at stage point (N), AREA (J) is the surface area in acres at stage point (J), and STAGE (J) is the stage in ft at stage point (J). An example is shown in figure 7.



$$CAPAC(5) = \sum_{J=2}^{5} \left(AREA(J) + AREA(J-1) \right) \left(STAGE(J) - STAGE(J-1) \right) / 2$$

Figure 7.--Detention basin geometry.

The average depth of water is a volume-weighted average of the water depth throughout the basin. The average depth AVDPTH (N) for each stage point (N) is defined by a relationship adapted from the work of Ward and others (1979):

$$\mathrm{AVDPTH(N)} = \frac{\sum_{J=2}^{N} \mathrm{DEPO(J)}^2 \cdot (\mathrm{AREA(J)} - \mathrm{AREA(J-1)})}{\sum_{J=2}^{N} \mathrm{DEPO(J)} \cdot (\mathrm{AREA(J)} - \mathrm{AREA(J-1)})} \qquad (22)$$

where

$$\mathrm{DEPO(J)} = \mathrm{STAGE\ (N)} - (\mathrm{STAGE\ (J)} + \mathrm{STAGE\ (J-1)})/2.0 \qquad (23)$$

The inflow of water to a detention basin is defined by the sum of the upstream flows to the basin at each time step. Constituent concentrations at the upstream end of a detention basin are determined as a volume-weighted average of inflow concentrations at each time step.

The water volume in the detention basin is subdivided into separate plugs each one containing all the outflow during one time increment. The time increment used is the unit time interval specified in the model (PTIME). If the time interval for flow routing (DT) is less than the unit time interval, then the effluent constituent concentrations at the DT increments are determined by linear interpolation of the effluent concentrations at the unit time intervals.

DR_3M-QUAL retrieves the inflow and outflow hydrographs for a reservoir from the segment flow files created by DR_3M. From this information the model can determine the volume of each plug. Figure 8 shows the location of plug number 7 on typical inflow and outflow hydrographs for a unit-time interval (PTIME) of 1 hour. The starting point location of the plug on the inflow hydrograph is determined by first finding the point at which the accumulated inflow (point A on inflow hydrograph) is equal to the accumulated outflow (point C on outflow hydrograph). The ending point location of the plug on the inflow hydrograph (point B) is determined in a similar manner but based on point D. The detention time (DETTME) of the plug is then assumed equal to the time between the centroids of the inflow and outflow plugs as shown on figure 8. The constituent mass of the inflow plug is determined by linear interpolation at the plug starting and ending times (A and B) of a curve relating accumulated constituent mass of the inflow and time since start of storm.

Finally, the average depth of a plug (DEPTH), while in a detention basin, is determined by first computing the area under the average depth versus time curve contained between the centroid of the plug on the inflow hydrograph and the centroid of the plug on the outflow hydrograph. This area is then divided by the detention time of the plug (DETTME). This computation is also illustrated in figure 8. The average depth versus time curve is obtained by linear interpolation of the stage-average depth curves defined by equation 22.

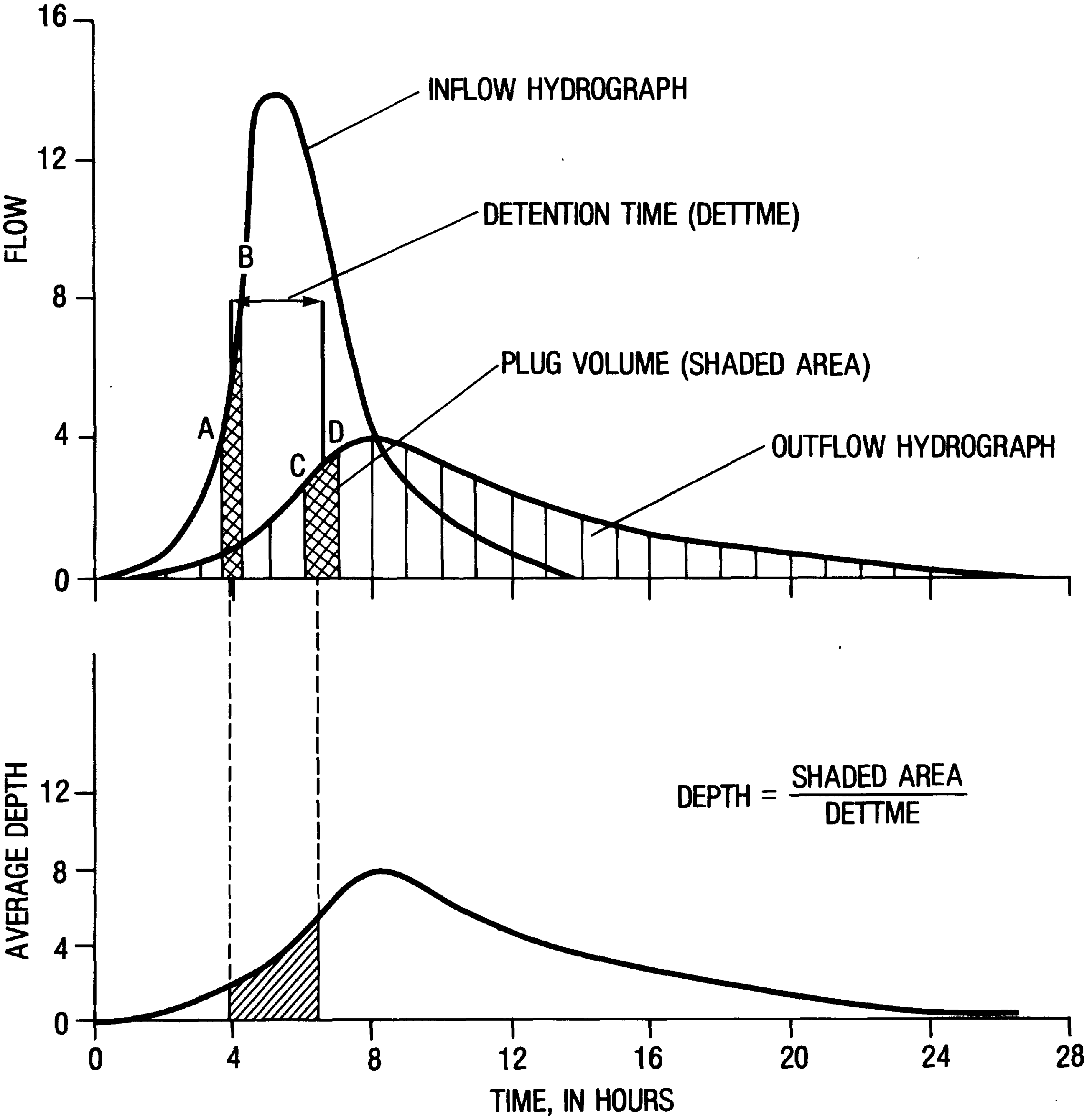


Figure 8.--Plug routing (after Ward and others, 1979).

Constituent Settling

The rate at which water-quality constituents settle in a detention basin depends to a large extent on the particle-size distribution of the constituent. For this reason, the input to the model includes a particle-size distribution curve for influent concentrations of each water-quality constituent to each detention basin. Although, particle-size distributions of the influent should vary throughout a storm event, no data presently exist to accurately quantify this effect. Therefore, for a given constituent and detention basin, the particle-size distribution of the influent is assumed to remain constant.

The model assumes that constituents are evenly distributed throughout a plug at its entrance to the detention basin. Constituents then gradually settle out as the plug moves through the basin. Settling of discrete non-flocculating particles that settle without hindrance from other particles can be described by Stokes' Law. If the settling velocity of a particle is assumed to be sufficiently low, such that fluid flowing past the particle is within the laminar range, and if the particle is assumed spherical, Stokes' Law can be written as:

$$V_s = KD^2 \tag{24}$$

where

$$K = 5.15 \times 10^{-5} \frac{(SG-1)}{\nu} \tag{25}$$

where V_s is the settling velocity of the particle in ft/hr, SG is the specific gravity of the particle, D is the diameter of the particle in microns, and ν is the kinematic viscosity in cm^2/s. Kinematic viscosity and specific gravity are assumed to be constant by the model. The factor 5.15×10^{-5} is 0.8 times the acceleration due to gravity (32.2 ft/s^2) times a conversion factor to account for the different units used in the equation. The factor 0.8 is used to correct for the non-spherical nature of clay and colloidal particles (Ward and others, 1979). The assumption is made that if some of the fine particles satisfying Stokes' Law are trapped, then the coarser particles will be trapped also.

The paths traced by discrete particles that are settling in a rectangular basin are shown in figure 9. They are determined by the vector sums of the settling velocity (V_s) of the particle and the horizontal displacement velocity of the surrounding fluid (V_d). All particles with a settling velocity $V_s \geq V_o$ are removed, V_o being the velocity of that particle that falls through the full depth (H_o) of the settling zone (L_o) in the detention time (T_o). The parameter V_o can be defined as:

$$V_o = \frac{H_o}{T_o} \tag{26}$$

For the case of plug flow through a non-rectangular basin, as in this model, V_o is defined as:

$$V_o = \frac{DEPTH}{DETTME} \tag{27}$$

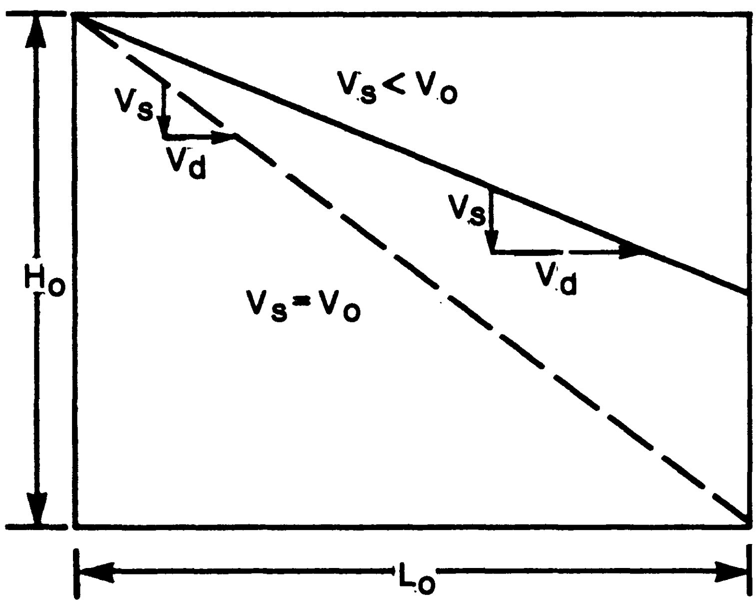


Figure 9.--Paths taken by discrete particles through a rectangular tank of length L_o and height H_o (after Fair and others, 1968).

where DEPTH and DETTME have been previously determined in the plug routing component.

If N_o (out of N_t) particles having a settling velocity $V_s \leq V_o$ compose a particle-size fraction, the proportion X of the particles to be removed is:

$$X = \frac{N_o}{N_t} = \frac{V_s}{V_o} \quad (28)$$

The composition of a particular suspension can be expressed by a curve showing the cumulative frequency distribution of settling velocities (see fig. 10). Defining F_o as the fraction of particles having settling velocities $V_s \leq V_o$, then $(1-F_o)$ of the particles have settling velocities $V_s \geq V_o$ and are totally removed. The fraction of the particles with settling velocities $V_s < V_o$ that are removed is:

$$r_2 = \int_o^{F_o} (V_s/V_o)dF \quad (29)$$

Total removal (r) is then:

$$r = (1-F_o) + \frac{1}{V_o} \int_o^{F_o} V_s dF \quad (30)$$

The fraction of particles remaining in suspension (R) is:

$$R = 1 - r$$

$$= F_o - \frac{1}{V_o} \int_o^{F_o} V_s dF \quad (31)$$

A cumulative distribution of particle diameters, D, is input to the model by specifying as many as 10 particle sizes and their respective respective cumulative frequencies. The model uses these data to construct a piecewise-linear cumulative frequency distribution curve. An example for 5 particle sizes (including $D_1 = 0$) is shown in figure 11.

For a piecewise-linear cumulative frequency curve equation 31 can be rewritten as:

$$R = F_o - \frac{1}{V_o} [V_t(m) + \int_{F_m}^{F_o} V_s dF] \quad (32)$$

where m is the number of particle-size values read into the program that are less than D_o (3 in the example shown in figure 11) and

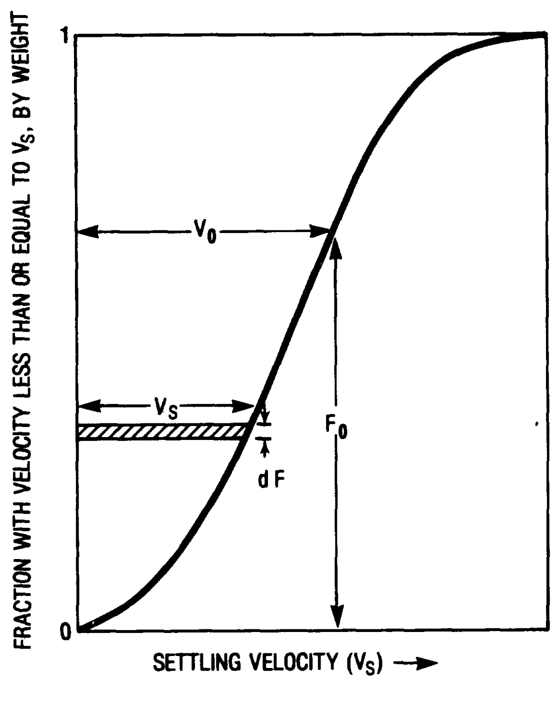


Figure 10.--Cumulative frequency distribution of settling velocity.

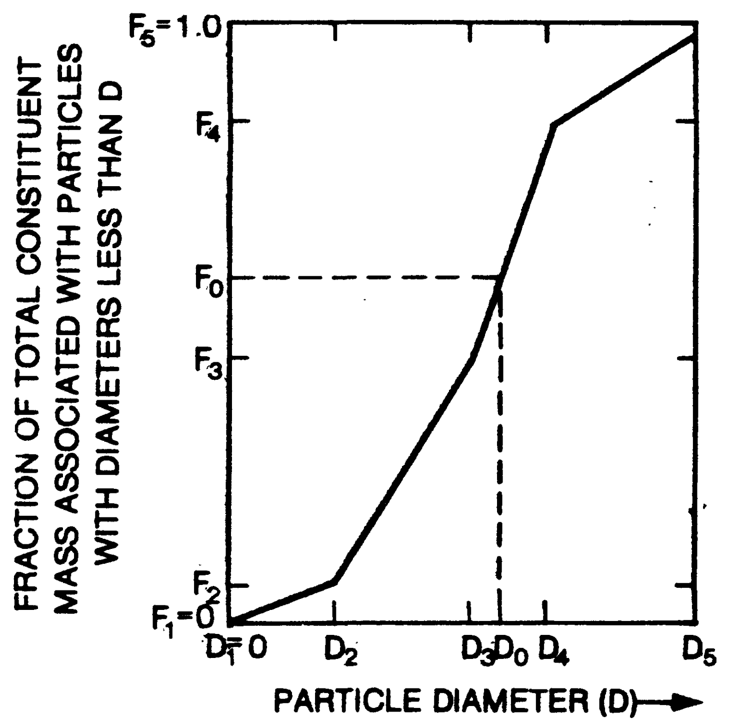


Figure 11.--Cumulative frequency distribution of particle size.

$$V_t(m) = \begin{cases} 0 & , \text{ if } m = 1 \\ \sum_{i=2}^{m} [\int_{F_{i-1}}^{F_i} V_s dF] & , \text{ if } m > 1 \end{cases} \tag{33}$$

The integral in equation 33 can be evaluated by substituting equation 24, letting

$$dF = \frac{(F_i - F_{i-1})}{(D_i - D_{i-1})} dD \tag{34}$$

and changing the limits of integration:

$$\int_{F_{i-1}}^{F_i} V_s dF = \int_{D_{i-1}}^{D_i} KD^2 \frac{(F_i - F_{i-1})}{(D_i - D_{i-1})} dD$$

$$= \frac{(F_i - F_{i-1})}{(D_i - D_{i-1})} \frac{K}{3} [D_i^3 - D_{i-1}^3] \tag{35}$$

Likewise, the last integral in equation 32 can be evaluated as

$$\int_{F_m}^{F_o} V_s dF = \frac{(F_o - F_m)}{(D_o - D_m)} (\frac{K}{3}) [D_o^3 - D_m^3] \tag{36}$$

For a given reservoir and constituent the cumulative frequency distribution of particle sizes is assumed constant. Therefore, the numerical method for evaluating equation 32 is as follows:

1. $V_t(m)$ for all m is computed once at the start of the program and is stored for later use.

2. The model routes the plug through the reservoir and determines the values of the parameters DEPTH and DETTME.

3. V_o is computed based on equation 27 and a corresponding D_o is computed.

4. The value of m is determined based on D_o and the value of F_o is determined by linear interpolation of the cumulative frequency distribution of particle sizes between D_m and D_{m+1}.

5. Equation 32 is solved by substituting equation 36 into it, as well as the previously determined values of F_o, V_o, and $V_t(m)$.

Non-Ideal Basins

Typically, water is not released from a detention reservoir uniformly from all depths. For example, basins with a large permanent pool and a drop inlet riser will normally exhibit complete withdrawal from near the surface of the reservoir storage provided the head of flow above the riser crest is much smaller than the total depth of flow. Risers with the crest near the bed of the basin or trickle dewatering tubes, will exhibit withdrawal from the bed layer. Perforated risers tend to exhibit withdrawal characteristics between these extremes (Ward and others, 1979). This model does not contain a method for approximating selective withdrawal at the reservoir outlet. Some plug flow models such as the DEPOSITS model (Ward and others, 1979) do contain such features. However, they tend to be very crude approximations due to the basic incompatibility of plug flow and selective withdrawal.

Under idealized conditions, plug flow will occur and the flow will pass through the pond on a first in, first out basis. However, flow in all detention basins is characterized by mixing, turbulence, short-circuiting, and resuspension. A well-designed detention pond will minimize these factors. A pond which is likely to exhibit a high degree of short-circuiting, mixing, and turbulence will probably have a low trap efficiency and will not be suitable for evaluation with the detention storage algorithms included in this model.

Many detention basins will retain some water even during dry weather. This volume of water is referred to as the permanent pool capacity and the reservoir can be assumed to be at its permanent pool capacity at the start of the storm. The model treats this permanent pool capacity in one of two ways, depending on the value of the parameter DEAD (for dead storage). If DEAD is set equal to the permanent pool capacity, then this volume of water is bypassed during plug routing. If DEAD is set equal to zero, then the permanent pool capacity is not bypassed and initial effluent concentrations will be low. In reality, the detention basin will respond somewhere in between these two extremes, particularly if part of the permanent pool becomes filled with sediment.

Nodal Segments

Nodal segments are used when more than three segments contribute inflow to the upstream end of a segment or as an input hydrograph or input discharge point. The user of DR_3M-QUAL can assign a constant concentration for each water-quality constituent to each input hydrograph and input discharge. Nodal segments used as input hydrographs or input discharge points should not have any upstream segments.

MODEL APPLICATIONS

DR_3M-QUAL can be used for a wide variety of applications. The model is continuous in time; and therefore, an accounting of impervious-area constituent accumulation is made between storm events. Rather than operating on a fixed time step, the model provides short-time interval simulation of those storm events specified by the user and a daily accounting of constituent accumulation and washoff between these storm events. Therefore, many of the advantages of continuous simulation are combined with those of a single event model.

One possible use of a calibrated model for a watershed is to extend records of storm-runoff loads for the watershed. One application of record extension might be to synthesize runoff loads for comparison with concurrent flow rates and quality characteristics of receiving waters. In addition, one might have a complete or nearly complete record of rainfall and (or) runoff at the site, but water-quality data for only a small portion of the runoff. DR_3M-QUAL could be used to estimate constituent concentrations during the unsampled portion of the runoff. In this manner, annual runoff loads or other characteristics of interest could be estimated for the study period.

DR_3M-QUAL could potentially be used to transform long-term precipitation records into long-term records of storm-runoff loads from which frequency distributions of loads could be estimated. One problem in using DR_3M-QUAL for this purpose lies in the dearth of long-term records of precipitation. Long-term precipitation records at hourly intervals can be obtained for many cities from the National Weather Service. However, shorter time interval data are less available. Although long-term records of precipitation data at 5-minute intervals can be obtained from the National Weather Service for many cities in the U.S., the data are only for anywhere from 3 to over 10 "major" storms per year for a period of record often exceeding 50 years. These storms are those which would most likely produce the largest peak flows from a watershed. However, for determination of characteristics such as annual loads, all significant runoff-load producing storms are required.

DR_3M-QUAL can conceptually account for the effects of the management practices of street sweeping and detention storage. The detention storage subroutine of DR_3M-QUAL can be removed and used as a separate program (see attachment D).

DR_3M-QUAL has been designed for ease of calibration. A user has the option of reading in measured runoff-quality data for graphical and numerical comparisons. Output from the model can include graphs of measured versus simulated constituent concentrations and/or measured versus simulated load characteristic curves for each storm event. Scatter plots of measured versus simulated runoff loads are also output.

The model can be run on one of three spatial modes:

1. Lumped-parameter
2. Distributed (no transport)
3. Distributed transport

As a lumped-parameter model, DR_3M-QUAL uses runoff data from the watershed outlet. These data can be measured values or can be simulated values from DR_3M. Runoff loads are assumed to originate predominately from the effective impervious areas of the watershed. An approach has been presented for estimating best-fit parameter values for lumped-parameter runs. Lumped-parameter simulations can be used as an end in themselves or to inexpensively estimate impervious-area model parameters for later, more detailed distributed runs.

When an application involves determining the effects of runoff quality on the quality of local receiving waters, the time interval of interest may be days or even weeks or months, while DR3M-QUAL simulation intervals are on the order of minutes. Thus, the magnitude of storm-runoff loads may be much more important than within-storm variations. For this reason, DR3M-QUAL contains an option for distributed simulations without constituent transport. Model simulations in this mode are equivalent to a distributed-parameter run with instantaneous transport instead of Lagrangian transport. Considerable savings in computer costs can be made by using this mode. Unlike lumped-parameter simulations, pervious-area contributions to runoff loads can be accounted for as well as spatial variations in runoff-quality parameters. Like lumped-parameter simulations, distributed (no transport) runs can be used for initial calibration of model parameters prior to final calibration as a distributed transport model. Output from distributed (no transport) runs includes storm-runoff loads but no information about within-storm variation is given. Distributed (no transport) runs cannot account for the effects of reservoir segments, input hydrograph points or input discharge points.

Typically, distributed watershed models such as DR3M-QUAL are calibrated and verified using data on the total watershed response at outfalls. These data represent the combined effects of many complicated processes including impervious-area accumulation and washoff, routing, sedimentation, biological and chemical reactions, and atmospheric deposition. DR3M-QUAL has been developed such that there are a minimum of parameters to calibrate. Advantages of the parsimonious nature of this model include that:

1. Calibration procedures are facilitated.
2. Computer costs are reduced, thus enhancing the utility of the model for record extension and other applications of continuous simulation.
3. Model calibration is less likely to degenerate into a curve fitting process, whereby a multitude of parameters are randomly varied until a good fit is achieved.

However, considerable caution should be used in applying the model to assure that it properly represents the system being simulated. The model has been designed for urban applications and it is recommended for use only when impervious-area runoff is considered to be the principal source of runoff loads. Because the constituent transport algorithm in the model for channel segments assumes plug flow and conservative constituents, the model should only be used on small urban watersheds. The model should be most useful on watersheds having drainage areas on the order of several square miles or less. It is particularly important that model parameters estimated for a watershed not be transferred to a watershed having a much larger or smaller drainage area.

In general, attempts should be made to minimize the number of segments used in distributed runs. Often the number of segments can be dictated by the number required to characterize the assumed variations in accumulation and washoff parameters and to differentiate pervious and impervious-area runoff. Use of just a few segments is commensurate with the underlying simplicity of the model. Subdividing the watershed into two segments, one pervious and the other impervious, may be a valid approach under some circumstances.

REFERENCES

Alley, W. M., 1981, Estimation of impervious-area washoff parameters: Water Resources Research, v. 17, no. 4., p. 1161-1166.

Alley, W. M., and Smith, P. E., 1981, Estimation of accumulation parameters for urban runoff-quality modeling: Water Resources Research, v. 17, no. 6, p. 1657-1664.

______________________________, 1982, Distributed routing rainfall-runoff model--Version II: U.S. Geological Survey Open-File Report 82-344, 201 p.

Barkdoll, M. P., Overton, D. E., and Betson, R. P., 1977, Some effects of dustfall on urban stormwater quality: Water Pollution Control Federation Journal, v. 49, no. 9, p. 1976-1984.

Barnwell, T. O., 1980, Least squares estimates of BOD parameters: American Society of Civil Engineers, Journal Environmental Engineering Division, v. 106, no. EE6, p. 1197-1202.

Bruce, R. R., Harper, L. A., Leonard, R. A., Snyder, W. M., and Thomas, A. W., 1975, A model for runoff of pesticides from small upland watersheds: Journal of Environmental Quality, v. 4, no. 4, p. 541-548.

Ellis, F. W., and Sutherland, R. C., 1979, An approach to urban pollutant washoff modeling, in International Symposium on Urban Storm Runoff, July 23-26, 1979, Proceedings: University of Kentucky, Lexington, p. 325-340.

Fair, G. M., Geyer, J. C., and Okun, D. A., 1968, Water and waste water engineering: New York, NY, John Wiley and Sons, Inc.

Jennings, M. E., 1977, Downstream-upstream flow routing: U.S. Geological Survey Computer Contribution, 42 p.

Jennings, M. E., and Bauer, D. P., 1976, Determination of biochemical-oxygen-demand parameters: U.S. Geological Survey Computer Contribution, 55 p.

Kuo, C. Y., 1975, Evaluation of sediment yield due to housing construction--a case study: Department of Civil Engineering, Old Dominion University, Norfolk, Virginia

Metcalf and Eddy, Inc., University of Florida, and Water Resources Engineers, Inc., 1971, Storm Water Management Model: U.S. Environmental Protection Agency, EPA-11024 DOC 07/71, 4 volumes.

Pitt, Robert, 1979, Demonstration of nonpoint pollution abatement through improved street cleaning practices: U.S. Environmental Protection Agency, EPA-600/2-79-161, 269 p.

Rendon-Herrero, Oswald, 1974, Estimation of washload produced on certain small watersheds: American Society of Civil Engineers, Journal of the Hydraulics Division, v. 100, no. HY7, p. 835-848.

Sartor, J. D., and Boyd, G. B., 1972, Water pollution aspects of street surface contaminants: U.S. Environmental Protection Agency, EPA-R2-72-081, 236 p.

U.S. Army Corps of Engineers, 1976, Storage, treatment, overflow, runoff model (STORM): Hydrologic Engineering Center, Davis, California.

Ward, A. J., Haan, C. T., and Tapp, John, 1979, The DEPOSITS sedimentation pond design manual: Institute for Mining and Minerals Research, University of Kentucky, Lexington, 190 p.

Williams, J. R., 1975, Sediment-yield prediction with universal equation using runoff energy factor, in Present and Prospective Technology for Predicting Sediment Yields and Sources: Agricultural Research Service, ARS-5-40, p. 244-252.

_______________, 1978, A sediment graph model based on an instantaneous unit sediment graph: Water Resources Research, v. 14, no. 4, p. 659-664.

Wischmeier, W. H., and Smith, D. D., 1965, Predicting rainfall-erosion losses from cropland east of the Rocky Mountains: Agricultural Handbook No. 282, Agricultural Research Service.

ATTACHMENT A

DATA INPUT SPECIFICATIONS

Data input specifications for this program are listed below. All listing of numeric data is right justified. All listing of alphanumeric data is left justified. The letter "Oh" is written Ø to contrast with the number zero--written 0.

Experience with the program has indicated that great care must be exercised in preparing the input card deck. Definitions of most of the input variables are provide in Attachment E. Computer requirements are discussed in Attachment C. Model users are referred to the section on program debugging and interpretation (Attachment B) and the sample runs shown in Attachment G for additional assistance.

Input item	Program variable	Format	Card columns
	Card Group 1		
	Model options (1 card)		
Option to list data. If ØPTIØN=LIST, all input discharge and daily rainfall data are listed in output from program.	ØPTIØN	A4	1-4
Number of water-quality constituents simulated (1 to 4)	NWQ	I2	5-6
Option to list data used in computing measured storm-runoff loads. Code a 1 if this option is desired. Otherwise, leave blank.	NØPT	I1	7
Type of adjustment for precipitation quality: 0 = No adjustment 1 = Monthly adjustment (see card group 4) 2 = Storm-by-storm adjustment (see card group 14)	NWF	I1	8
Street sweeping option: 0 = No street sweeping 1 = Street sweeping according to fixed schedule (see card columns 13-15) 2-40 = Street sweeping on ISS user specified days (see card group 5)	ISS	I2	9-10
Spatial mode: 1 = Lumped-parameter 2 = Distributed (no transport) 3 = Distributed transport	IMØDE	I1	11
Option to list impervious-area runoff loads simulated during daily accounting. Code a 1 if this option is desired. Otherwise, leave blank.	ILØAD	I1	12

Input item	Program variable	Format	Card columns
If ISS=1 in card columns 9-10, then code the street sweeping frequency, in days.	ISSFRQ	I3	13-15
Number of records on direct access file required for storage of segment discharge data (maximum=10,000). If segment discharge data from DR_3M are not used, set JRECDS=0. Otherwise, this parameter is contained in output from DR_3M run under label, "Number of records used for direct-access file."	JRECDS	I6	16-21

Card Group 2

Water-quality labels (1 card)

Alphanumeric labels for water-quality constituents	IPA(I), I=1,NWQ	4A6	1-6 7-12 13-18 19-24

These labels are used to identify the water-quality constituents both in the program output and on subsequent input cards. Alphanumeric labels used on subsequent input cards must be identical to those coded on this card.

Card Group 3

Water-quality units (1 card)

Units for water-quality constituents (1--milligrams per liter, 2--micrograms per liter). Input in same order as constituents are listed on card group 2.	NQU(I), I=1,NWQ	4I2	1-2 3-4 5-6 7-8

Input item	Program variable	Format	Card columns

Card Group 4

Monthly adjustments for precipitation

(1 card for each water-quality constituent)

(Skip this card group, if NWF not equal to 1 on card group 1.)

Input item	Program variable	Format	Card columns
Alphanumeric label for water-quality constituent J	IPA(J)	A6	1-6
Precipitation concentration, in milligrams per liter, if NQU(J)=1; in micrograms per liter, if NQU(J)=2.			
January	WFALL(J,1)	F5.2	7-11
February	WFALL(J,2)	F5.2	12-16
March	WFALL(J,3)	F5.2	17-21
April	WFALL(J,4)	F5.2	22-26
May	WFALL(J,5)	F5.2	27-31
June	WFALL(J,6)	F5.2	32-36
July	WFALL(J,7)	F5.2	37-41
August	WFALL(J,8)	F5.2	42-46
September	WFALL(J,9)	F5.2	47-51
October	WFALL(J,10)	F5.2	52-56
November	WFALL(J,11)	F5.2	57-61
December	WFALL(J,12)	F5.2	62-66

Card Group 5

Street sweeping days (1 card)

(If ISS is less than 2 on card group 1, then skip to card group 6.)

Input item	Program variable	Format	Card columns
Dates on which streets were swept, coded as:			
month, day, and year	NSSDAY(1,I), NSSDAY(2,I), and NSSDAY(3,I), I=1, ISS	10(I3,I2,I2)	1-7 8-14 9-21 etc.

These dates must be in chronological order (10 dates per card up to a maximum of 40 dates). Use 2 digits for month, 2 for day and 2 for year.

Input item	Program variable	Format	Card columns
	Card Group 6		
	Discharge station (1 card)		
Discharge station number	STAD	I8	1-8
Name of discharge station	TITLD	50A1	9-58
Drainage area of basin (square miles)	DA	F6.2	59-64
	Card Group 7		
	Daily rainfall station (1 card)		
Daily rainfall station number	STAP	I8	1-8
Name of daily rainfall station	TITLP	50A1	9-58
	Card Group 8		
	Period of record (1 card)		
Beginning year,	IYR	I3	21-23
month, and	IMØ	I3	24-26
day of record	IDY	I3	27-29
Ending year,	EYR	I3	33-35
month, and	EMØ	I3	36-38
day of record	EDY	I3	39-41

Impervious-area land-surface loads are set equal to zero at the start of simulation. Therefore, the beginning day of record should be one to two months prior to the first unit day. The ending day of record should be at least 1 day after the last unit day. Use 2 digits for year, 2 for month and 2 for day.

Input item	Program variable	Format	Card columns
	Card Group 9		
	Time interval (1 card)		
Time interval of output, in minutes	PTIME	F5.0	1-5

Input item	Program variable	Format	Card columns
Code a 1 if unit discharge data are input to the program using card group 11 or are read from file JPUN. Otherwise, leave blank.	IUDATA	I5	6-10
File number where card group 11 is stored. Leave blank if card group 11 is read from cards or IUDATA=0. JPUN will then be automatically set to 5.	JPUN	I5	11-15

PTIME is restricted to one of the following values (in minutes): 1, 2, 3, 4, 5, 10, 15, 30, 45, or 60. If unit discharge data are input to the model using card group 11; then, PTIME should be the same as the time interval of the unit discharge data. If DR_3M-QUAL is linked with DR_3M, then PTIME should be the same for both models.

If unit discharge data are input to the model, they can either be included directly in the input deck or read from file JPUN. The latter case may occur if a previous run of DR_3M was used to simulate the unit discharge data. See card group 1 of DR_3M manual and discussion of JPUN in Attachment C for additional guidance.

Card Group 10

Storm-runoff dates

(If IUDATA=1 on card group 9, skip to card group 11.)

Input item	Program variable	Format	Card columns
Discharge station number (same as on card group 6)	STAD	I8	1-8
Dates on which storm-runoff loads are to be simulated:			
year	YR	I2	9-10
month	MØ	I2	11-12
day	DY	I2	13-14

Card group 10 is used to specify dates on which storm-runoff loads are simulated. A card for each unit day should be included. For example, suppose a period of storm runoff that occurred between May 1, 1978, and May 2, 1978, is to be simulated. Then, two cards are necessary, one for May 1, 1978, and one for May 2, 1978. All cards should be arranged in chronologic order.

At the end of card group 10, insert a card with a 9 punched in column 80.

Input item	Program variable	Format	Card columns

Card Group 11

Cards for unit discharge data

(If IUDATA=0 on card group 9, skip to card group 12.)

One of two different formats are used in coding data, depending on the value of PTIME on card group 9. If PTIME is less than 5.0 minutes, use format 11a. If PTIME is greater than or equal to 5.0 minutes, use format 11b.

Format 11a

Input item	Program variable	Format	Card columns
Discharge station number (same as on card group 6)	STAD	I8	1-8
Date on which discharge occurred:			
year	YR	I2	9-10
month	MØ	I2	11-12
day	DY	I2	13-14
Time interval, in minutes (must equal PTIME on card group 9).	CT	I2	15-16
Card sequence number	CN	I3	17-19
Discharge, in cubic feet per second (12 data items per card)	UD	12F5.0	20-79
Data type (CØDE=2 in column 80)	CØDE	I1	80

Format 11b

Input item	Program variable	Format	Card columns
Discharge station number (same as on card group 6)	STAD	I8	1-8
Date on which discharge occurred:			
year	YR	I2	9-10
month	MØ	I2	11-12
day	DY	I2	13-14
Time interval, in minutes (must equal PTIME on card group 9).	CT	I2	15-16

Input item	Program variable	Format	Card columns
Card sequence number	CN	I2	17-18
Discharge, in cubic feet per second (12 data items per card)	UD	12F5.0	19-78
Data type (CØDE=2 in column 80)	CØDE	I1	80

Card group 11 is used to enter the unit discharge (UD) data to the model. The card format for listing UD provides 12 fields for these data. Each set of 12 units of data is numbered in chronologic sequence by variable CN. The UD array is initialized to zero. If all 12 units of data for UD are zero, the card may be omitted from the input card deck. However, its card sequence number for this day must be taken into account in listing CN on subsequent cards. At least one card should be included for each unit day specified on card group 12. The sample runs in Attachment G can be referred to for guidance in coding this card group.

At the end of unit discharge data, insert a card with a CØDE of 9 punched in column 80. If card group 11 is read from file JPUN (specified on card group 9), then this final card should also be included in file JPUN.

Card Group 12

Cards for daily rainfall data

Input item	Program variable	Format	Card columns
Daily rainfall station number (same as card group 7)	STAP	I8	1-8
Year and month for data	YR MØ	I2 I2	9-10 11-12
Card sequence number (1 or 2)	CN	I1	13
Daily rainfall in inches (up to and including 16 items per card)	DP	16F4.2	14-77
Data type (CØDE=3 in column 80)	CØDE	I1	80

Two cards are required for listing a complete month of daily precipitation data. Use as many cards as necessary to list data for all months. The card format for listing these daily data provides 16 fields: the first 16 days of data are listed on the first card, identified by the card sequence number CN=1, and the remaining days of data in the month on the second card (CN=2). For a unit day, insert a negative number as the daily rainfall. A negative number for daily rainfall signals the model to perform detailed storm simulation for that day.

Input item	Program variable	Format	Card columns

It may be desirable to skip a large gap in time rather than continue with daily accounting (for example, no winter records). In such cases, a 9999 should be punched as the daily rainfall for the first and last day of the gap in record. No daily rainfall cards are required for intervening days. Land-surface loads of impervious areas are set equal to zero immediately following a gap in the daily precipitation record. Therefore, the model should be run for 1 to 2 months on a daily accounting basis between the end of a gap in record and the first subsequent unit day.

At the end of the daily precipitation data, insert a card with a CØDE of 9 punched in column 80.

Card Group 13

Storm-sequencing card(s)

Number of storms	I	I2	1-2
Number of storms in the continuous sequence of storm days containing a given storm.	NF(K) K=1, I	39I2	3-4 5-6 etc.

The following example should assist in explaining card group 13: Suppose eight storms are to be simulated by the model. These storms occur on the following days:

Storm Number	Date
1	March 1, 1976
2	March 1, 1976
3	May 20, 1976
4	June 1, 1976
5	June 1-2, 1976
6	June 2, 1976
7	April 1, 1977
8	April 2, 1977

Then, the following numbers would be punched on the card representing card group 13:

Card Column:	2	4	6	8	10	12	14	16	18
Number:	8	2	2	1	3	3	3	2	2

Notice that the number of storms in a set of storm days is entered as many times as there are storms in the set.

Input item	Program variable	Format	Card columns

Card Group 14

Storm-identification

A set of cards representing card groups 14-17 are input to the model for each storm to be simulated. An example of the program deck setup for card groups 14-17 is shown in figure 12. Card groups 15-17 are skipped, if no measured water-quality data are input to the model.

Input item	Program variable	Format	Card columns
Starting time increment for storm	KS	I4	1-4
Ending time increment for storm	KE	I4	5-8
Number of water-quality constituents for which measured concentrations are input to the program. NWQP can range from 0 to 4 and can be different for different storms.	NWQP	I2	9-10
Precipitation concentrations for storm. These should be in same order as constituent labels are read in card group 2 and should be in units specified in card group 3. Precipitation concentrations should only be coded on this card group, if NWF=2 on card group 1.	WFALL(J,I1), J=1,NWQ	4F5.2	11-15 16-20 21-25 26-30

There should be one storm-separation card for each of the I storms shown on the storm-sequencing card. Starting and ending time increments are specified as the number of the time interval in the sequence of days containing the storm. The value of KS or KE can be calculated using the following formula:

$$\text{KS or KE} = \frac{[60 \cdot \text{HR} + \text{MIN} + 1440 \cdot (\text{NDSD}-1)]}{\text{PTIME}}$$

where HR is the hour of the day (from 0 to 24), MIN is the minutes past the hour, and NDSD is the number of the storm day in the sequence of storm days. For example, if the time interval is 15 minutes and the starting time of a

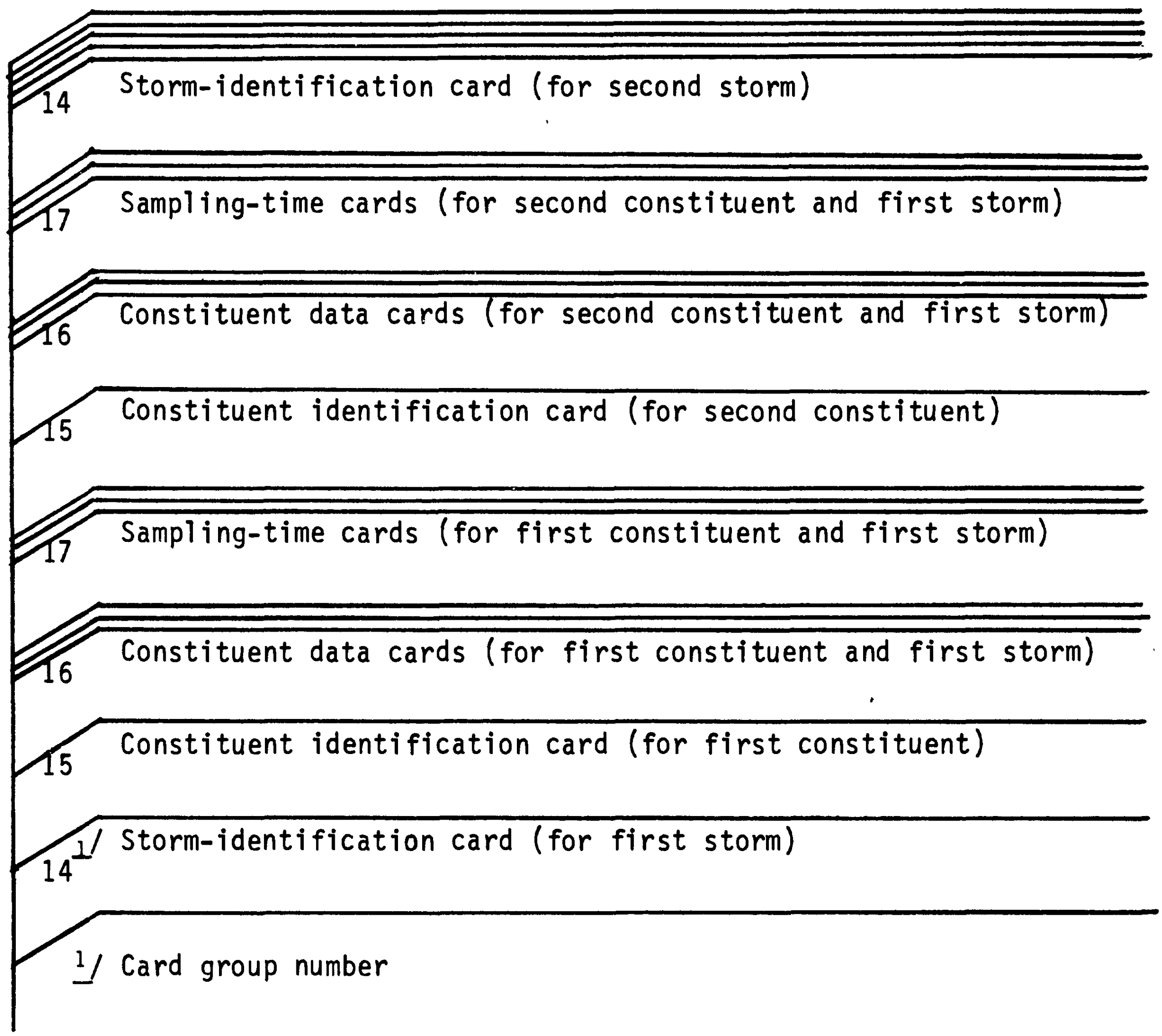


Figure 12.--Example program deck setup for card groups 14-17.

Input item	Program variable	Format	Card columns

storm is 0700 on the first day of a sequence of storm days, KS should be specified as 28. Likewise, if the starting time was 0700 on the second day of a sequence of storm days, KS should be specified as 124. Other examples are shown in table 1.

The model assumes that the storm started PTIME minutes prior to KS. Runoff and constituent concentrations at that time are assumed equal to zero. If discharge data from DR_3M are used by the model, then the values of KS and KE should be the same as in the DR_3M run.

Table 1.--Example for specifying KS and KE

				PTIME = 1.0 minutes		PTIME = 5.0 minutes	
Storm Number	Data	Starting time (24-hour)	Ending time (24-hour)	KS	KE	KS	KE
1	March 1, 1976	0700	1115	420	675	84	135
2	March 1, 1976	1305	1610	785	970	157	194
3	May 20, 1976	1205	1425	725	865	145	173
4	June 1, 1976	0010	0555	10	355	2	71
5	June 1-2, 1976	2310	0105	1390	1505	278	301
6	June 2, 1976	0810	0955	1930	2035	386	407
7	April 1, 1977	1015	1235	615	755	123	151
8	April 2, 1977	1055	1400	2095	2280	419	456

Card Group 15

Constituent identification

(If no measured concentrations are input to model, then card groups 15-17 are not needed.)

Alphanumeric identification label for water-quality constituent (J) with measured data input to program. This label must correspond to one of the labels specified in card group 2.	IPA(J)	A6	1-6
Number of water-quality measurements for storm I and constituent J. (A maximum of 24)	NWQM(I,J)	I2	7-8

Input item	Program variable		Card columns

Card Group 16

Constituent data for storm

Input item	Program variable		Card columns
Measured constituent concentrations for storm I and constituent J. Units should be in milligrams per liter or micrograms per liter, depending on the specification on card group 3.	CØ(I,J,N), N=1,NWQM(I,J)	10F8.2	1-8 9-16 etc.

Card Group 17

Sampling times for storm

Input item	Program variable		Card columns
Time when water-quality samples were collected, in minutes since start of storm	TIME(I,J,N), N=1,NWQM(I,J)	10F8.2	1-8 9-16 etc.

Care should be exercised to assure that the time values for KS on card group 14 and the sampling times on card group 17 are synchronized. Remember that the model assumes a storm started PTIME minutes prior to KS. A check of the data input to the program on card groups 16 and 17 can be obtained by setting NØPT=1 on card group 1.

Card Group 18

Output card

Input item	Program variable		Card columns
0 = no listing of simulated concentrations for storm 1 = simulated concentrations are listed for storm	KØUT(I) I=1,NØFE	40I2	1-2 3-4 etc.

Card Group 19

Plotting card

Input item	Program variable		Card columns
0 = no plotting of water-quality data for storm 1 = instantaneous concentrations are plotted for storm	IPL(I) I=1,NØFE	40I2	1-2 3-4 etc.

Input item	Program variable	Format	Card columns
2 = both instantaneous concentrations and load characteristic curves are plotted for storm			
3 = load characteristic curves are plotted for storm			

All plots include measured water-quality data when they are input to program. If measured concentrations are input to program, then load characteristic curves are for the portion of runoff between the first and last sample.

Card Group 20

Catchment-data card

Input item	Program variable	Format	Card columns
Effective impervious area of watershed, in acres	DAE	F5.2	1-5
Maximum impervious retention, in inches	AIMP	F5.2	6-10
Fraction of effective impervious area of watershed that is swept by street sweepers. (Only code a value for SSAREA, for a lumped-parameter simulation)	SSAREA	F5.2	11-15

Card Group 21

Land-use card

Input item	Program variable	Format	Card columns
Number of land-use types in watershed (a maximum of 4). NLU must equal 1 for a lumped-parameter run.	NLU	I5	1-5
Alphanumeric designation for land-use type 1	LUSE(1)	3A3	6-14

Input item	Program variable	Format	Card columns
Alphanumeric designation for land-use type 2 (if NLU greater than 1)	LUSE(2)	3A3	15-23
Alphanumeric designation for land-use type 3 (if NLU greater than 2)	LUSE(3)	3A3	24-32
Alphanumeric designation for land-use type 4 (if NLU=4)	LUSE(4)	3A3	33-41

Card Group 22

Street sweeping effectiveness (1 card)

(Skip to card group 24, if ISS=0 on card group 1)

Input item	Program variable	Format	Card columns
Fraction of initial land-surface load in excess of base residual removed by street sweeping (data are listed for each constituent in same order as card group 2).	SSEFF(J), J=1,NWQ	4F5.2	1-5 6-10 11-15 16-20

Card Group 23

Street sweeping data (1 card)

Input item	Program variable	Format	Card columns
Base residual land-surface load which cannot be removed by street sweeping, in pounds per acre of effective impervious area (data are listed for each constituent in same order as on card group 2).	SSMIN(J), J=1,NWQ	4F5.2	1-5 6-10 11-15 16-20

Input item	Program variable	Format	Card columns

Card Group 24

Segment-control card

If this is a lumped-parameter run (IMØDE=1 on card group 1), skip to card group 33).

Input item	Program variable	Format	Card columns
Total number of segments in DR_3M run	NSEG	I5	1-5
Values of a and b for each water-quality constituent (see equation 15) for pervious-area runoff. The values of a and b should be listed for the first constituent on card group 2 followed by the values of a and b for the second constituent, etc. If no pervious segments are used, then no values for a and b should be coded.	AKA(J), AKB(J), J=1,NWQ	4(2F5.0)	6-10 and 11-15 16-20 and 21-25 26-30 and 31-35 36-40 and 41-45

Card Group 25

Segment characteristics

There is one card for each segment. Cards must be arranged in the same order as the computational sequence shown on the output from DR_3M.

With the following restrictions, DR_3M-QUAL will handle any segmentation allowed by DR_3M:

1. Input hydrograph and input discharge points should have no upstream segments.

2. DR_3M-QUAL handles a maximum of 5 reservoir segments.

3. The same channel segment cannot be used more than once, if IMØDE=2 on card group 1.

Input item	Program variable	Format	Card columns
Alphanumeric identification for segment (Required for all segments; any alphanumeric identification can be used.)	ISEG(I)	A4	1-4
Alphanumeric identification for up to 3 segments which contribute inflow to the upstream end of this segment (leave blank where upstream segments are not present.)	IUP(I,J) J=1,3	3A4	5-8 9-12 13-16
Alphanumeric identification for up to 4 segments which contribute uniform lateral inflow into this segment (leave blank where lateral inflow segments are not present.)	ILAT(I,J) J=1,4	4A4	17-20 21-24 25-28 29-32
Type of segment 1 = channel segment 2 = overland-flow segment 3 = reservoir segment 4 = nodal segment	ITYPE(I)	I2	33-34
Outflow print indicator = 1 if the outflow concentrations for this segment are to be printed in output = 0 if the outflow concentrations for this segment are not to be printed in output	IPR(I)	I2	35-36

The value of IPR(I) is ignored by the model in three instances. First, outflow concentrations from overland-flow segments cannot be listed. Second, the decision to list outflow concentrations from the most downstream segment is made on a storm event basis depending on the values input to the model for the KOUT array on card group 18. Third, if IMØDE=2 simulated concentrations will not be listed.

Input item	Program variable	Format	Card columns
Length of segment (feet). Leave blank for segment types 3 or 4	FLGTH(I)	F7.0	37-43
A set of parameters which depend on type of segment	PARAM(I,J), J=1,4	4F5.0	44-48 49-53 54-58 59-63

Segment Type	Parameter Definitions
1	PARAM(I,1) = kinematic wave parameter (α) PARAM(I,2) = kinematic wave parameter (m) PARAM(I,3) = PFAC1 in equation 16 PARAM(I,4) = PFAC2 in equation 16
2	PARAM(I,1) = fraction of effective impervious area that is swept by street sweepers (0.0 to 1.0). Leave blank, if street sweeping is not simulated. PARAM(I,2) = effective imperviousness (0.0 to 1.0). If an overland-flow plane has been split into a pervious and impervious segment, then PARAM(I,2) should be the effective imperviousness of the flow plane for the impervious segment and PARAM(I,2) should be 0.0 for the pervious segment. PARAM(I,3) = leave blank PARAM(I,4) = leave blank
3	leave blank
4	PARAM(I,1) = concentration in input flow of 1st constituent on card group 2 PARAM(I,2) = concentration in input flow of 2nd constituent on card group 2 PARAM(I,3) = concentration in input flow of 3rd constituent on card group 2 PARAM(I,4) = concentration in input flow of 4th constituent on card group 2

Input item	Program variable	Format	Card columns
Alphanumeric designation for land use of lateral inflow segments. Leave blank, if ITYPE(I) is <u>not</u> equal to 1. A land-use type should be designated for all type 1 (channel) segments, even if there are no adjacent lateral inflow segments.	IP1	A9	64-72

Input item	Program variable	Format	Card columns

Card Group 26

Sediment and water characteristics (1 card)

(If no reservoir segments are simulated, skip to card group 33.)

Input item	Program variable	Format	Card columns
Number of sizes in particle-size distribution (a maximum of 10.)	NS	I5	1-5
Specific gravity of sediment. If no value is entered, a default value of 2.65 is assigned.	SG	F5.0	6-10
Viscosity of the flow in cm^2/s. (Default = 0.0114 cm^2/s).	VISCØS	F5.0	11-15

Card Group 27

Particle sizes (1 card)

Input item	Program variable	Format	Card columns
Particle sizes, in microns, corresponding to values of PERCNT on card group 29. Note that the same particle sizes are used for all reservoir segments and all constituents. Particle sizes should be listed in ascending order of magnitude.	SIZE(I), I=1,NS	10F8.0	1-8 9-16 etc.

Input item	Program variable	Format	Card columns

Card Group 28

Reservoir data (1 card)

Card groups 28 to 32 are input to the model for the first reservoir segment read-in using card group 26. A second set of these card groups is then input for the second reservoir (if applicable) read-in card group 26. This process is repeated for all reservoir segments.

Input item	Program variable	Format	Card columns
Number of plug layers. Always set NLAY equal to 1.	NLAY	I5	1-5
Set of JFLØW = 1.	JFLØW	I5	6-10
Number of coordinates on stage-area and stage-discharge curve. (A maximum of 10.)	N	I5	11-15
Dead storage, in acre-ft. Dead storage should not exceed permanent pool capacity.	DEAD	F5.0	16-20

Card Group 29

Cumulative frequency curve (1 card for each water-quality constituent)

Input item	Program variable	Format	Card columns
Percent finer values on cumulative frequency curve corresponding to values of SIZE on card group 27. Percent values should be listed in ascending order of magnitude. The first value should be 0.0 and the last value 100.	PERCNT(I), I=1,NS	10F8.0	1-8 9-16 etc.

Input item	Program variable	Format	Card columns
Card Group 30			
Stage data (1 card)			
Stage values, in ft, at the basin outlet used to determine the stage-area and stage-discharge curves. Values of STAGE should be listed in ascending order of magnitude. The first value should be 0.0.	STAGE(I), I=1,N	10F8.0	1-8 9-16 etc.
Card Group 31			
Surface-area data (1 card)			
Surface areas, in acres, corresponding to stage values on card group 30.	AREA(I), I=1,N		1-8 9-16 etc.
Card Group 32			
Discharge data (1 card)			
Discharges, in ft^3/s, corresponding to stage values on card group 30. The first value for DISCH should be 0.0. Values of DISCH should be input in increasing order. No two values should be the same. If the reservoir has a permanent pool, then the second STAGE value should be the stage of the permanent pool and the second DISCH value should be a number greater than 0.0 but less than 0.005 ft^3/s.	DISCH(I), I=1,N	10F8.0	1-8 9-16 etc.

Input item	Program variable	Format	Card columns

Card Group 33

Impervious area accumulation/washoff parameters

One card is required for each combination of land use and water-quality parameter (i.e., NWQ x NLU cards are required).

Input item	Program variable	Format	Card columns
Alphanumeric designation for land use	IP1	A9	1-9
Alphanumeric designation for water-quality parameter	IPA(J)	A6	10-15
Impervious area accumulation/washoff parameters:			
K_1, in pounds per acre of effective impervious area	BK(1,J,I)	F8.3	16-23
K_2, in $days^{-1}$	BK(2,J,I)	F8.3	24-31
K_3, in $inches^{-1}$ if equation 5 is used for washoff or K_{3*}, in $hr/inch^2$ if equation 9 is used. If K_{3*} is input, then it should be a negative number.	BK(3,J,I)	F8.3	32-39
K_4, in hour per inch. If the availability factor (equation 8) is not used, then leave K_4 blank.	BK(4,J,I)	F8.3	40-47
K_3 for daily accounting in $inches^{-1}$	BK(5,J,I)	F8.3	48-55

If a value of K_4 is input to the model or equation 9 is used for washoff, then a value of K_3 for daily accounting should be input. This value of K_3 may differ from that used during unit days, since only equation 5 without the availability factor is used for daily accounting. If the value of K_3 is to be the same for both unit and daily days, then card columns 48-55 can be left blank.

ATTACHMENT B

PROGRAM DEBUGGING AND INTERPRETATION

Experience with the program has indicated that great care must be exercised in preparing the input card deck. The time and effort used to carefully prepare and check an input data deck may save considerable frustration later when using the model.

Even with painstaking effort, some errors may occur. Many diagnostic messages are contained within the program in the event of errors. Most of the input data are output by the program soon after being read. Hence, if a failure occurs, where the program is located in outputting data will often give a clue as to the location of the error in the input data. This is particularly true of the unit discharge data. For this reason, it is highly recommended that ØPTIØN = LIST on card group 1 during program debugging. If measured concentrations are input to the program, then NØPT on card group 1 should be set equal to 1 during program debugging. This will result in a list of the data used in computing measured storm-runoff loads and will serve as a check on the data input on card groups 16 and 17.

If erroneous data are input to the program, errors may occur in the program output even though the program appeared to run correctly. For example, the impervious retention might be mistakenly read into the program as 0.5 rather than the intended value of 0.05. These types of errors can be identified by carefully checking much of the output against the data that are assumed to be input to the program. Particularly important items to check include the "Header Records from the Runoff File," the segment characteristics, and the "Summary of Measured Data."

Data entitled "Header Records from the Runoff File" is output by DR_3M-QUAL if a link is made with DR_3M. This information contains a number of descriptors of the DR_3M run including the segments used in DR_3M (listed in computational order). Also included on the second page of this output is a listing of the storms for which the segment flow data are available. These storms should correspond to those simulated by the DR_3M-QUAL run.

For distributed DR_3M-QUAL runs the segment data (card group 26) are included in the model output. The order of listing of these segments should be compared to the segment order listed under "Header Records from the Runoff File" to assure that the segments are ordered the same on both lists.

If measured unit discharge data are input to the model (card group 11), then the measured runoff volume and peak discharge are output for each storm. If measured concentration data are also input to the program, then measured loads are also output. Characteristics describing the extent of sampling of each storm-runoff period are also output, as described below.

Field or automatic sampling logistics result in the first sample of storm runoff being collected sometime after the start of the runoff and the last sample collected before the end of the runoff period. Often a considerable portion of the total storm runoff may not be included in runoff occurring between the first and last sample. Therefore, if measured concentrations are input to DR_3M-QUAL, storm-runoff loads are based on the runoff occurring between the first and last samples collected for the runoff period.

The method of computation of measured storm-runoff loads is to estimate a constituent concentration corresponding to each discharge measurement. This is done by linear interpolation between measured concentrations. Each discharge measurement is then multiplied by its corresponding constituent concentration and an appropriate conversion factor to compute instantaneous loads. These are then integrated to determine storm-runoff loads. An example of this method of computing loads is illustrated in figure 13. Note that because the first sampling time in figure 13 did not have a measured discharge value, the first discharge value prior to this time is included in the load computation along with the assumption that the constituent was equal to the first measured concentration. A similar approach is taken at the end of the sampling period. The runoff volume used in load computations and total storm-runoff volumes are reported in the program output.

If measured concentrations are input to DR_3M-QUAL, then simulated loads reported by the model are also based on the runoff occurring between the time the first sample was collected and the time the last sample was collected. This enables a valid comparison between simulated and measured loads.

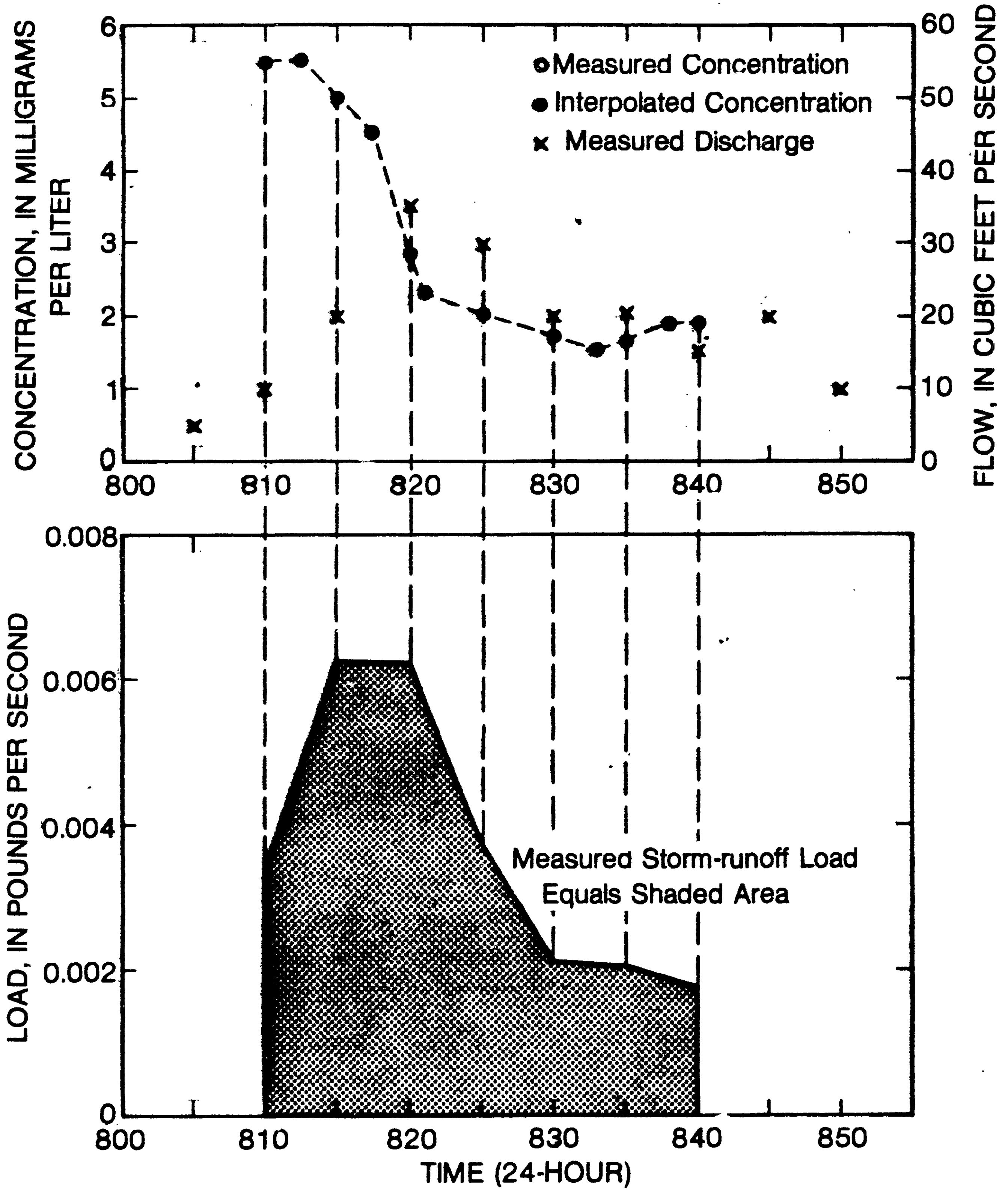


Figure 13.--Computation of measured storm-runoff loads.

ATTACHMENT C

COMPUTER REQUIREMENTS

The computer program was written in FØRTRAN IV programming language. The program, as dimensioned, will handle 60 storms comprising at most 180 unit days and 7,310 days (20 years) of record. As many as four water-quality constituents can be run simultaneously by the model. Several of the program's limits can be changed easily by redimensioning the program. Three examples are presented below.

Period of record: The maximum period of record simulated by the program can be changed by setting NDYS to the desired maximum period of record in days (see line A 320 in Attachment F), and by changing the array size of DP to the value of NDYS.

Number of unit time intervals: If PTIME on card group 9 is always 5 minutes or greater, then the UD array can be reduced from 2881 to 1441 and IUD (see line A 340 in Attachment F) should be set to 1441.

Number of water-quality constituents: Extending the program's capability to handle more than four constituents would require many changes to the source code. However, if less than four constituents are to be run simultaneously then the core storage requirements of the program can be reduced by changing the QWUP(4,1442) and QWLAT(4,1442) arrays to QWUP(NWQ,1442) and QWLAT(NWQ,1442) where NWQ is the number of water-quality constituents.

JCL Information for Geological Survey Computer

The load module for DR_3M has been stored in the partitioned data set AG4254J.URBAN.LMOD under member name Q347. It resides on WRD system disk CCD810. To execute the program on the USGS Amdahl[3] computer, use the JCL cards shown as follows:

Job Card

```
//  EXEC PGM=Q347,REGION=340K
//STEPLIB DD DSN=AG4254J.URBAN.LMOD,DISP=SHR
//FT06F001 DD SYSOUT=A,DCB=(RECFM=FBA,LRECL=133,BLKSIZE=3458)
//FT25F001 DD DSN=Azzzzzz.aaaaaaaa,DISP=SHR
//FT26F001 DD UNIT=SYSDK,DISP=(,PASS),
// SPACE=(11520,(60,10),RLSE)
//FT27F001 DD UNIT=SYSDK,DISP=(,PASS),DCB=DSORG=DA,
// SPACE=(480,(?????,10),RLSE)
//FT05F001 DD *
```

Data Cards

```
/*
//
$$$
```

3/ The use of brand names in this report is for identification purposes only and does not imply endorsement by the U.S. Geological Survey.

where

FT05F001	is a card reader,
FT06F001	is a printer,
FT25F001	is a semipermanent work file where the segment discharge data are located,
FT26F001	is a temporary work file for unit discharge data, and
FT27F001	is a temporary work file for the segment concentration data.

File 26 is a sequential file on magnetic disk and is used for temporary storage during program execution. The space defined for this file is sufficient to provide storage for any possible run, with the program's present dimension. Files 25 and 27 are direct-access files which require special attention when setting up JCL for a DR_3M run.

The FT25 and FT27 cards are required for distributed runs of DR_3M-QUAL only. For lumped-parameter simulations they can be dropped from the JCL. For distributed (no transport) runs, only the FT25 card is required in the JCL.

FT25 Card

The FT25 card is simply coded as

```
//FT25F001 DD DSN=Azzzzzz.aaaaaaaa,DISP=SHR
```

where

zzzzzz	are the six characters of the account name where the segment flow files have been stored
aaaaaaaa	is the 1 to 8 character name used to designate the name of the data set during the DR_3M run.

FT27 Card

During distributed transport runs, simulated concentrations from each model segment and for each constituent must be temporarily stored. Storage is by means of a direct access file. The direct access--rather than sequential--organization is necessary because file records are accessed in a nonconsecutive sequence that is defined by the ordering and re-using of segments.

The direct access data set is defined by a DEFINE FILE statement in the program and by the FT27 card in the JCL string. The DEFINE FILE statement and the FT27 card each indicate the amount of space that is required to store the data set. The form of the DEFINE FILE statement is as follows:

```
DEFINE FILE 27(?????,480,L,IRECD)
```

where ????? is the number of records in the data set and 480 is the number of bytes (characters) per record. Parameters L and IRECD are standard descriptors and do not vary. The 27 establishes the connection between the program and the FT27 card in the JCL.

The program (subroutine FILE27) contains many DEFINE FILE statements of the form given above with ????? defined between 50 and 5,000. This range of record numbers is provided to accomodate users that might have very different storage requirements. The model selects the appropriate DEFINE FILE statement for each run. However, the user is responsible to specify the value of ????? on the FT27 card in JCL. For a particular model run, ????? on the FT27 card, hereafter referred to as JRECQW, defines the number of records available for storing segment concentration data. In order that JRECQW be compatible with the DEFINE FILE statements in subroutine FILE27, it should be greater than or equal to the number of records specified in the DEFINE FILE statement.

JRECQW can be computed by first observing the DR_3M run used to create the segment flow files. Letting NRECDS be the maximum value (for the routed storms), output by DR_3M under the heading "Records required for routing," QRECDS is then NWQ*NRECDS, where NWQ is the number of water-quality constituents that are simulated. JRECQW is then simply determined by rounding up QRECDS to the nearest 100 if less than 500 or to the nearest 500 if greater than 500.

JPUN on Card Group 9

JPUN on card group 9 should normally be left blank. However, it may be desired to use the outlet hydrographs simulated by DR_3M as the discharge data for a lumped-parameter DR_3M-QUAL run. This can be achieved by setting JPUN to the file where the outlet hydrographs are stored and to include a card in the JCL to define the sequential file where the values are stored. The unit discharge data will then be read from this file.

Considerations for Other Computer Systems

With the exception of the plotting routine and the direct access files, the program will run on most computers with sufficient core storage. The [3/] plotting routing included in the program listing (Attachment F) is IBM-System dependent. This plotting routine can be eliminated by removing all lines from subroutine PLT except AG 10, AG 70, AG 370 and AG 420 and removing subroutine PRPLOT (lines AH 10 through AH 2000).

Subroutines FILES and FILE27 contain DEFINE FILE statements for the direct access files. Since the direct access file organization is always unique to a particular computer system, subroutine FILES and FILE27 may require reprogramming if the program is used at a computer system other than the USGS. On an IBM system the program will only compile without revision using the FORTRAN G level compiler because of the use of multiple DEFINE FILE statements.

To the authors' knowledge the only extensions beyond the ANSI standard, in subroutines other than the three given, is the use of mixed-mode expressions and the T format code.

3/ The use of brand names in this report is for identification purposes only and does not imply endorsement by the U.S. Geological Survey.

ATTACHMENT D

DETENTION STORAGE PROGRAM

Subroutine RESVR which simulates the effects of detention storage can be removed from DR_3M-QUAL and used separately in conjunction with a simple main program consisting of the following 4 lines:

```
CALL RESVR(1,1)
CALL RESVR(3,1)
STOP
END
```

This may facilitate calibration of a detention reservoir segment, if measured data are available on the quality and quantity of inflow and outflow from a given reservoir.

When used separately the program is limited to 1 reservoir, 1 water-quality constituent and 1 storm-runoff period for a given run. Influent discharges and concentrations and effluent discharges are part of the input requirement of the model. A program documented by Jennings (1977) can be used to determine effluent discharges, if only influent discharges are available.

Input requirements for this program are as follows:

Input item	Program variable	Format	Card columns
	Card Group 1		
	Time parameters (1 card)		
Number of time intervals (maximum of 1442)	ICT	I5	1-5
Time interval, in hours	DELTAT	F5.0	6-10
Time interval for routing and output of results, in hours	DELPLG	F5.0	11-15

DELPLG should be a multiple of DELTAT and ICT should be divisible by the ratio DELPLG/DELTAT an integer number of times.

Input item	Program variable	Format	Card columns
	Card Group 2		
	Alphanumeric designations (1 card)		
Alphanumeric label for detention pond	ISEG	A4	1-4
Alphanumeric label for water-quality constituent	IPA, IPB	2A3	5-10

Card Groups 3-9

Input specifications are the same as card groups 26 to 32 as described in Attachment A.

Input item	Program variable	Format	Card columns
	Card Group 10		
	Cards for influent discharges		
Inflow to reservoir in cubic feet per second	FUP(I), I=1,ICT	10F8.0	1-8 9-16 etc.
	Card Group 11		
	Cards for effluent discharges		
Outflow from reservoir in cubic feet per second	FLW(I), I=1,ICT	10F8.0	1-8 9-16 etc.
	Card Group 12		
	Cards for influent concentrations		
Concentrations in inflow to reservoir, in milligrams per liter	QWUP(I), I=1,ICT	10F8.0	1-8 9-16 etc.

When used as a separate program, the output includes an estimate of basin trap efficiency. The user should note that this trap efficiency is based on the simulated effluent load. Thus, the basin should be simulated over a sufficient period of time for the majority of effluent to be discharged in order to obtain a precise estimate of basin trap efficiency.

ATTACHMENT E

DEFINITIONS OF SELECTED VARIABLES

(A) Alphanumeric, (I) Integer, (R) Real

AIMP -- Impervious retention storage capacity. (R)
BK -- Array of impervious-area accumulation and washoff parameters. (R)
CN -- Card sequence number for various data types. (I)
CØ -- Array of measured concentrations. (R)
CØDE -- Identifier of data type. (I)
CØN -- Array of constituent concentrations at downstream end of segment. (R)
DA -- Watershed drainage area, in square miles. (R)
DAE -- Effective impervious area of watershed, in acres. (R)
DP -- Array containing daily rainfall data. (R)
EDY -- Ending day of record. (I)
EMØ -- Ending month of record. (I)
EYR -- Ending year of record (last two digits). (I)
FLGTH -- Array of segment flow lengths. (R)
FLW -- Array of flows at downstream end of segment. (R)
FUP -- Array of flows at upstream end of segment. (R)
IDY -- Beginning day of record. (I)
ILAT -- Array of segments contributing lateral inflow to indexed
IMØDE -- Indicator of model spatial structure. (I)
IPA -- Label for water quality constituent. (A)
IPL -- Array of indicators of outflow hydrograph printing for segments. (I)
ISEG -- Segment label. (A)
ISS -- Street sweeping option. (I)
ISSFRQ-- Street sweeping frequency, in days. (I)
ITYPE -- Array of segment types. (I)
IUP -- Array of segments contributing upstream inflow to indexed segment. (A)
IYR -- Beginning year of record (last two digits). (I)
JPUN -- File number where outflow hydrographs from DR_3M are stored. (I)
KE -- Ending unit time interval for storm. (I)
KØUT -- Array specifying storms for which simulated concentrations are listed. (I)
KS -- Starting unit time interval for storm. (I)
LUSE -- Array of land-use types. (A)
N -- (a) Number of coordinates on stage-area-discharge curve.
(b) Counter. (I)
NF -- Number of storms in sequence of days containing a given storm. (I)
NLU -- Number of land-use types. (I)
NØFE -- Number of runoff periods simulated. (I)
NQU -- Concentration units for water-quality constituents. (I)
NS -- Number of sizes in particle-size distribution. (I)
NSEG -- Number of segments for distributed run. (I)
NSSDAY-- Array of dates on which streets were swept. (I)
NWF -- Type of adjustment for precipitation quality. (I)
NWQ -- Number of water-quality constituents simulated. (I)
NWQM -- Number of water-quality measurements for a given storm and constituent. (I)
NWQP -- Number of water-quality constituents for which measured concentrations are input to program for a storm. (I)

PTIME -- Time interval of output, in minutes. (R)
QWLAT -- Array of constituent concentrations in lateral inflow to a segment. (R)
QWUP -- Array of constituent concentrations at upstream end of segment.
SSAREA-- Fraction of effective impervious area of watershed that is swept by street sweepers. (R)
TIME -- Array of times when water-quality samples were collected, in minutes since start of storm. (R)
UD -- Array containing unit discharge data. (R)
WFALL -- Array of concentrations of constituents in precipitation. (R)
YR -- Year (last two digits). (I)

ATTACHMENT F

PROGRAM LISTING

```
      ****************************************************************

C     ************************************************************
C     *                                                          *
C     *      R347--DISTRIBUTED ROUTING RUNOFF QUALITY MODEL      *
C     *                                                          *
C     ************************************************************
      REAL ISEG
      COMMON /C1/ NSEG,ISEG(99),JUP(99,3),JLAT(99,4),ITYPE(99),EFF(51)
      COMMON /C2/ IPR(99),FLGTH(99),PARAM(99,4),LAND(99),NCAT,NRES
      COMMON /C3/ SSEFF(4),SSMIN(4),SSAREA(51),AKA(4),AKB(4)
      COMMON /F1/ ICT,Q(1442),R(1442),QMX,I1,DELTAT,DELPLG,NRV(10)
      COMMON /F2/ IYR,IMO,IDY,NDYS,ICK(60),JCONC,JLOAD,JRECQW
      COMMON /F3/ IFILE,IFILED,IFILEQ,JRECDS,IRECD,IRECQ,NSTRMS,IMODE
      COMMON /F5/ HEAD1,HEAD2,HEAD3
      COMMON /F4/ FUP(1442),FLAT(1442),QWUP(4,1442),QWLAT(4,1442),FLW(14
     142)
      COMMON /I1/ NLU,LUSE(3,4),BK(5,4,4),ISSFRQ,NPAGE
      COMMON /QWD/ CO(60,4,24),TIME(60,4,24),NWQM(60,4),RVLC(60,4)
      COMMON /ST1/ FPK(60),FVOL(60),FLD(60,4),CMX(60,4),WFALL(4,60),NWF
      COMMON /ST2/ NOFE,NF(60),NQU(4),TESTNO,KNN,CF1(4),IND(4)
      COMMON /ST3/ IPL(180),K1(60),K2(60),KOUT(180),IPA(4),IPB(4),NOPT
      COMMON /TIME/ NDELS,NOUD(180),INDP(20),NDATE(60,3)
      COMMON /UNIT/ NUDD,NWQ,UD(2881),DP(7310),RODYS,DA,ISS,PTIME,JPR
      COMMON /WQ1/ AT,ATD(4,51),XO(4,51),CON(4,1442),NSSDAY(3,40),ISWEEP
      COMMON /WQ2/ EAT(60,4),XOS(60,4),SFLD(60,4),BFL(60),RVB(60,4)
      COMMON /Z1/ IUD,ILOAD,I2CFSP,QWINT,AIMP,DAE,DT,DTS,NDTS
      INTEGER RODYS,TESTNO(180),HEAD1(120),HEAD2(60,2),HEAD3(60)
      WRITE (6,14)
      NSEG=0
C           SET SEQUENTIAL FILE NUMBER
      IFILED=26
C           SET ARRAY LIMITS
      NDYS=7310
      NDTS=1442
      IUD=2881
C           CALL PROGRAM SUBROUTINES
      CALL INPUT1
      IF (IFILED.GT.0) REWIND IFILED
      IF (JRECDS.GT.0) CALL FILES
      IF (JRECDS.GT.0) CALL HEADR
      CALL INPUT2
      IF (JRECQW.GT.0) CALL FILE27
      CALL CTCHMT
      DELPLG=PTIME/60.
      DELTAT=DT/60.
      IF (NRES.GT.0) CALL RESVR(2,1)
      CALL INPUT3
      CALL SIMQ
C           PLOT MEAS. AND SIM. LOADS
      DO 12 J=1,NWQ
      YMAX=0.0
```

****

```
      N=0
      DO 11 I=1,NOFE
      IF (SFLD(I,J).LE.0.0.OR.FLD(I,J).LE.0.0) GO TO 11
      N=N+1
      Q(N)=FLD(I,J)
      R(N)=SFLD(I,J)
      IF (Q(N).GT.YMAX) YMAX=Q(N)
      IF (R(N).GT.YMAX) YMAX=R(N)
   11 CONTINUE
      IF (N.EQ.0) GO TO 12
      CALL PLT(Q,R,N,1,YMAX,I1,J,4)
      CALL PLT(Q,R,N,3,YMAX,I1,J,4)
      WRITE (6,13)
   12 CONTINUE
      STOP
C
   13 FORMAT (16X,11HMEAS. LOADS)
   14 FORMAT (1H1,37X,51(1H*)/38X,1H*,14X,22HU.S. GEOLOGICAL SURVEY,13X,
     11H*/38X,51H*DISTRIBUTED ROUTING RAINFALL-RUNOFF-QUALITY MODEL*,/,3
     28X,1H*,16HVERSION  8/11/82,17X,1H*,/,38X,51(1H*))
      END
```

SUBROUTINE SIMQ

```
      SUBROUTINE SIMQ
      REAL ISEG
      INTEGER RODYS,TESTNO(180),W,FLAG
      COMMON /C1/ NSEG,ISEG(99),JUP(99,3),JLAT(99,4),ITYPE(99),EFF(51)
      COMMON /C2/ IPR(99),FLGTH(99),PARAM(99,4),LAND(99),NCAT,NRES
      COMMON /F1/ ICT,Q(1442),R(1442),QMX,I1,DELTAT,DELPLG,NRV(10)
      COMMON /F2/ IYR,IMO,IDY,NDYS,ICK(60),JCONC,JLOAD,JRECQW
      COMMON /F3/ IFILE,IFILED,IFILEQ,JRECDS,IRECD,IRECQ,NSTRMS,IMODE
      COMMON /F4/ FUP(1442),FLAT(1442),QWUP(4,1442),QWLAT(4,1442),FLW(14
     142)
      COMMON /I1/ NLU,LUSE(3,4),BK(5,4,4),ISSFRQ,NPAGE
      COMMON /QWD/ CO(60,4,24),TIME(60,4,24),NWQM(60,4),RVLC(60,4)
      COMMON /ST1/ FPK(60),FVOL(60),FLD(60,4),CMX(60,4),WFALL(4,60),NWF
      COMMON /ST2/ NOFE,NF(60),NQU(4),TESTNO,KNN,CF1(4),IND(4)
      COMMON /ST3/ IPL(180),K1(60),K2(60),KOUT(180),IPA(4),IPB(4),NOPT
      COMMON /TIME/ NDELS,NOUD(180),INDP(20),NDATE(60,3)
      COMMON /UNIT/ NUDD,NWQ,UD(2881),DP(7310),RODYS,DA,ISS,PTIME,JPR
      COMMON /WQ1/ AT,ATO(4,51),XO(4,51),CON(4,1442),NSSDAY(3,40),ISWEEP
      COMMON /WQ2/ EAT(60,4),XOS(60,4),SFLD(60,4),BFL(60),RVB(60,4)
      COMMON /Z1/ IUD,ILOAD,I2CFSP,QWINT,AIMP,DAE,DT,DTS,NDTS
      DATA FLAG/1/
C             INITIALIZE VARIABLES
      NSTRMS=0
      ISWEEP=0
      NPAGE=1
      NSS=1
      IF (JRECDS.GT.0) IDEL=(PTIME+0.001)/DT
      IF (IMODE.EQ.1) IDEL=1
      N1=1
      N2=NWQ
      DO 1 J=1,NWQ
      DO 1 NC=1,NCAT
      ATO(J,NC)=0.0
      XO(J,NC)=0.0
    1 CONTINUE
      DO 2 I=1,NOFE
      DO 2 J=1,NWQ
      SFLD(I,J)=0.0
    2 CONTINUE
      AT=0.5
      I1=1
      KP=1
      NSD=0
      NFD=0
      NFD1=0
      W=0
C             BEGIN SIMULATION
      DO 21 IW=1,RODYS
      W=W+1
      IF (W.GT.RODYS) GO TO 21
```

```
C             FOR GAP IN RECORD, INITIALIZE ACCUMULATION  TO ZERO
      IF (W.NE.INDP(KP)) GO TO 5
      JW=W
      LJ=KP+1
      W=INDP(LJ)+1
      LV=W-INDP(KP)
      KP=KP+2
      DO 3 I=1,NWQ
      DO 3 NC=1,NCAT
      XO(I,NC)=0.0
      ATO(I,NC)=0.0
    3 CONTINUE
      DO 4 I=1,LV
      CALL DATE
      IF (ISS.LT.2) GO TO 4
      IF (NSS.GT.ISS) GO TO 4
      IF (JW.EQ.NSSDAY(1,NSS)) NSS=NSS+1
      JW=JW+1
    4 CONTINUE
    5 IF (W.GT.1) CALL DATE
      IF (DP(W).LT.0.0) GO TO 17
C             IF FLAG=0, DO STORM COMPUTATIONS
C             IF FLAG=1, DO DAILY ACCOUNTING
      IF (FLAG.NE.0) GO TO 16
      NFD1=0
      NFD=NFD+1
      AIK=K1(I1)
      AT=AT-0.5+AIK/NDELS
C             START OF STORM LOOP FOR SEQUENCE OF DAYS
    6 IF (I1.GT.NOFE) GO TO 14
C             DETERMINE OUTPUT MODE
      JOUTPT=2
      IF (KOUT(I1).EQ.0) JOUTPT=1
      IF (JOUTPT.EQ.1.AND.IPL(I1).EQ.0) JOUTPT=0
      IF (JOUTPT.EQ.1.AND.IPL(I1).EQ.3) JOUTPT=0
      LJ=K1(I1)
      LK=K2(I1)
      IF (JOUTPT.EQ.2) WRITE (6,22)
C             COMPUTE LAND-SURFACE LOADS AT START OF STORM
      DO 7 J=N1,N2
      DO 7 NC=1,NCAT
      LU=LAND(NC)
      CAT=AT+ATO(J,NC)
      XO(J,NC)=BK(1,J,LU)*(1.0-EXP(-BK(2,J,LU)*CAT))
      EAT(I1,J)=CAT
      XOS(I1,J)=XO(J,1)
    7 CONTINUE
      NWET=NDATE(I1,1)
      IF (NWF.EQ.2) NWET=I1
C          ** COMPUTE CONCENTRATIONS FOR STORM I1 **
```

```
      IF (JRECDS.GT.0) GO TO 9
C           ** WASHOFF BASED ON UD'S
      LV=LK-LJ+1
      JJJ=0
      DO 8 I=LJ,LK
      JJJ=JJJ+1
    8 FLW(JJJ)=UD(I)
      CALL WASH(LV,N1,N2,1,1,QWINT,DAE,0)
      CALL CONC(LV,N1,N2,NWET)
      GO TO 11
C           ** WASHOFF BASED ON DISCHARGE FROM
C           DISTRIBUTED ROUTING RAINFALL-RUNOFF MODEL
    9 CALL TRNSPT(N1,N2,NWET)
C              OUTPUT HYDROGRAPH FROM RAINFALL-RUNOFF MODEL
C              IS TO BE USED IN LOAD COMPUTATIONS
      IF (IDEL.EQ.1) GO TO 11
      JJ=0
      JJJ=0
      DO 10 I=LJ,LK
      JJJ=JJJ+1
      JJ=JJ+IDEL
   10 FLW(JJJ)=FLW(JJ)
   11 CONTINUE
      AT=0.0
      DO 13 J=N1,N2
      IF (J.GT.N1.AND.KOUT(I1).NE.0) WRITE (5,22)
      DO 12 NC=1,NCAT
      LU=LAND(NC)
      ATO(J,NC)=(-1.0/BK(2,J,LU))*ALOG(1.0-XO(J,NC)/BK(1,J,LU))
   12 CONTINUE
C              OUTPUT DETAILED SIMULATED DATA
      CALL OUTPT(ICNT,IDEL,J,JOUTPT)
C              COMPUTE LOADS AND PERFORM DESIRED PLOTTING
      CALL QWLOAD(J)
   13 CONTINUE
      I1=I1+1
      NFD1=NFD1+1
C              IF HAVE ANALYZED ALL EVENTS OF SET OF EVENTS, GO TO 1680
      IF (NF(NFD).EQ.NFD1) GO TO 14
      AIK=LK+1
      BIK=K1(I1)
      IF (AIK.EQ.BIK) KINIT=1
      AT=AT+(BIK-AIK)/NDELS
      GO TO 6
   14 NFD=NFD+NFD1-1
      IF (MOD(LK,NDELS).EQ.0) GO TO 15
      ALK=LK
      LKT=LK/NDELS
      AKT=(LKT+1.0)*NDELS
      AT=AT+0.5+(AKT-ALK)/NDELS
```

SUBROUTINE SIMQ

```
   15 IF (W.GT.RODYS) GO TO 21
      FLAG=1
      NFD1=0
C               ** DAILY ACCOUNTING **
   16 CONTINUE
      DPP=DP(W)
      CALL DACC(DPP,N1,N2,W,NSS)
C               FINISHED WITH DAY
      GO TO 21
C               BEGIN UNIT-TIME SIMULATION
   17 FLAG=0
C               CHECK FOR STREET SWEEPING
      NPAGE=1
      IF (ISS.NE.1) GO TO 18
      NSS=NSS+1
      IF (NSS.GE.ISSFRQ) NSS=0
      GO TO 19
   18 IF (ISS.LT.2) GO TO 19
      IF (NSS.GT.ISS) GO TO 19
      IF (W.NE.NSSDAY(1,NSS)) GO TO 19
      NSS=NSS+1
   19 CONTINUE
      NFD1=NFD1+1
      IF (IFILED.EQ.0) GO TO 21
C            IF 1ST DAY OF SEQUENCE OF STORM DAYS THEN, READ DISCHARGE
C            DATA FROM IFILED
      IF (NFD1.GT.1) GO TO 21
      NSD=NSD+1
      READ (IFILED) K4D,(UD(I),I=1,K4D)
      K4DP=K4D+1
      DO 20 I=K4DP,IUD
   20 UD(I)=0.0
      CALL STORM(I1)
   21 CONTINUE
C            END OF SIMULATION PERIOD
      IF (IFILED.GT.0) REWIND IFILED
C            SUMMARIZE ALL STORM DATA
      IF (IFILED.GT.0) CALL RITE1
      CALL RITE2(N1,N2)
      RETURN
C
   22 FORMAT (1H1)
      END
```

SUBROUTINE DACC(DPP,N1,N2,W,NSS)

```
      SUBROUTINE DACC(DPP,N1,N2,W,NSS)
C              THIS SUBROUTINE IS FOR DAILY ACCOUNTING
      DIMENSION DLYWSH(4)
      INTEGER RODYS,W
      REAL I2CFSP,ISEG
      COMMON /C1/ NSEG,ISEG(99),JUP(99,3),JLAT(99,4),ITYPE(99),EFF(51)
      COMMON /C2/ IPR(99),FLGTH(99),PARAM(99,4),LAND(99),NCAT,NRES
      COMMON /C3/ SSEFF(4),SSMIN(4),SSAREA(51),AKA(4),AKB(4)
      COMMON /F2/ IYR,IMO,IDY,NDYS,ICK(60),JCONC,JLOAD,JRECQW
      COMMON /I1/ NLU,LUSE(3,4),BK(5,4,4),ISSFRQ,NPAGE
      COMMON /ST3/ IPL(180),K1(60),K2(60),KOUT(180),IPA(4),IPB(4),NOPT
      COMMON /UNIT/ NUDD,NWQ,UD(2881),DP(7310),RODYS,DA,ISS,PTIME,JPR
      COMMON /WQ1/ AT,ATO(4,51),XO(4,51),CON(4,1442),NSSDAY(3,40),ISWEEP
      COMMON /Z1/ IUD,ILOAD,I2CFSP,QWINT,AIMP,DAE,DT,DTS,NDTS
      DPP=DPP-AIMP
C             CHECK FOR STREET SWEEPING
      IF (ISS.LT.1) GO TO 2
      IF (ISS.NE.1) GO TO 1
      NSS=NSS+1
      IF (NSS.LT.ISSFRQ) GO TO 2
      ISWEEP=1
      NSS=0
      GO TO 2
    1 IF (NSS.GT.ISS) GO TO 2
      IF (W.NE.NSSDAY(1,NSS)) GO TO 2
      NSS=NSS+1
      ISWEEP=1
    2 CONTINUE
      IF (DPP.LE.0.0) GO TO 6
C              ADJ. LOADS FOR DAILY PPT.
      ISWEEP=0
      DO 3 J=N1,N2
      DLYWSH(J)=0.0
      DO 3 NC=1,NCAT
      LU=LAND(NC)
      AKD=BK(5,J,LU)
      CAT=AT+ATO(J,NC)
      XOTMP=BK(1,J,LU)*(1.0-EXP(-BK(2,J,LU)*CAT))
      DELTAP=XOTMP*(1.0-EXP(-AKD*DPP))
      XO(J,NC)=XOTMP-DELTAP
      DLYWSH(J)=DLYWSH(J)+DELTAP*EFF(NC)
    3 CONTINUE
      IF (ILOAD.NE.1) GO TO 8
C              OUTPUT DAILY LOADS, IF DESIRED
      IF (NPAGE.EQ.0) GO TO 4
      WRITE (6,12)
      NPAGE=0
    4 CONTINUE
      DO 5 NW=N1,N2
      IF (NW.EQ.N1) WRITE (6,11) IMO,IDY,IYR,IPA(NW),IPB(NW),DLYWSH(NW)
```

SUBROUTINE DACC(DPP,N1,N2,W,NSS)

```
      IF (NW.GT.N1) WRITE (6,13) IPA(NW),IPB(NW),DLYWSH(NW)
    5 CONTINUE
      GO TO 8
    6 AT=AT+1.0
      IF (ISWEEP.EQ.0) RETURN
C             ADJUST LAND-SURFACE LOADS FOR STREET SWEEPING
      AT=AT-1.0
      ISWEEP=0
      IF (ILOAD.EQ.1) WRITE (6,10) IMO,IDY,IYR
      DO 7 J=N1,N2
      DO 7 NC=1,NCAT
      LU=LAND(NC)
      CAT=AT+ATO(J,NC)
      XO(J,NC)=BK(1,J,LU)*(1.0-EXP(-BK(2,J,LU)*CAT))
      IF (XO(J,NC).LE.SSMIN(J)) GO TO 7
      XOTMP=XO(J,NC)
      XOTMP=XOTMP-(XOTMP-SSMIN(J))*SSEFF(J)
      XO(J,NC)=(1.-SSAREA(NC))*XO(J,NC)+XOTMP*SSAREA(NC)
    7 CONTINUE
    8 DO 9 J=N1,N2
      DO 9 NC=1,NCAT
      LU=LAND(NC)
      ATO(J,NC)=(-1.0/BK(2,J,LU))*ALOG(1.0-XO(J,NC)/BK(1,J,LU))
    9 CONTINUE
      AT=1.0
      RETURN
C
   10 FORMAT (1H0,2HON,I3,1H/,I2,1H/,I2,19H STREETS WERE SWEPT)
   11 FORMAT (1H0,2HON,I3,1H/,I2,1H/,I2,41H SIMULATED IMPERVIOUS AREA WA
     1SHOFF OF :  ,2A3,4H WAS,F8.2,7H POUNDS)
   12 FORMAT (1H1)
   13 FORMAT (1H ,52X,2A3,4H WAS,F8.2,7H POUNDS)
      END
```

SUBROUTINE DATE

```
      SUBROUTINE DATE
      DIMENSION IDAYS(12)
      COMMON /F2/ IYR,IMO,IDY,NDYS,ICK(60),JCONC,JLOAD,JRECQW
      DATA IDAYS/31,28,31,30,31,30,31,31,30,31,30,31/
      IDY=IDY+1
      IF (IDAYS(IMO).GE.IDY) RETURN
      IF (IMO.NE.2) GO TO 1
      IF (MOD(IYR,4).NE.0) GO TO 1
      IF (IDY.LE.29) RETURN
    1 IMO=IMO+1
      IDY=1
      IF (IMO.LE.12) RETURN
      IMO=1
      IYR=IYR+1
      RETURN
      END
```

SUBROUTINE WASH(LV,N1,N2,NC,LU,DTIME,DAET,LBS)

```
      SUBROUTINE WASH(LV,N1,N2,NC,LU,DTIME,DAET,LBS)
C               IMPERVIOUS AREA WASHOFF ROUTINE
      INTEGER TESTNO(180)
      COMMON /F4/ FUP(1442),FLAT(1442),QWUP(4,1442),QWLAT(4,1442),FLW(14
     1442)
      COMMON /I1/ NLU,LUSE(3,4),BK(5,4,4),ISSFRQ,NPAGE
      COMMON /ST1/ FPK(60),FVOL(60),FLD(60,4),CMX(60,4),WFALL(4,60),NWF
      COMMON /ST2/ NOFE,NF(60),NQU(4),TESTNO,KNN,CF1(4),IND(4)
      COMMON /WQ1/ AT,ATO(4,51),XO(4,51),CON(4,1442),NSSDAY(3,40),ISWEEP
C               CONVF CONVERTS CUBIC FT TO INCHES OVER CONTRIB. AREA
C               12*640/5280**2=2.754821E-4
      CONVF=2.754821E-4/DAET
      CONVF2=3600./DTIME
      DO 4 JJ=1,LV
      IF (JJ.EQ.1) UDI=FLW(1)
      IF (JJ.GT.1) UDI=FLW(JJ-1)
      RVCF=0.5*(FLW(JJ)+UDI)*DTIME
      RVIN=RVCF*CONVF
      RRATE=RVIN*CONVF2
      DO 3 J=N1,N2
      IF (RVCF.GT.0.00001) GO TO 1
      CON(J,JJ)=0.0
      GO TO 3
    1 CONTINUE
      IF (BK(3,J,LU).GT.0.0) WSHOFF=XO(J,NC)*(1.0-EXP(-BK(3,J,LU)*RVIN))
      IF (BK(3,J,LU).LT.0.0) WSHOFF=XO(J,NC)*(1.0-EXP(BK(3,J,LU)*RVIN*RR
     1ATE))
      IF (BK(4,J,LU).LE.0.001) GO TO 2
      AVAIL=BK(4,J,LU)*RRATE
      IF (AVAIL.LT.1.0) WSHOFF=WSHOFF*AVAIL
    2 XO(J,NC)=XO(J,NC)-WSHOFF
      CON(J,JJ)=WSHOFF*DAET
      IF (LBS.GT.0) GO TO 3
C               LBS/CUBIC FT * 16018.9 = MG/L
      CON(J,JJ)=CON(J,JJ)/RVCF*16018.9
      IF (CF1(J).LT.1.0E-6) CON(J,JJ)=CON(J,JJ)*1000.
    3 CONTINUE
    4 CONTINUE
      RETURN
      END
```

SUBROUTINE CONC(LV,N1,N2,NWET)

```
      SUBROUTINE CONC(LV,N1,N2,NWET)
C             THIS SUBROUTINE CONVERTS AVERAGE CONCS.
C             OVER TIME INTERVALS TO POINT CONCS.
      COMMON /F4/ FUP(1442),FLAT(1442),QWUP(4,1442),QWLAT(4,1442),FLW(14
     1442)
      COMMON /ST1/ FPK(60),FVOL(60),FLD(60,4),CMX(60,4),WFALL(4,60),NWF
      COMMON /WQ1/ AT,ATO(4,51),XO(4,51),CON(4,1442),NSSDAY(3,40),ISWEEP
      DO 3 J=N1,N2
      WET=WFALL(J,NWET)
      DO 2 JJ=1,LV
      IF (JJ.EQ.LV) GO TO 2
      IF (FLW(JJ).GT.0.0) GO TO 1
      CON(J,JJ)=0.0
      GO TO 2
    1 CONCEN=0.5*(CON(J,JJ)+CON(J,JJ+1))
C             ADD WETFALL CONTRIBUTION
      CON(J,JJ)=CONCEN+WET
    2 CONTINUE
      CON(J,LV)=0.0
      IF (FLW(LV).LE.0.0) GO TO 3
C            SET LAST POINT CONC. = LAST INTERVAL CONC. + WETFALL
      CON(J,LV)=CON(J,LV)+WET
    3 CONTINUE
      RETURN
      END
```

SUBROUTINE PERV(I,LK,N1,N2,DTIME,NWET)

```
      SUBROUTINE PERV(I,LK,N1,N2,DTIME,NWET)
C             PERVIOUS AREA WASHOFF ROUTINE
      INTEGER TESTNO(180)
      COMMON /C2/ IPR(99),FLGTH(99),PARAM(99,4),LAND(99),NCAT,NRES
      COMMON /C3/ SSEFF(4),SSMIN(4),SSAREA(51),AKA(4),AKB(4)
      COMMON /F1/ ICT,Q(1442),R(1442),QMX,I1,DELTAT,DELPLG,NRV(10)
      COMMON /F4/ FUP(1442),FLAT(1442),QWUP(4,1442),QWLAT(4,1442),FLW(14
     142)
      COMMON /ST1/ FPK(60),FVOL(60),FLD(60,4),CMX(60,4),WFALL(4,60),NWF
      COMMON /ST2/ NOFE,NF(60),NQU(4),TESTNO,KNN,CF1(4),IND(4)
      COMMON /WQ1/ AT,ATO(4,51),XO(4,51),CON(4,1442),NSSDAY(3,40),ISWEEP
      CONV=DTIME/43560./2.0
      Q(1)=FLW(1)*CONV
      SUMVOL=Q(1)*Q(1)
      TOTVOL=Q(1)
      QPK=FLW(1)
      DO 1 IV=2,LK
      Q(IV)=(FLW(IV)+FLW(IV-1))*CONV
      IF (Q(IV).LE.0.0) GO TO 1
      SUMVOL=SUMVOL+Q(IV)*Q(IV)
      TOTVOL=TOTVOL+Q(IV)
      IF (FLW(IV).GT.QPK) QPK=FLW(IV)
    1 CONTINUE
      TMASS=PARAM(I,3)*(TOTVOL*QPK)**PARAM(I,4)
      TERM=TMASS*735.48/SUMVOL
      DO 3 IV=1,LK
      SED=Q(IV)*TERM
      DO 3 JJJ=N1,N2
      IF (SED.GT.0.0) GO TO 2
      CON(JJJ,IV)=0.0
      GO TO 3
    2 CONC=AKA(JJJ)+AKB(JJJ)*SED
      IF (CF1(JJJ).LT.1.0E-6) CONC=CONC*1000.
      CON(JJJ,IV)=CONC+WFALL(JJJ,NWET)
    3 CONTINUE
      RETURN
      END
```

```
      SUBROUTINE TRNSPT(N1,N2,NWET)
C             SET UP FLOW AND CONC. ARRAYS FOR CONSTITUENT ROUTING
      INTEGER RODYS,HEAD1(120),HEAD2(60,2),HEAD3(60)
      REAL ISEG,I2CFSP
      COMMON /C1/ NSEG,ISEG(99),JUP(99,3),JLAT(99,4),ITYPE(99),EFF(51)
      COMMON /C2/ IPR(99),FLGTH(99),PARAM(99,4),LAND(99),NCAT,NRES
      COMMON /F1/ ICT,Q(1442),R(1442),QMX,I1,DELTAT,DELPLG,NRV(10)
      COMMON /F3/ IFILE,IFILED,IFILEQ,JRECDS,IRECD,IRECQ,NSTRMS,IMODE
      COMMON /F4/ FUP(1442),FLAT(1442),QWUP(4,1442),QWLAT(4,1442),FLW(14
     142)
      COMMON /F5/ HEAD1,HEAD2,HEAD3
      COMMON /ST3/ IPL(180),K1(60),K2(60),KOUT(180),IPA(4),IPB(4),NOPT
      COMMON /TIME/ NDELS,NOUD(180),INDP(20),NDATE(60,3)
      COMMON /UNIT/ NUDD,NWQ,UD(2881),DP(7310),RODYS,DA,ISS,PTIME,JPR
      COMMON /WQ1/ AT,ATO(4,51),XO(4,51),CON(4,1442),NSSDAY(3,40),ISWEEP
      COMMON /Z1/ IUD,ILOAD,I2CFSP,QWINT,AIMP,DAE,DT,DTS,NDTS
      INTVAL=(PTIME+0.001)/DT
      LJ=1
      LV=(K2(I1)-K1(I1)+1)
      LK=LV*INTVAL
      ICT=LK
      NSTRMS=NSTRMS+1
      IF (LK.NE.HEAD2(NSTRMS,2)) GO TO 34
      NSTRCD=HEAD2(NSTRMS,1)
      NRPSEG=LK/120+1-(1-MIN0(1,MOD(LK,120)))
      NC=0
      IRS=0
      DO 33 I=1,NSEG
      IF (ITYPE(I).EQ.2) GO TO 33
C             INITIALIZE ARRAYS TO ZERO
      JPERV=0
      DO 3 L=1,LK
      FLAT(L)=0.0
      FLW(L)=0.0
      IF (NC.GT.0.AND.IMODE.EQ.2) GO TO 1
      FUP(L)=0.0
    1 DO 2 J=N1,N2
      QWLAT(J,L)=0.0
      CON(J,L)=0.0
      IF (NC.GT.0.AND.IMODE.EQ.2) GO TO 2
      QWUP(J,L)=0.0
    2 CONTINUE
    3 CONTINUE
      IF (ITYPE(I).GT.2) GO TO 15
      NC=NC+1
      LU=LAND(NC)
      DAET=EFF(NC)
C             COMPUTE IMPERVIOUS AREA LATERAL INFLOW
      DO 6 J=1,4
      IF (JLAT(I,J)) 6,6,4
```

```
    4 JJ=JLAT(I,J)
      IF (PARAM(JJ,2).LE.0.001) JPERV=1
      IF (PARAM(JJ,2).LE.0.001) GO TO 6
      IRECD=NSTRCD+NRPSEG*(JJ-1)
      READ (IFILE'IRECD) (Q(IV),IV=1,LK)
C           CONVERT CFS/FT TO CFS
      DO 5 IV=1,LK
    5 FLW(IV)=FLW(IV)+Q(IV)*FLGTH(I)
    6 CONTINUE
      IF (DAET.LE.0.0) GO TO 7
C          COMPUTE AVERAGE INTERVAL CONCS. OF CONSTITUENTS
C          IN IMPERVIOUS AREA LATERAL INFLOW
      CALL WASH(LK,N1,N2,NC,LU,DTS,DAET,0)
C             CONVERT TO POINT CONCS.
      CALL CONC(LK,N1,N2,NWET)
    7 DO 8 IV=1,LK
      FLAT(IV)=FLW(IV)
      DO 8 JJJ=N1,N2
      QWLAT(JJJ,IV)=CON(JJJ,IV)
    8 CONTINUE
C             COMPUTE PERVIOUS LATERAL INFLOW
      IF (JPERV.EQ.0) GO TO 15
      DO 9 L=1,LK
    9 FLW(L)=0.0
      DO 12 J=1,4
      IF (JLAT(I,J)) 12,12,10
   10 JJ=JLAT(I,J)
      IF (PARAM(JJ,2).GT.0.001) GO TO 12
      IRECD=NSTRCD+NRPSEG*(JJ-1)
      READ (IFILE'IRECD) (Q(IV),IV=1,LK)
      DO 11 IV=1,LK
   11 FLW(IV)=FLW(IV)+Q(IV)*FLGTH(I)
   12 CONTINUE
C             COMPUTE CONCS. IN PERVIOUS AREA LATERAL INFLOW
      CALL PERV(I,LK,N1,N2,DTS,NWET)
C             ADD PERV. + IMP. FLOW AND CONCS.
      DO 14 IV=1,LK
      QSUM=FLAT(IV)+FLW(IV)
      DO 13 JJJ=N1,N2
   13 QWLAT(JJJ,IV)=(QWLAT(JJJ,IV)*FLAT(IV)+CON(JJJ,IV)*FLW(IV))/QSUM
      FLAT(IV)=QSUM
   14 CONTINUE
   15 CONTINUE
      IF (IMODE.NE.2) GO TO 16
C             DISTRIBUTED (NO ROUTING RUN)
      IF (ITYPE(I).GT.1) GO TO 33
      CALL NOROUT(I,LK,N1,N2)
      GO TO 33
C             COUNT NUMBER OF UPSTREAM SEGMENTS
   16 NUP=0
```

```
      DO 17 J=1,3
   17 IF (JUP(I,J).GT.0) NUP=NUP+1
      IF (NUP.EQ.0) GO TO 26
C              COMPUTE UPSTREAM FLOW AND CONCENTRATIONS
      DO 24 J=1,3
      IF (JUP(I,J)) 24,24,18
   18 JJ=JUP(I,J)
      IRECD=NSTRCD+NRPSEG*(JJ-1)
      READ (IFILE'IRECD) (FLW(IV),IV=1,LK)
      DO 19 IV=1,LK
   19 FUP(IV)=FUP(IV)+FLW(IV)
      DO 23 JJJ=N1,N2
      IRECQ=1+(JJJ-1)*NRPSEG*NSEG+(JJ-1)*NRPSEG
      READ (IFILEQ'IRECQ) (Q(IV),IV=1,LK)
      IF (NUP.LT.2) GO TO 21
      DO 20 L=1,LK
   20 QWUP(JJJ,L)=QWUP(JJJ,L)+Q(L)*FLW(L)
      GO TO 23
   21 CONTINUE
      DO 22 L=1,LK
   22 QWUP(JJJ,L)=Q(L)
   23 CONTINUE
   24 CONTINUE
      IF (NUP.LT.2) GO TO 26
      DO 25 IV=1,LK
      DO 25 JJJ=N1,N2
      IF (FUP(IV).LE.0.0) GO TO 25
      QWUP(JJJ,IV)=QWUP(JJJ,IV)/FUP(IV)
   25 CONTINUE
   26 CONTINUE
      IF (ITYPE(I).NE.4) GO TO 28
C              MASS BALANCE AT NODAL SEGMENT
      DO 27 IV=1,LK
      DO 27 JJJ=N1,N2
      CON(JJJ,IV)=QWUP(JJJ,IV)+PARAM(I,JJJ)
   27 CONTINUE
      GO TO 31
C              READ DOWNSTREAM FLOW
   28 IRECD=NSTRCD+NRPSEG*(I-1)
      READ (IFILE'IRECD) (FLW(IV),IV=1,LK)
C              DO RESERVOIR ROUTING
      IF (ITYPE(I).NE.3) GO TO 30
      IRS=IRS+1
      IF (NUP.GT.0) GO TO 29
      WRITE (6,37)
      STOP
   29 CALL RESVR(4,IRS)
      GO TO 31
C              PERFORM LAGRANGIAN TRANSPORT
   30 CALL LNGRN(I,LK,N1,N2)
```

SUBROUTINE TRNSPT(N1,N2,NWET)

```
C              SAVE OUTFLOW CONC. ARRAY
   31 DO 32 JJJ=N1,N2
      IRECQ=1+(JJJ-1)*NRPSEG*NSEG+(I-1)*NRPSEG
      WRITE (IFILEQ'IRECQ) (CON(JJJ,IV),IV=1,LK)
C              LIST SEGMENT CONCENTRATIONS IF DESIRED
      IF (KOUT(I1).EQ.0) GO TO 32
      IF (IPR(I).EQ.0) GO TO 32
      JSEG=I
      CALL OUTPT(JSEG,INTVAL,JJJ,3)
      IF (JJJ.EQ.N2) WRITE (6,36)
   32 CONTINUE
   33 CONTINUE
      RETURN
   34 WRITE (6,35) I1
      STOP
C
   35 FORMAT (1H1,35HNUMBER OF TIME INCREMENTS FOR STORM,I3,42H DOES NOT
     1 MATCH BETWEEN DR3M AND DR3M-QUAL)
   36 FORMAT (1H1)
   37 FORMAT (1H1,39HRESERVOIR MUST HAVE AN UPSTREAM SEGMENT)
      END
```

SUBROUTINE NOROUT(I,LK,N1,N2)

```
      SUBROUTINE NOROUT(I,LK,N1,N2)
C             DETERMINE CONCS. FOR DIST. (NO ROUTING) RUN
C             FUP=SUM OF FLOWS
C             QWUP=SUM OF FLOW*CONC.
      COMMON /C1/ NSEG,ISEG(99),JUP(99,3),JLAT(99,4),ITYPE(99),EFF(51)
      COMMON /F4/ FUP(1442),FLAT(1442),QWUP(4,1442),QWLAT(4,1442),FLW(14
     142)
      COMMON /WQ1/ AT,ATO(4,51),XO(4,51),CON(4,1442),NSSDAY(3,40),ISWEEP
      REAL ISEG
      DO 2 L=1,LK
      FUP(L)=FUP(L)+FLAT(L)
      DO 1 JJJ=N1,N2
      QWUP(JJJ,L)=QWUP(JJJ,L)+FLAT(L)*QWLAT(JJJ,L)
      IF (I.LT.NSEG) GO TO 1
      IF (FUP(L).LE.0.0) GO TO 1
      CON(JJJ,L)=QWUP(JJJ,L)/FUP(L)
    1 CONTINUE
      FLW(L)=FUP(L)
    2 CONTINUE
      RETURN
      END
```

```
      SUBROUTINE LNGRN(I,LK,N1,N2)
C
C     SUBROUTINE LNGRN IS THE DRIVE SUBROUTINE FOR THE LAGRANGIAN
C     CHANNEL TRANSPORT. A CALL TO LNGRN ROUTES UP TO FOUR CONSTITUENTS
C     THROUGH A SEGMENT.
C
      REAL ISEG,M,CLAT(4),CA(4),CB(4),CC(4),CD(4)
      COMMON /C1/ NSEG,ISEG(99),JUP(99,3),JLAT(99,4),ITYPE(99),EFF(51)
      COMMON /C2/ IPR(99),FLGTH(99),PARAM(99,4),LAND(99),NCAT,NRES
      COMMON /F4/ FUP(1442),FLAT(1442),QWUP(4,1442),QWLAT(4,1442),FLW(14
     142)
      COMMON /WQ1/ AT,ATO(4,51),XO(4,51),CON(4,1442),NSSDAY(3,40),ISWEEP
      COMMON /Z1/ IUD,ILOAD,I2CFSP,QWINT,AIMP,DAE,DT,DTS,NDTS
      COMMON /LGN1/ XL,DTS1,JSEG,ALPHA,M,QLAT,CLAT,CC,J,N1T,N2T
      DATA CA,CB/0.,0.,0.,0.,0.,0.,0.,0./
C
C     ..... SET UP COMMON VARIABLES IN LGN1 .....
      XL=FLGTH(I)
      DTS1=DTS
      ALPHA=PARAM(I,1)
      M=PARAM(I,2)
      JSEG=I
      N1T=N1
      N2T=N2
C
C     ..... DEFINE INITIAL CONDITIONS FOR SEGMENT .....
      CALL LAGRN1(0.0,0.0,CA,CB)
C
C     ..... START TIME LOOP .....
      DO 3 J=1,LK
      QLAT=FLAT(J)/XL
      DO 1 II=N1,N2
      CLAT(II)=QWLAT(II,J)
    1 CC(II)=QWUP(II,J)
C
C     ..... PERFORM TRANSPORT COMPUTATIONS FOR ONE TIME STEP .....
      CALL LAGRN(FUP(J),FLW(J),CD)
      DO 2 II=N1,N2
    2 CON(II,J)=CD(II)
    3 CONTINUE
C
      RETURN
      END
```

SUBROUTINE LAGRN1(QA,QB,CA,CB)

```
      SUBROUTINE LAGRN1(QA,QB,CA,CB)
C
C     SUBROUTINE LAGRN1 MUST BE CALLED ONCE PRECEDING TRANSPORT
C     COMPUTATIONS TO DEFINE INITIAL CONDITIONS FOR A SEGMENT
C     (THIS SUBROUTINE ACCEPTS A SEGMENT HAVING AN INITIAL FLOW EVEN
C     THOUGH WITH THE MODELS PRESENT STRUCTURE THIS COULD NOT OCCUR)
C
      DIMENSION PV(100), PC(4,100), CLAT(4), CA(4), CB(4), CC(4)
      REAL M
      COMMON /LGN/ PV,PC,NPRCLS
      COMMON /LGN1/ XL,DTS,ISEG,ALPHA,M,QLAT,CLAT,CC,J,N1,N2
      COMMON /LGN2/ Q1,Q2,Q3,Q4
C
C     ..... DEFINE INITIAL CONDITIONS FOR TRANSPORT .....
      NPRCLS=2
      IF (QA.EQ.0.) GO TO 1
      AA=(QA/ALPHA)**(1./M)
      GO TO 2
    1 AA=0.0
    2 IF (QB.EQ.0.0) GO TO 3
      AB=(QB/ALPHA)**(1./M)
      GO TO 4
    3 AB=0.0
    4 PV(1)=.25*(AA+AB)*XL
      PV(2)=PV(1)
      DO 5 I=1,N2
      PC(I,1)=CA(I)
      PC(I,2)=CB(I)
    5 CONTINUE
      Q1=QA
      Q2=QB
      RETURN
      END
```

```
      SUBROUTINE LAGRN(QC,QD,CD)
C
C     SUB LAGRN - PLUG FLOW LAGRANGIAN TRANSPORT WITH LATERAL INFLOW
C
C     ..... DEFINITION OF VARIABLES .....
C     PV(K)  - VOLUME OF PARCEL K (CUBIC FEET)
C     PC(I,K)- CONCENTRATION OF CONSTITUENT I IN PARCEL K (MG/L)
C     XL     - LENGTH OF SEGMENT (FEET)
C     DTS    - TIME STEP SIZE (SECONDS)
C     CA     - CONC AT U/S END OF SEGMENT AT OLD TIME STEP (MG/L)
C     CC     - CONC AT U/S END OF SEGMENT AT NEW TIME STEP (MG/L)
C     CD     - CONC AT D/S END OF SEGMENT AT NEW TIME STEP (MG/L)
C     Q1,Q2  - FLOW AT U/S AND D/S ENDS OF SEGMENT AT OLD TIME STEP(CFS)
C     Q3,Q4  - FLOW AT U/S AND D/S ENDS OF SEGMENT AT NEW TIME STEP(CFS)
C     U2,U4  - VEL'S AT D/S END OF SEGMENT AT OLD AND NEW TIME STEP(FPS)
C     NPRCLS - NUMBER OF PARCELS IN THE SEGMENT DURING A GIVEN TIME STEP
C     UAVG   - AVERAGE VELOCITY IN SEGMENT (USING ALL 4 CORNERS) (FPS)
C     DX     - DISTANCE IMAGINARY PARCEL BOUNDARY MOVES (FT)
C
      DIMENSION PV(100), PC(4,100), CLAT(4), CA(4), CC(4), CD(4), PCN(4)
      REAL M
      COMMON /LGN/ PV,PC,NPRCLS
      COMMON /LGN1/ XL,DTS,ISEG,ALPHA,M,QLAT,CLAT,CC,J,N1,N2
      COMMON /LGN2/ Q1,Q2,Q3,Q4
      Q3=QC
      Q4=QD
      IF (J.NE.1) GO TO 4
      IF (Q1.NE.0.) GO TO 2
      DO 1 I=1,N2
    1 CA(I)=CC(I)
      GO TO 4
    2 CONTINUE
      DO 3 I=1,N2
    3 CA(I)=PC(I,1)
    4 IF (Q1.EQ.0.) CALL ZEROS(CD,&6,&17,&23)
C
C     ..... ADD A NEW PARCEL TO THE SEGMENT AND ASSIGN IT THE VOLUME AND
C           CONCENTRATION OF FLOW ENTERING FROM U/S END .....
      PVN=.5*(Q1+Q3)*DTS
      DO 5 I=N1,N2
    5 PCN(I)=.5*(CA(I)+CC(I))
      VOLX=0.0
      GO TO 10
C     SPECIAL CASE OF ZERO U/S BOUNDARY CONDITION (Q1=Q3=0)
    6 IF (QLAT.EQ.0.) GO TO 17
      UAVG=.25*(UX(Q2)+UX(Q4))
      DX=UAVG*DTS
      IF (DX.GT.XL) GO TO 7
      PVN=.5*DX*QLAT*DTS
      GO TO 8
```

SUBROUTINE LAGRN(QC,QD,CD)

```
    7 PVN=.5*XL*QLAT*(XL/UAVG)+XL*QLAT*(DTS-(XL/UAVG))
    8 VOLX=PVN
      DO 9 I=N1,N2
    9 PCN(I)=CLAT(I)
   10 NPRCLS=NPRCLS+1
      NP=NPRCLS-1
      DO 12 L=1,NP
      K=(NPRCLS+1)-L
      PV(K)=PV(K-1)
      DO 11 I=N1,N2
   11 PC(I,K)=PC(I,K-1)
   12 CONTINUE
      PV(1)=PVN
      DO 13 I=N1,N2
   13 PC(I,1)=PCN(I)
C
C     ..... COMPUTE SUMMATION OF PARCEL VOLUMES .....
      SUMV=0.0
      L=1
      IF (VOLX.GT.0.0) L=2
      DO 14 K=L,NPRCLS
   14 SUMV=SUMV+PV(K)
C
C     ..... COMPUTE VOLUME OF LATERAL INFLOW .....
      VLAT=QLAT*XL*DTS-(VOLX)
C
C     ..... DISTRIBUTE LATERAL INFLOW AMONG PARCELS IN PROPORTION TO
C           PARCEL VOLUMES AND ADJUST CONCENTRATIONS .....
      L=1
      IF (VOLX.GT.0.0) L=2
      DO 16 K=L,NPRCLS
      ZVOL=VLAT*(PV(K)/SUMV)
      PVN=PV(K)+ZVOL
      DO 15 I=N1,N2
   15 PC(I,K)=(PC(I,K)*PV(K)+ZVOL*CLAT(I))/PVN
      PV(K)=PVN
   16 CONTINUE
C
C     ..... COMPUTE VOLUME LEAVING SEGMENT .....
   17 CONTINUE
      VOUT=.5*(Q2+Q4)*DTS
C
C     ..... DETERMINE WHICH PARCELS LEAVE SEGMENT .....
      DO 20 I=1,NPRCLS
      K=(NPRCLS+1)-I
      VOUT=VOUT-PV(K)
      IF (VOUT) 18,19,20
   18 PV(K)=ABS(VOUT)
      GO TO 21
   19 K=K-1
```

```
      GO TO 21
   20 CONTINUE
C
C     .....UPDATE NUMBER OF PARCELS AND ASSIGN CONC AT D/S END .....
   21 NPRCLS=K
      DO 22 I=N1,N2
   22 CD(I)=PC(I,NPRCLS)
C
C     ..... SET NEW TIME STEP TO OLD TIME STEP .....
   23 Q1=Q3
      Q2=Q4
      DO 24 I=N1,N2
   24 CA(I)=CC(I)
      RETURN
      END
```

SUBROUTINE ZEROS(CD,*,*,*)

```
      SUBROUTINE ZEROS(CD,*,*,*)
C
C     SUB ZEROS TAKES CARE OF SITUATIONS IN THE CHANNEL TRANSPORT
C     COMPUTATIONS WHERE THERE ARE ZERO FLOWS
C
      DIMENSION PV(100), PC(4,100), CLAT(4), CC(4), CD(4)
      REAL M
      COMMON /LGN/ PV,PC,NPRCLS
      COMMON /LGN1/ XL,DTS,ISEG,ALPHA,M,QLAT,CLAT,CC,J,N1,N2
      COMMON /LGN2/ Q1,Q2,Q3,Q4
C
      IF ((Q4.EQ.0.).AND.(Q3.EQ.0.).AND.(Q2.EQ.0.)) GO TO 1
      IF ((Q3.EQ.0.).AND.(Q2.EQ.0.)) GO TO 3
      IF (Q3.EQ.0.) GO TO 5
      IF (Q2.EQ.0.) GO TO 6
C
    1 CONTINUE
      DO 2 I=N1,N2
    2 CD(I)=0.0
      RETURN 3
    3 CONTINUE
      PV(1)=.5*QLAT*XL*DTS
      PV(2)=PV(1)
      DO 4 I=N1,N2
      PC(I,1)=CLAT(I)
      PC(I,2)=PC(I,1)
    4 CD(I)=CLAT(I)
      RETURN 2
    5 RETURN 1
    6 VLAT=QLAT*XL*DTS
      UAVG=.25*(UX(Q3)+UX(Q4))
      DX=UAVG*DTS
      IF (DX.GT.XL) GO TO 7
      ZVOL=.5*DX*QLAT*DTS
      GO TO 8
    7 ZVOL=.5*XL*QLAT*(XL/UAVG)+XL*QLAT*(DTS-(XL/UAVG))
    8 PVN=0.5*Q3*DTS
      PV(1)=PVN+ZVOL
      PV(2)=VLAT-ZVOL
      IF (PV(2).LE.0.0) PV(2)=1.E-10
      DO 9 I=N1,N2
      PC(I,1)=(CC(I)*PVN+ZVOL*CLAT(I))/PV(1)
    9 PC(I,2)=CLAT(I)
      RETURN 2
      END
```

FUNCTION UX(Q)

```
      FUNCTION UX(Q)
C
C     FUNCTION UX COMPUTES VEL FOR CHANNEL TRANSPORT GIVEN Q, ALPHA, & M
C
      REAL M,CLAT(4),CC(4)
      COMMON /LGN1/ XL,DTS,ISEG,ALPHA,M,QLAT,CLAT,CC,J,N1,N2
C
      IF (Q.EQ.0.) GO TO 1
      YM=1./M
      UX=(Q**(1.-YM))*(ALPHA**YM)
      GO TO 2
    1 UX=0.
    2 RETURN
      END
```

SUBROUTINE RESVR(ISTEP,IRS)

```
      SUBROUTINE RESVR(ISTEP,IRS)
      COMMON /R1/ AREA(5,10),DISCH(5,10),CAPAC(5,10),STAGE(5,10)
      COMMON /R2/ AVDPTH(5,10),PERCNT(5,4,10),SIZE(10),DCUBE(10)
      COMMON /R3/ FALL(8),SG,STOKES,VISCOS,VELOC(8),DIAMT(8),PERCT(8)
      COMMON /R4/ TMEIN(2),OUTFL(5,8),M,LR,MR,DETK1,DETK2
      COMMON /R5/ VTSUM(5,4,10),REMN(8),REM(8),FIX(5),RISER(5)
      COMMON /R6/ NLAYER(5),JFLOW(5),DEAD(5),NDX(5),NNS(5),CAPOOL(5)
      COMMON /R7/ TMASS(4),QWT1(4),QWT2(4),QWND1(4),QWND2(4),QWOUT(4)
      COMMON /C1/ NSEG,ISEG(99),JUP(99,3),JLAT(99,4),ITYPE(99),EFF(51)
      COMMON /C2/ IPR(99),FLGTH(99),PARAM(99,4),LAND(99),NCAT,NRES
      COMMON /F1/ ICT,STGAR(1442),STP(1442),QMX,I1,DELTAT,DELPLG,NRV(10)
      COMMON /F4/ FUP(1442),T1(1442),QWUP(4,1442),QWTOT(4,1442),FLW(1442
     1)
      COMMON /ST2/ NOFF,NF(60),NQU(4),TESTNO,KNN,CF1(4),IND(4)
      COMMON /ST3/ IPL(180),K1(60),K2(60),KOUT(180),IPA(4),IPB(4),NOPT
      COMMON /UNIT/ NUDD,NWQ,UD(2881),DP(7310),RODYS,DA,ISS,PTIME,JPR
      COMMON /WQ1/ AT,ATO(4,51),XO(4,51),CON(4,1442),NSSDAY(3,40),ISWEEP
      REAL ISEG
      INTEGER TESTNO(180)
      GO TO (1,3,22,22), ISTEP
    1 CONTINUE
C              START HERE IF SEPARATE PROGRAM
      NRES=1
      IPR(1)=1
      NRV(1)=1
      I1=1
      KOUT(I1)=1
      NWQ=1
      READ (5,76) ICT,DELTAT,DELPLG
      M=ICT
      READ (5,73) ISEG(1),(IPA(JQW),IPB(JQW),JQW=1,NWQ)
      IF (ICT.GT.1442) STOP
      DO 2 JQW=1,NWQ
    2 CF1(JQW)=10.
    3 CONTINUE
C              START HERE FOR DR3M-QUAL
      READ (5,76) NS,SG,VISCOS
      IF (SG.LE.1.0) SG=2.65
      IF (VISCOS.LE.0.005) VISCOS=0.0114
      STOKES=5.15E-5*(SG-1.)/VISCOS
      DETK1=DELTAT*.08264/2.0
      DETK2=2000.*SG
      READ (5,74) (SIZE(NL),NL=1,NS)
      SIZE(1)=0.0
      DO 4 I=2,NS
      IF (SIZE(I).LE.SIZE(I-1)) GO TO 72
    4 CONTINUE
C              LOOP THROUGH RESERVOIRS AND READ INPUT DATA
      DO 21 IRES=1,NRES
      READ (5,75) NLAY,JFL,N,DEAD(IRES),RISER(IRES)
```

```
      IF (JFL.EQ.1) NLAY=1
      NLAYER(IRES)=NLAY
      JFLOW(IRES)=JFL
      NDX(IRES)=N
      NNS(IRES)=NS
      FIX(IRES)=1.0
      ANLAY=NLAY
      NV=NRV(IRES)
      WRITE (6,90) ISEG(NV),NLAYER(IRES),JFLOW(IRES),DEAD(IRES),N,NS
      WRITE (6,91) SG,VISCOS
      IF (JFL.EQ.4) WRITE (6,92) RISER(IRES)
      DO 6 JQW=1,NWQ
      READ (5,74) (PERCNT(IRES,JQW,NL),NL=1,NS)
      IF (PERCNT(IRES,JQW,1).NE.0.0) PERCNT(IRES,JQW,1)=0.0
      IF (PERCNT(IRES,JQW,NS).NE.100.) PERCNT(IRES,JQW,NS)=100.
      DO 5 I=2,NS
      IF (PERCNT(IRES,JQW,I).GE.PERCNT(IRES,JQW,I-1)) GO TO 5
      WRITE (6,79)
      STOP
    5 CONTINUE
    6 CONTINUE
      READ (5,74) (STAGE(IRES,I),I=1,N)
      READ (5,74) (AREA(IRES,I),I=1,N)
      READ (5,74) (DISCH(IRES,I),I=1,N)
      DO 7 I=2,N
      IF (STAGE(IRES,I).LE.STAGE(IRES,I-1)) GO TO 72
      IF (AREA(IRES,I).LE.AREA(IRES,I-1)) GO TO 72
      IF (DISCH(IRES,I).LE.DISCH(IRES,I-1)) GO TO 72
    7 CONTINUE
C              COMPUTE RES. CAPACITY AT EACH PT. ON S-O CURVE
      AVDPTH(IRES,1)=0.0
      CAPAC(IRES,1)=0.0
      DO 8 J=2,N
      CAPAC(IRES,J)=(AREA(IRES,J)+AREA(IRES,J-1))*(STAGE(IRES,J)-STAGE(I
     1RES,J-1))/2.0+CAPAC(IRES,J-1)
    8 CONTINUE
      DO 9 J=1,N
      IF (DISCH(IRES,J).LE..005) CAPOOL(IRES)=CAPAC(IRES,J)
    9 CONTINUE
      WRITE (6,95) CAPOOL(IRES)
      IF (DEAD(IRES).LE.CAPOOL(IRES)) GO TO 10
      WRITE (6,77)
      STOP
C              COMPUTE AVERAGE DEPTHS FOR EACH STAGE
   10 DO 12 I=2,N
      SUM1=0.0
      SUM2=0.0
      DO 11 J=2,I
      DEPO=STAGE(IRES,I)-(STAGE(IRES,J)+STAGE(IRES,J-1))/2.0
      SUM1=DEPO*DEPO*(AREA(IRES,J)-AREA(IRES,J-1))+SUM1
```

```
      SUM2=DEPO*(AREA(IRES,J)-AREA(IRES,J-1))+SUM2
   11 CONTINUE
      AVDPTH(IRES,I)=SUM1/SUM2
   12 CONTINUE
      WRITE (6,89)
      WRITE (6,80)
      WRITE (6,81)
      DO 13 IL=1,N
      WRITE (6,82) STAGE(IRES,IL),AREA(IRES,IL),AVDPTH(IRES,IL),DISCH(IR
     1ES,IL),CAPAC(IRES,IL)
   13 CONTINUE
      WRITE (6,83)
      WRITE (6,84) (SIZE(I),I=1,NS)
      DO 14 JQW=1,NWQ
      WRITE (6,85) IPA(JQW),IPB(JQW),(PERCNT(IRES,JQW,I),I=1,NS)
   14 CONTINUE
C             COMPUTE INTEGRATED SETTLING VELOCITIES
      DO 15 JQW=1,NWQ
      DO 15 I=1,NS
      VTSUM(IRES,JQW,I)=0.0
      IF (I.EQ.1) GO TO 15
      SLOPE=(PERCNT(IRES,JQW,I)-PERCNT(IRES,JQW,I-1))/(SIZE(I)-SIZE(I-1)
     1)
      D2CUBE=SIZE(I)*SIZE(I)*SIZE(I)
      DCUBE(I-1)=SIZE(I-1)*SIZE(I-1)*SIZE(I-1)
      VTINC=(SLOPE*STOKES/3.)*(D2CUBE-DCUBE(I-1))
      VTSUM(IRES,JQW,I)=VTSUM(IRES,JQW,I-1)+VTINC
   15 CONTINUE
C             SET UP OUTFLOW DISTRIBUTION
      DO 16 NL=1,NLAY
   16 OUTFL(IRES,NL)=0.0
      GO TO (17,19,20,21), JFL
   17 DO 18 NL=1,NLAY
   18 OUTFL(IRES,NL)=100./ANLAY
      GO TO 21
   19 OUTFL(IRES,1)=100.
      GO TO 21
   20 OUTFL(IRES,NLAY)=100.
   21 CONTINUE
      IF (ISTEP.EQ.2) RETURN
C             READ IN ADDITIONAL DATA IF SEPARATE PROGRAM
      READ (5,74) (FUP(I),I=1,M)
      READ (5,74) (FLW(I),I=1,M)
      READ (5,74) (QWUP(1,I),I=1,M)
      RETURN
   22 CONTINUE
C             ** MAIN BODY OF SUBROUTINE **
      IRES=IRS
      M=ICT
      LR=(DELPLG+.01)/DELTAT
```

```
      MR=M/LR+.01
      N=NDX(IRES)
      NS=NNS(IRES)
      NLAY=NLAYER(IRES)
      ANLAY=NLAY
C              DETERMINE CHAR. OF INFLOW QW AND FLOW AT EACH TIME STEP
      DO 27 JQW=1,NWQ
      IF (CF1(JQW).GT.1.0E-6) GO TO 24
C              CORRECT FOR UG/L
      DO 23 I=1,M
      QWUP(JQW,I)=QWUP(JQW,I)/1000.
   23 CONTINUE
   24 VOLUME=FUP(1)*DETK1
      STP(1)=VOLUME+CAPOOL(IRES)
      TMASS(JQW)=.001359*QWUP(JQW,1)*VOLUME/2.0
      QWTOT(JQW,1)=QWUP(JQW,1)*VOLUME/DETK2
      DO 26 I=2,M
      VOLUME=(FUP(I-1)+FUP(I))*DETK1
      IF (JQW.GT.1) GO TO 25
      STP(I)=STP(I-1)+VOLUME
   25 AMULT=(QWUP(JQW,I)+QWUP(JQW,I-1))*VOLUME
      TMASS(JQW)=TMASS(JQW)+0.001359*AMULT/2.0
      QWTOT(JQW,I)=QWTOT(JQW,I-1)+AMULT/DETK2
   26 CONTINUE
   27 CONTINUE
      AVSTG1=0.0
C              COMPUTE CUMULATIVE STAGE AFTER EACH INCREMENT OF INFLOW
      DO 30 J=1,M
      T1(J)=J*DELTAT
      DO 28 K=2,N
      IF (FLW(J).LT.DISCH(IRES,K)) GO TO 29
   28 CONTINUE
   29 AMULT=(FLW(J)-DISCH(IRES,K-1))/(DISCH(IRES,K)-DISCH(IRES,K-1))
      AVSTG2=AVDPTH(IRES,K-1)+AMULT*(AVDPTH(IRES,K)-AVDPTH(IRES,K-1))
      IF (J.EQ.1) STGAR1=0.0
      IF (J.GT.1) STGAR1=STGAR(J-1)
      STGAR(J)=ABS((AVSTG2+AVSTG1)*(DELTAT/2.0))+STGAR1
      AVSTG1=AVSTG2
   30 CONTINUE
C              INITIALIZE FOR ROUTING PLUGS
      PLGVOL=0.0
      PLGTME=0.0
      STRMOT=0.0
      DO 31 JQW=1,NWQ
      QWT1(JQW)=0.0
      QWND1(JQW)=0.0
   31 CONTINUE
      VOLIN=DEAD(IRES)
      TMEIN(1)=0.0
      IF (ISTEP.EQ.4) GO TO 32
```

```
      WRITE (6,86)
      WRITE (6,87)
C              ** LOOP FOR ROUTING PLUGS **
   32 JCT=0
      NN=0
      DO 66 NNN=1,M
      JCT=JCT+1
      IF (NNN.EQ.1) PLGVOL=FLW(1)*DETK1
      IF (NNN.GT.1) PLGVOL=PLGVOL+(FLW(NNN-1)+FLW(NNN))*DETK1
      IF (JCT.LT.LR) GO TO 65
      JCT=0
      NN=NN+1
      DETTME=0.0
      DEPTH=0.0
      PLGTME=PLGTME+DELPLG
      PLGCEN=PLGTME-DELPLG/2.0
      INCR=LR*NN
      INCRM=INCR-LR
      VOLIN1=VOLIN
      VOLIN=VOLIN1+PLGVOL
      IF (VOLIN.EQ.0.0) GO TO 53
C             FIND TMEIN FROM VOLIN
      DO 33 NP=1,M
      IF (VOLIN.LT.STP(NP)) GO TO 34
   33 CONTINUE
      VOLIN=STP(M)
   34 DIFF=STP(NP)
      IF (NP.GT.1) DIFF=DIFF-STP(NP-1)
      IF (DIFF.GT.0.0001) GO TO 35
      AMULT=1.0
      GO TO 36
   35 IF (NP.GT.1) AMULT=(VOLIN-STP(NP-1))/DIFF
      IF (NP.EQ.1) AMULT=VOLIN/DIFF
   36 TMEIN(2)=AMULT*DELTAT
      IF (NP.GT.1) TMEIN(2)=TMEIN(2)+T1(NP-1)
C              FIND DETENTION TIME FOR PLUG
      VOLTME=(TMEIN(2)+TMEIN(1))/2.0
      TMEIN(1)=TMEIN(2)
      DETTME=PLGCEN-VOLTME
      IF (DETTME.LT.0.0) DETTME=0.0
C              FIND INITIAL QW CONTENT OF PLUG
      DO 37 JQW=1,NWQ
      IF (NP.EQ.1) QWT2(JQW)=AMULT*QWTOT(JQW,1)
      IF (NP.GT.1) QWT2(JQW)=QWTOT(JQW,NP-1)+AMULT*(QWTOT(JQW,NP)-QWTOT(
     1JQW,NP-1))
      QWOUT(JQW)=(QWT2(JQW)-QWT1(JQW))/QWTOT(JQW,M)
      IF (QWOUT(JQW).LT.0.0) QWOUT(JQW)=0.0
      QWT1(JQW)=QWT2(JQW)
   37 CONTINUE
C              FIND AVERAGE DEPTH OF PLUG
```

```
      DO 38 II=1,M
      IF (VOLTME.LT.T1(II)) GO TO 39
   38 CONTINUE
   39 IF (II.GT.1) GO TO 40
      STGIN=STGAR(1)*(VOLTME/DELTAT)
      GO TO 41
   40 AMULT=(VOLTME-T1(II-1))/DELTAT
      STGIN=STGAR(II-1)+AMULT*(STGAR(II)-STGAR(II-1))
   41 IF (NN.GT.1) GO TO 42
      STGOUT=STGAR(INCR)/2.0
      GO TO 43
   42 STGOUT=(STGAR(INCR)+STGAR(INCRM))/2.0
   43 IF (DETTME.EQ.0.0) GO TO 53
      DEPTH=(STGOUT-STGIN)/DETTME
      IF (DEPTH.LE.0.0) DEPTH=0.0
C              IF(JFLOW=4: FIND OUTFLOW DIST.
      IF (JFLOW(IRES).NE.4) GO TO 53
      DO 44 KP=2,N
      IF (FLW(INCR).LT.DISCH(IRES,KP)) GO TO 45
   44 CONTINUE
   45 AMULT=(FLW(INCR)-DISCH(IRES,KP-1))/(DISCH(IRES,KP)-DISCH(IRES,KP-1
     1))
      STG=STAGE(IRES,KP-1)+AMULT*(STAGE(IRES,KP)-STAGE(IRES,KP-1))
      STGR=STG/ANLAY
      JF=1
      IF (RISER(IRES).GT.STGR) GO TO 46
      JF=3
      GO TO 47
   46 STGR=STG-STGR
      IF (RISER(IRES).GE.STGR) JF=2
   47 DO 48 NL=1,NLAY
   48 OUTFL(IRES,NL)=0.0
      GO TO (49,51,52), JF
   49 DO 50 NL=1,NLAY
   50 OUTFL(IRES,NL)=100./ANLAY
      GO TO 53
   51 OUTFL(IRES,1)=100.
      GO TO 53
   52 OUTFL(IRES,NLAY)=100.
C              CONSTITUENT SETTLING COMPONENT
   53 DO 65 JQW=1,NWQ
      IF (PLGVOL.LT.1.0E-12) GO TO 63
      IF (DETTME.EQ.0.0) GO TO 61
      IF (DEPTH.LE.0.0) GO TO 63
C              ** LOOP THRU LAYERS **
      DO 59 NL=1,NLAY
      ANL=NL
      FALL(NL)=(ANL/ANLAY)*DEPTH*FIX(IRES)
      VELOC(NL)=FALL(NL)/DETTME
      DIAMT(NL)=SQRT(VELOC(NL)/STOKES)
```

```
                      SUBROUTINE RESVR(ISTEP,IRS)
C             DETERMINE PERCENT OF PARTICLES HAVING DIAMETERS
C             LESS THAN FALL DIAMETERS FROM EACH LAYER
      IF (DIAMT(NL).LT.SIZE(NS)) GO TO 54
      PERCT(NL)=100.
      VTDF=VTSUM(IRES,JQW,NS)
      GO TO 58
   54 DO 55 LP=2,NS
      IF (DIAMT(NL).LT.SIZE(LP)) GO TO 56
   55 CONTINUE
   56 PERCT(NL)=PERCNT(IRES,JQW,LP-1)+((DIAMT(NL)-SIZE(LP-1))/(SIZE(LP)-
     1SIZE(LP-1)))*(PERCNT(IRES,JQW,LP)-PERCNT(IRES,JQW,LP-1))
      IF (PERCT(NL).LT.0.0) PERCT(NL)=0.0
      IF (PERCT(NL).GT.100.) PERCT(NL)=100.
      DIV=DIAMT(NL)-SIZE(LP-1)
      IF (DIV.GT.0.0001) GO TO 57
      VTDF=VTSUM(IRES,JQW,LP-1)
      GO TO 58
   57 D2CUBE=DIAMT(NL)*DIAMT(NL)*DIAMT(NL)
      SLOPE=(PERCT(NL)-PERCNT(IRES,JQW,LP-1))/DIV
      VTDF=VTSUM(IRES,JQW,LP-1)+(SLOPE*STOKES/3.)*(D2CUBE-DCUBE(LP-1))
   58 REMN(NL)=PERCT(NL)-VTDF/VELOC(NL)
      IF (NL.EQ.1) REM(NL)=REMN(NL)
      IF (NL.GT.1) REM(NL)=NL*REMN(NL)-(NL-1)*REMN(NL-1)
   59 CONTINUE
C             SUM FOR ALL LAYERS
      QWPLG=0.0
      DO 60 NL=1,NLAY
      QWPLG=QWPLG+REM(NL)*QWOUT(JQW)*OUTFL(IRES,NL)
   60 CONTINUE
      QWPLG=QWPLG/100.
      IF (QWPLG.GT.QWOUT(JQW)*100.) QWPLG=100.*QWOUT(JQW)
      IF (QWPLG.LE.0.0) QWPLG=0.0
      GO TO 62
   61 QWPLG=100.0*QWOUT(JQW)
   62 CONTINUE
      QWND2(JQW)=QWND1(JQW)+QWPLG
      QWND1(JQW)=QWND2(JQW)
      CON(JQW,INCR)=(QWPLG/PLGVOL)*TMASS(JQW)*7.3548
      IF (CF1(JQW).LT.1.0E-6) CON(JQW,INCR)=CON(JQW,INCR)*1000.
      GO TO 64
   63 CON(JQW,INCR)=0.0
   64 CONTINUE
      IF (ISTEP.EQ.4) GO TO 65
      WRITE (6,88) PLGTME,FUP(INCR),FLW(INCR),DETTME,DEPTH,QWUP(JQW,INCR
     1),CON(JQW,INCR)
   65 CONTINUE
      STRMOT=STRMOT+PLGVOL
      PLGVOL=0.0
   66 CONTINUE
C             ** END OF LOOP FOR ROUTING PLUGS **
```

```
      SUBROUTINE RESVR(ISTEP,IRS)
      IRES=IRS
      NV=NRV(IRES)
      IF (IPR(NV).LT.1.OR.KOUT(II).LT.1) GO TO 68
      PCT=STRMOT/(STP(M)-CAPOOL(IRES))*100.
      IF (PCT.GT.100.) PCT=100.
      WRITE (6,93) PCT,ISEG(NV)
      DO 67 JQW=1,NWQ
      TRAP=(100.0-QWND2(JQW))
      WRITE (6,94) IPA(JQW),IPB(JQW),TRAP
   67 CONTINUE
   68 IF (LR.EQ.1) RETURN
      LRR=LR
      LRM=LR-1
      ALR=LR
C              INTERPOLATE FOR CONCS. AT DT INTERVALS
      DO 71 JQW=1,NWQ
      OLDCON=0.0
      DO 70 I=LRR,M,LR
      DO 69 II=1,LRM
      IV=I-LR+II
      AII=II
      CON(JQW,IV)=OLDCON+AII/ALR*(CON(JQW,I)-OLDCON)
   69 CONTINUE
      OLDCON=CON(JQW,I)
   70 CONTINUE
   71 CONTINUE
      RETURN
   72 WRITE (6,78)
      STOP
C
   73 FORMAT (A4,8A3)
   74 FORMAT (10F8.0)
   75 FORMAT (3I5,3F5.0)
   76 FORMAT (I5,2F5.0)
   77 FORMAT (1H0,46HDEAD SHOULD NOT EXCEED PERMANENT POOL CAPACITY)
   78 FORMAT (1H0,61HSIZE,STAGE,AREA,AND DISCH SHOULD BE INPUT IN INCREA
     1SING ORDER)
   79 FORMAT (1H0,46HPERCENT DATA SHOULD BE INPUT IN ASCENDING ORDER)
   80 FORMAT (//,15X,5HSTAGE,10X,4HAREA,7X,13HAVERAGE DEPTH,5X,9HDISCHAR
     1GE,7X,8HCAPACITY)
```

```
81 FORMAT (/,16X,4H(FT),9X,7H(ACRES),10X,4H(FT),10X,5H(CFS),9X,9H(ACR
  1E-FT))
82 FORMAT (/,10X,F10.4,5X,F10.5,5X,F10.2,5X,F10.2,5X,F11.5,5X)
83 FORMAT (///,5X,48H***** PARTICLE SIZE DISTRIBUTION OF INFLOW *****
  1)
84 FORMAT (//,15X,15HSIZE (MICRONS) ,10F8.1)
85 FORMAT (/,15X,8H% FINER(,2A3,1H),10F8.1)
86 FORMAT (1H1,3X,4HTIME,8X,6HINFLOW,7X,9HDISCHARGE,5X,14HDETENTION T
  1IME,8X,5HDEPTH,8X,8HINFLUENT,7X,8HEFFLUENT)
87 FORMAT (1H ,3X,5H(HRS),8X,5H(CFS),9X,5H(CFS),11X,5H(HRS),14X,4H(FT
```

SUBROUTINE RESVR(ISTEP,IRS)

```
  1),9X,6H(MG/L),9X,6H(MG/L))
 88 FORMAT (F8.2,6X,F7.2,8X,F7.2,8X,F7.2,13X,F7.2,6X,F8.1,8X,F7.1)
 89 FORMAT (//,42X,26H***** BASIN GEOMETRY *****)
 90 FORMAT (1H1,10HRESERVOIR ,A4/1H0,9X,8HNLAYER =,I2/10X,7HJFLOW =,I2
  1/10X,6HDEAD =,F7.2,9H  ACRE-FT/10X,3HN =,I3/10X,4HNS =,I3)
 91 FORMAT (1H ,9X,4HSG =,F5.2/10X,8HVISCOS =,F7.4)
 92 FORMAT (10X,7HRISER =,F6.2,5H FEET)
 93 FORMAT (///1H0,F6.2,38H PERCENT OF INFLOW TO DETENTION BASIN ,A4,2
  10H HAS BEEN DISCHARGED)
 94 FORMAT (1H ,26HBASIN TRAP EFFICIENCY FOR ,2A3,2H =,F6.2,2H %)
 95 FORMAT (10X,25HPERMANENT POOL CAPACITY =,F7.2,8H ACRE-FT)
    END
```

```
      SUBROUTINE QWLOAD(J)
C             THIS SUBROUTINE COMPUTES SIM. LOADS
C             BOTH INSTANTANEOUS AND CUMULATIVE
C             AND LOAD CHARACTERISTIC CURVES
      INTEGER TESTNO(180),RODYS
      REAL I2CFSP
      COMMON /F1/ ICT,Q(1442),R(1442),QMX,I1,DELTAT,DELPLG,NRV(10)
      COMMON /F2/ IYR,IMO,IDY,NDYS,ICK(60),JCONC,JLOAD,JRECQW
      COMMON /F3/ IFILE,IFILED,IFILEQ,JRECDS,IRECD,IRECQ,NSTRMS,IMODE
      COMMON /F4/ FUP(1442),FLAT(1442),QWUP(4,1442),QWLAT(4,1442),FLW(14
     142)
      COMMON /QWD/ CO(60,4,24),TIME(60,4,24),NWQM(60,4),RVLC(60,4)
      COMMON /ST1/ FPK(60),FVOL(60),FLD(60,4),CMX(60,4),WFALL(4,60),NWF
      COMMON /ST2/ NOFE,NF(60),NQU(4),TESTNO,KNN,CF1(4),IND(4)
      COMMON /ST3/ IPL(180),K1(60),K2(60),KOUT(180),IPA(4),IPB(4),NOPT
      COMMON /TIME/ NDELS,NOUD(180),INDP(20),NDATE(60,3)
      COMMON /UNIT/ NUDD,NWQ,UD(2881),DP(7310),RODYS,DA,ISS,PTIME,JPR
      COMMON /WQ2/ EAT(60,4),XOS(60,4),SFLD(60,4),BFL(60),RVB(60,4)
      COMMON /Z1/ IUD,ILOAD,I2CFSP,QWINT,AIMP,DAE,DT,DTS,NDTS
      IV=NWQM(I1,J)
      YMAX=CMX(I1,J)
      LK=K1(I1)
      LJ=K2(I1)
      LV=LJ-LK+1
      IF (IPL(I1).LT.1) GO TO 4
      IF (IPL(I1).GT.2) GO TO 4
C          PLOT SIMULATED CONCENTRATION DATA
      DO 1 JJJ=1,LV
    1 IF (R(JJJ).GT.YMAX) YMAX=R(JJJ)
      CALL PLT(Q,R,LV,1,YMAX,I1,J,2)
      IF (IV.EQ.0) GO TO 3
C          PLOT MEASURED CONCENTRATION DATA
      DO 2 K=1,IV
      AJ=TIME(I1,J,K)
      JJ=AJ
      JAJ=JJ-LK+1
      TME=Q(JAJ)
      F2=CO(I1,J,K)
      CALL PLT(TME,F2,1,2,YMAX,I1,J,2)
    2 CONTINUE
    3 CONTINUE
      CALL PLT(Q,R,JJJ,3,YMAX,I1,J,2)
    4 CONTINUE
C             ** INSTANTANEOUS LOADS **
C             COMPUTE SIMULATED INSTANTANEOUS LOADS
      DO 5 JJJ=1,LV
      R(JJJ)=R(JJJ)*FLW(JJJ)*CF1(J)
    5 CONTINUE
C             ** CUMULATIVE LOADS **
      YMAX=FLD(I1,J)
```

SUBROUTINE QWLOAD(J)

```
      LB=K1(I1)
      LC=K2(I1)
      IF (JLOAD.EQ.1) GO TO 6
      IF (IV.EQ.0) GO TO 6
      LB=TIME(I1,J,1)
      LC=TIME(I1,J,IV)
    6 JJJ=0
C            COMPUTE SIM. CUMULATIVE LOAD
      JJ=LB-LK
      DO 9 K=LB,LC
      JJ=JJ+1
      JJJ=JJJ+1
      R2=R(JJ)
      IF (K.LE.LB) GO TO 7
      R(JJJ)=R(JJJ-1)+(R2+R1)*PTIME*30.
      GO TO 8
    7 R(JJJ)=0.0
    8 R1=R2
      IF (K.EQ.LC) SFLD(I1,J)=R(JJJ)
    9 CONTINUE
      IF (IPL(I1).LT.2) RETURN
C            SIMULATED LOAD CHARACTERISTIC CURVE
      TLD=SFLD(I1,J)
      RUNVOL=0.0
      JJJ=0
      Q(1)=0.0
      R(1)=0.0
      JJ=LB-LK
      DO 10 K=LB,LC
      JJ=JJ+1
      JJJ=JJJ+1
      IF (K.EQ.LB) GO TO 10
      R(JJJ)=R(JJJ)/TLD
      RUNVOL=FLW(JJ)+RUNVOL
      Q(JJJ)=RUNVOL
   10 CONTINUE
      DO 11 K=1,JJJ
   11 Q(K)=Q(K)/RUNVOL
      YMAX=0.91
      CALL PLT(Q,R,JJJ,1,YMAX,I1,J,3)
      IF (IFILED.EQ.0) GO TO 15
      IF (JLOAD.EQ.1) GO TO 15
      IF (IV.EQ.0) GO TO 15
      IF (IFILED.EQ.0) GO TO 15
C            MEASURED  LOAD CHARACTERISTIC CURVE
      TLD=FLD(I1,J)
      JJJ=0
      DO 14 K=LB,LC
      JJJ=JJJ+1
      AK=K
```

SUBROUTINE QWLOAD(J)

```
      CALL QWTAB(I1,J,AK,F2,JK,IV)
      IF (K.EQ.LR) GO TO 12
      R2=F2*JD(K)
      R(JJJ)=ROLD+(R2+R1)*PTIME*30.*CF1(J)
      R1=R2
      GO TO 13
   12 R1=F2*JD(K)
      R(JJJ)=0.0
   13 ROLD=R(JJJ)
      R(JJJ)=R(JJJ)/TLD
   14 CONTINUE
      CALL PLT(Q,R,JJJ,2,YMAX,I1,J,3)
   15 CONTINUE
      CALL PLT(Q,R,JJJ,3,YMAX,I1,J,3)
      RETURN
      END
```

SUBROUTINE INPUT1

```
      SUBROUTINE INPUT1
C             THIS SUBROUTINE READS UNIT AND DAILY DATA
      INTEGER PCCN,DCCN,RODYS,DPD,DATERF,DATERL,BTIME,DATE
      INTEGER STA,STAD,STAD1,STAUP,STAUP1,STAP,STAP1
      INTEGER YR,MO,DY,EYR,EMO,EDY,CN,CT,CODE,OPTION,OPT
      INTEGER YN(99),UPD,UDD,TESTNO(180)
      DIMENSION TITLD(50), TITLUP(50), TITLP(50), IOUT(2), X2(16), MN(13
     1)
      COMMON /F2/ IYR,IMO,IDY,NDYS,ICK(60),JCONC,JLOAD,JRECQW
      COMMON /F3/ IFILE,IFILED,IFILEQ,JRECDS,IRECD,IRECQ,NSTRMS,IMODE
      COMMON /ST1/ FPK(60),FVOL(60),FLD(60,4),CMX(60,4),WFALL(4,60),NWF
      COMMON /ST2/ NOFE,NF(60),NQU(4),TESTNO,KNN,CF1(4),IND(4)
      COMMON /ST3/ IPL(180),K1(60),K2(60),KOUT(180),IPA(4),IPB(4),NOPT
      COMMON /TIME/ NDELS,NOUD(180),INDP(20),NDATE(60,3)
      COMMON /UNIT/ NUDD,NWQ,UD(2881),DP(7310),RODYS,DA,ISS,PTIME,JPR
      COMMON /WQ1/ AT,ATO(4,51),XO(4,51),CON(4,1442),NSSDAY(3,40),ISWEEP
      COMMON /Z1/ IUD,ILOAD,I2CFSP,QWINT,AIMP,DAE,DT,DTS,NDTS
      DATA IOUT/4HLIST,4HNONO/,PCCN/0/,DCCN/0/,UPD/0/,UDD/0/
      DATA MN/0,31,59,90,120,151,181,212,243,273,304,334,365/
C             JULIAN DATE FOR JAN. 1 OF EACH YEAR
C             STARTING FROM JAN. 1, 1901
      YN(1)=0
      DO 1 I=2,99
      YN(I)=YN(I-1)+365
      IF (MOD(I-1,4).EQ.0) YN(I)=YN(I)+1
    1 CONTINUE
C             INITIALIZE TO ZERO
      DO 2 I=1,IUD
    2 UD(I)=0.0
      DO 3 I=1,7310
      DP(I)=0.0
    3 CONTINUE
      DO 4 I=1,4
      DO 4 J=1,60
      WFALL(I,J)=0.0
    4 CONTINUE
      NSS=1
C             OPTION=IOUT(1) LISTS INPUT DATA.
      READ (5,51) OPTION,NWQ,NOPT,NWF,ISS,IMODE,ILOAD,ISSFRQ,JRECDS
      IF (JRECDS.GT.0.AND.IMODE.EQ.1) GO TO 50
      IF (NWQ.EQ.0) GO TO 8
      READ (5,53) (IPA(I),IPB(I),I=1,NWQ)
      READ (5,52) (NQU(I),I=1,NWQ)
      DO 5 I=1,NWQ
      IF (NQU(I).EQ.1) CF1(I)=6.243E-5
    5 IF (NQU(I).EQ.2) CF1(I)=6.243E-8
      IF (NWF.NE.1) GO TO 8
      DO 7 I=1,NWQ
      READ (5,54) IP1,IP2,(X2(J),J=1,12)
      CALL ID(IP1,IP2,NWQ,L)
```

```
      DO 6 J=1,12
    6 WFALL(L,J)=X2(J)
    7 CONTINUE
    8 IF (ISS.LT.2) GO TO 9
C           READ-IN DATES ON WHICH STREETS WERE SWEPT
      READ (5,55) (NSSDAY(1,J),NSSDAY(2,J),NSSDAY(3,J),J=1,ISS)
C             READ-IN STA.NOS. AND NAMES,DA,UNIT TIME, BEGIN AND END
C             DATES.  STATION NUMBERS READ 2A4 FOR IBM WORD SIZE.
    9 READ (5,59) STAD1,STAD,TITLD,DA
      READ (5,59) STAP1,STAP,TITLP
      READ (5,61) IYR,IMO,IDY,EYR,EMO,EDY
C             INITIALIZE VARIABLES
      DO 10 I=1,180
      NOUD(I)=0
   10 CONTINUE
      UPD=0
      PCCN=0
      NUDD=0
      READ (5,60) PTIME,IUDATA,JPUN
      JLOAD=0
      IF (JPUN.LE.0) JPUN=5
      IF (IUDATA.LT.1) IFILED=0
C             DETERMINE JULIAN DATE FOR BEGIN AND END OF RECORD.
C             DATE=1 FOR JAN. 1, 1901
      IF (MOD(IYR,4).NE.0) GO TO 11
      IF (IMO-2) 11,11,12
   11 LEAP=0
      GO TO 13
   12 LEAP=1
   13 DATERF=YN(IYR)+MN(IMO)+IDY+LEAP
      IF (MOD(EYR,4).NE.0) GO TO 14
      IF (EMO-2) 14,14,15
   14 LEAP=0
      GO TO 16
   15 LEAP=1
   16 DATERL=YN(EYR)+MN(EMO)+EDY+LEAP
C             CALCULATE NUMBER OF DAYS OF RECORD
      RODYS=DATERL-DATERF+1
      IF (RODYS.LE.NDYS) GO TO 17
      WRITE (6,75) NDYS
      STOP
   17 WRITE (6,62) STAD1,STAD,TITLD,STAP1,STAP,TITLP,DA,PTIME,IMO,IDY,IY
     1R,DATERF,EMO,EDY,EYR,DATERL
      WRITE (6,63) JRECDS,IUDATA,IMODE,JPUN
C           WRITE QW AND STREET SWEEPING INFO
      DO 18 I=1,NWQ
      J=NQJ(I)+2
      IF (I.EQ.1) WRITE (6,64) IPA(I),IPB(I),IND(J)
      IF (I.GT.1) WRITE (6,65) IPA(I),IPB(I),IND(J)
   18 CONTINUE
```

```
      IF (NWF.NE.1) GO TO 20
      DO 19 I=1,NWQ
      J=NQU(I)+2
      WRITE (6,66) IPA(I),IPB(I),IND(J),(WFALL(I,K),K=1,12)
   19 CONTINUE
   20 CONTINUE
      IF (ISS.EQ.1) WRITE (6,57) ISSFRQ
      IF (ISS.LT.2) GO TO 21
      WRITE (6,56)
      WRITE (6,58) (NSSDAY(1,J),NSSDAY(2,J),NSSDAY(3,J),J=1,ISS)
   21 CONTINUE
C             COMPUTE TIME PARAMETERS
      PDEL=PTIME/1440.0
      NDELS=1440/PTIME
      NOUT=IUD/NDELS
      NUPD=0
      BTIME=PTIME
      DT=PTIME
      QWINT=PTIME*60.
      DPD=DATERF-1
C             READ IN DATA FROM A CARD
C             PERFORM EDIT CHECK ON STATION NO., UNIT TIME, AND
C             CHRONOLOGICAL SEQUENCE OF CARD
C             ENTER DATA INTO ARRAYS ACCORDING TO CODING
C             CHECK LAST FOUR CHARACTERS OF STATION NOS. ONLY
      IF (OPTION.EQ.IOUT(1)) WRITE (6,76)
      KP=0
      NU=1
      NSD=1
   22 CONTINUE
      IF (PTIME.GT.4.9) READ (JPUN,67) STA1,STA,YR,MO,DY,CT,CN,(X2(I),I=
     11,12),CODE
      IF (PTIME.LT.4.9) READ (JPUN,68) STA1,STA,YR,MO,DY,CT,CN,(X2(I),I=
     11,12),CODE
      IF (CODE.LE.0) CODE=2
      IF (CN.LE.0) CN=1
      IF (CODE.NE.9) GO TO 23
      IF (IFILED.GT.0) WRITE (IFILED) K4DAY,(UD(I),I=1,K4DAY)
      ICK(NSD)=NU
      GO TO 35
   23 IF (MOD(YR,4).NE.0) GO TO 24
      IF (MO-2) 24,24,25
   24 LEAP=0
      GO TO 26
   25 LEAP=1
   26 DATE=YN(YR)+MN(MO)+DY+LEAP
C             DATA ENTRIES FOR CODE 2
      IF (STA.NE.STAD) GO TO 43
      IF (CT.NE.BTIME.AND.CT.GT.0) GO TO 43
      IF (DATE-UDD) 43,30,27
```

```
   27 NUDD=NUDD+1
      IPL(NUDD)=MO
      KOUT(NUDD)=DY
      TESTNO(NUDD)=YR
      NOUD(NUDD)=DATE
      UDD=DATE
      DCCN=CN
      IF (NUDD.EQ.1) GO TO 31
      ITES=NOUD(NUDD)-NOUD(NUDD-1)
      IF (ITES.EQ.1) GO TO 29
      ICK(NSD)=NU
      NU=1
      NSD=NSD+1
      IF (IUDATA.EQ.0) GO TO 33
      WRITE (IFILED) K4DAY,(UD(I),I=1,K4DAY)
      DO 28 I=1,K4DAY
   28 UD(I)=0.0
      GO TO 31
   29 NU=NU+1
      IF (NU.LE.NOUT) GO TO 31
      WRITE (6,78) NOUT
      STOP
   30 IF (CN.LE.DCCN) GO TO 43
      DCCN=CN
   31 K4DAY=NDELS*(NU-1)+12*CN
      IF (IUDATA.EQ.0) GO TO 33
C             ENTER DATA INTO ARRAYS ACCORDING TO CODE TYPE
      KK=K4DAY-11
      I=0
      DO 32 K=KK,K4DAY
      I=I+1
   32 UD(K)=X2(I)
   33 IF (OPTION.NE.IOUT(1)) GO TO 22
      IF (IFILED.GT.0) GO TO 34
      WRITE (6,69) STAD1,STA,YR,MO,DY
      GO TO 22
   34 WRITE (6,69) STAD1,STA,YR,MO,DY,CT,CN,(X2(I),I=1,12),CODE
      GO TO 22
C             DATES FOR CODES 3+4
   35 READ (5,70) STAD1,STA,YR,MO,CN,(X2(I),I=1,16),CODE
      IF (CODE.EQ.9) GO TO 44
      IF (OPTION.NE.IOUT(1)) GO TO 36
      WRITE (6,71) STAD1,STA,YR,MO,CN,(X2(I),I=1,16),CODE
   36 CONTINUE
      LEAP=0
      IF (MOD(YR,4).EQ.0) LEAP=1
      IF (CN.LT.2) GO TO 38
      DATE=YN(YR)+MN(MO)+17
      IF (MO.LE.2) GO TO 37
      DATE=DATE+LEAP
```

```
   37 II=YN(YR)+MN(MO+1)-DATE+1
      IF (MO.LE.1) GO TO 40
      II=II+LEAP
      GO TO 40
   38 DATE=YN(YR)+MN(MO)+1
      IF (MO.LE.2) GO TO 39
      DATE=DATE+LEAP
   39 II=16
   40 CONTINUE
C              DATA ENTRIES FOR CODE 3
      IF (STA.NE.STAP) GO TO 43
      IF (DATE.LE.DPD) GO TO 43
      DPD=DATE
      II=II+DPD-DATERF
      KK=DPD-DATERF+1
      IF (ISS.LT.2) GO TO 41
      IF (NSS.GT.ISS) GO TO 41
      CALL SSDAY(YR,MO,CN,KK,NSS)
   41 I=0
      DO 42 K=KK,II
      I=I+1
C              CHECK FOR GAP IN DAILY RECORD
      IF (X2(I).NE.99.990) GO TO 42
C              IF THERE IS A GAP SET UP INDICATORS FOR THIS
      KP=KP+1
      INDP(KP)=K
      X2(I)=0.0
   42 DP(K)=X2(I)
      GO TO 35
C              PRINT CARD WITH INCONSISTENT DATA
   43 WRITE (6,72) MO,YR,CN,CODE
      STOP
   44 CONTINUE
      INDP(KP+1)=II+1
      I=0
      J=1
C              CHECK FOR INPUT DATA ERRORS
      IDATE=TESTNO(NUDD)*10000+IPL(NUDD)*100+KOUT(NUDD)
      JDATE=EYR*10000+EMO*100+EDY
      IF (JDATE.GT.IDATE) GO TO 45
      WRITE (6,79)
      STOP
   45 IF (NSS.GE.ISS) GO TO 46
      WRITE (6,73)
      STOP
   46 L=NOUD(1)
      K=0
      GO TO 48
   47 WRITE (6,74) K,L
      STOP
```

```
48 DO 49 K=DATERF,DATERL
   I=I+1
   IF (DP(I).GE.0.0) GO TO 49
   IF (K.NE.L) GO TO 47
   J=J+1
   L=NOUD(J)
49 CONTINUE
   RETURN
50 WRITE (6,77)
   STOP
C
51 FORMAT (A4,I2,2I1,I2,2I1,I3,I6)
52 FORMAT (40I2)
53 FORMAT (20A3)
54 FORMAT (2A3,12F5.2)
55 FORMAT (10(I3,I2,I2))
56 FORMAT (//1H0,40HSTREETS ARE SWEPT ON THE FOLLOWING DAYS:)
57 FORMAT (//1H0,24HSTREETS ARE SWEPT EVERY ,I3,5H DAYS)
58 FORMAT (10(I4,I2,I2))
59 FORMAT (2A4,50A1,F6.2)
60 FORMAT (F5.0,3I5)
61 FORMAT (20X,3I3,3X,3I3)
62 FORMAT (1H0,22HDISCHARGE STATION      ,2A4,50A1/1H ,22HDAILY PRECIP
  1. STATION ,2A4,50A1/1H ,14HDRAINAGE AREA=,F6.2,8H SQ. MI./1H ,16HU
  2NIT DATA ARE IN,F9.3,18H MINUTE INCREMENTS/1H ,29HTHE PERIOD OF RE
  3CORD IS FROM ,I2,1H-,I2,1H-,I2,6H (DAY=,I7,5H) TO ,I2,1H-,I2,1H-,I
  42,6H (DAY=,I7,1H))
63 FORMAT (1H0,8HJRECDS =,I6/1H ,8HIUDATA =,I2/1H ,7HIMODE =,I2/1H ,6
  1HJPUN =,I3)
64 FORMAT (1H0,56HTHE FOLLOWING WATER-QUALITY CONSTITUENTS ARE SIMULA
  1TED: ,2A3,6H IN MI,A3,11HGRAMS/LITER)
65 FORMAT (1H ,56X,2A3,6H IN MI,A3,11HGRAMS/LITER)
66 FORMAT (/1H0,2A3,28H WETFALL CONTRIBUTIONS IN MI,A3,15HGRAMS/LITER
  1 ARE/1H ,2X,4HJAN.,2X,4HFEB.,1X,5HMARCH,1X,5HAPRIL,3X,3HMAY,2X,4HJ
  2UNE,2X,4HJULY,2X,4HAUG.,1X,5HSEPT.,2X,4HOCT.,2X,4HNOV.,2X,4HDEC./1
  3H ,12(1X,F5.2))
67 FORMAT (2A4,5I2,12F5.0,1X,I1)
68 FORMAT (2A4,4I2,I3,12F5.0,I1)
69 FORMAT (1H ,2A4,5I3,12F8.2,I3)
70 FORMAT (2A4,2I2,I1,16F4.2,2X,I1)
71 FORMAT (1H ,2A4,2I3,I2,16(1X,F4.2),I3)
72 FORMAT (1H0,29HERROR ON A UNIT OR DAILY CARD/1H ,35HDATE,CN, AND C
  1ODE OF THIS CARD ARE:,5X,I4,1H/,I2,5X,3HCN=,I4,5X,5HCODE=,I2)
73 FORMAT (1H0,44HERROR IN CARD ASSIGNING STREET SWEEPING DAYS)
74 FORMAT (20X,2I6/1H0,27HUNIT DAYS SPECIFIED ON UNIT,29HAND DAILY CA
  1RDS DO NOT MATCH)
75 FORMAT (1H0,30HPERIOD OF RECORD CANNOT EXCEED,I5,5H DAYS)
76 FORMAT (1H1)
77 FORMAT (1H0,33HJRECDS CANNOT EXCEED 0 IF IMODE=1)
78 FORMAT (1H ,37HPROGRAM IS DIMENSIONED FOR A MAX. OF ,I2,23H CONSEC
```

SUBROUTINE INPUT1

```
     1UTIVE STORM DAYS)                                                  Q3010
  79 FORMAT (1H0,56HEND OF RECORD MUST BE AT LEAST 1 DAY AFTER LAST UNI  Q3020
     1T DAY)                                                             Q3030
      END                                                                Q3040-
```

SUBROUTINE INPUT2

```
      SUBROUTINE INPUT2
      INTEGER RODYS,TESTNO(180),01
      REAL I2CFSP,ISEG
      COMMON /C1/ NSEG,ISEG(99),JUP(99,3),JLAT(99,4),ITYPE(99),EFF(51)
      COMMON /F2/ IYR,IMO,IDY,NDYS,ICK(60),JCONC,JLOAD,JRECQW
      COMMON /F3/ IFILE,IFILED,IFILEQ,JRECDS,IRECD,IRECQ,NSTRMS,IMODE
      COMMON /QWD/ CO(60,4,24),TIME(60,4,24),NWQM(60,4),RVLC(60,4)
      COMMON /ST1/ FPK(60),FVOL(60),FLD(60,4),CMX(60,4),WFALL(4,60),NWF
      COMMON /ST2/ NOFE,NF(60),NQU(4),TESTNO,KNN,CF1(4),IND(4)
      COMMON /ST3/ IPL(180),K1(60),K2(60),KOUT(180),IPA(4),IPB(4),NOPT
      COMMON /TIME/ NDELS,NOUD(180),INDP(20),NDATE(60,3)
      COMMON /UNIT/ NUDD,NWQ,UD(2881),DP(7310),RODYS,DA,ISS,PTIME,JPR
      COMMON /WQ1/ AT,ATD(4,51),XO(4,51),CON(4,1442),NSSDAY(3,40),ISWEEP
      COMMON /WQ2/ EAT(60,4),XOS(60,4),SFLD(60,4),BFL(60),RVB(60,4)
      COMMON /Z1/ IUD,ILOAD,I2CFSP,QWINT,AIMP,DAE,DT,DTS,NDTS
C             5280**2/12*60*60*24=26.888:  CONVERTS INCHES TO CFS
      INTVAL=(PTIME+0.001)/DT
      I2CFSP=26.8888889*DA*NDELS
      NOFE=1
      I1=1
      NSDD=0
      JRECQW=0
      JCONC=0
C             INITIALIZE VARIABLES
      DO 1 I=1,60
      FPK(I)=0.0
      FVOL(I)=0.0
      DO 1 J=1,4
      FLD(I,J)=0.0
      CMX(I,J)=0.0
      NWQM(I,J)=0
      RVLC(I,J)=0.0
      RVB(I,J)=0.0
      DO 1 K=1,24
      CO(I,J,K)=0.0
      TIME(I,J,K)=0.0
    1 CONTINUE
C             FOR EACH SET OF EVENTS, THE NO. OF EVENTS IN THE SET
C             IS ENTERED FOR AS MANY TIMES AS THERE ARE EVENTS IN THE
C             SET. A SET OF EVENTS CONSISTS OF A FRACTION OF A DAY OR
C             A SERIES OF CONTINUOUS DAYS.
      READ (5,19) I,(NF(K),K=1,I)
      WRITE (6,20) I,(NF(K),K=1,I)
C             BEGIN ANALYSIS OF A SET OF EVENTS
    2 DO 3 I=I1,NUDD
      IF (NOUD(I+1).NE.(NOUD(I)+1)) GO TO 4
    3 CONTINUE
    4 NFI1=NF(NOFE)
      I4=I1
      I1=I+1
```

SUBROUTINE INPUT2

```
      NSDD=NSDD+1
C               BEGIN ANALYSIS OF A STORM
    5 IF (NWF.EQ.2) READ (5,18) KS,KE,NWQP,(WFALL(J,NOFE),J=1,NWQ)
      IF (NWF.NE.2) READ (5,18) KS,KE,NWQP
      WRITE (6,22) NOFE,KS,KE
      JRCS=(KE-KS+1)*INTVAL
      IF (JRCS.LT.NDTS) GO TO 6
      WRITE (6,28) NOFE,NDTS
      STOP
    6 ITES=NDELS*ICK(NSDD)
      IF (KE.LE.ITES) GO TO 7
      WRITE (6,29) NOFE,ITES
      STOP
    7 JRCS=JRCS/120+1-(1-MIN0(1,MOD(JRCS,120)))
      JRCS=JRCS*NSEG*NWQ
      IF (JRECQW.LT.JRCS) JRECQW=JRCS
      IF (NWF.NE.2) GO TO 9
      DO 8 I=1,NWQ
      IF (CF1(I).GT.1.0E-6) J=3
      IF (CF1(I).LT.1.0E-6) J=4
      WRITE (6,21) IPA(I),IPB(I),WFALL(I,NOFE),IND(J)
    8 CONTINUE
    9 K1(NOFE)=KS
      K2(NOFE)=KE
      LJ=KS
      LM=KE
      NFI1=NFI1-1
      IF (NWQP.EQ.0) GO TO 14
C            READ WATER-QUALITY DATA FOR A STORM
      JCONC=1
      DO 13 K=1,NWQP
      READ (5,23) IP1,IP2,NNN
      IF (NNN.LE.24) GO TO 10
      WRITE (6,27) NOFE
      STOP
   10 CONTINUE
      CALL ID(IP1,IP2,NWQ,L)
      NWQM(NOFE,L)=NNN
      IV=NWQM(NOFE,L)
      READ (5,24) (CO(NOFE,L,J),J=1,IV)
      READ (5,24) (TIME(NOFE,L,J),J=1,IV)
      ALJ=LJ
      IVM=IV-1
      DO 11 J=1,IVM
      IF (CO(NOFE,L,J).GT.CMX(NOFE,L)) CMX(NOFE,L)=CO(NOFE,L,J)
   11 TIME(NOFE,L,J)=ALJ+TIME(NOFE,L,J)/PTIME
      ADD=TIME(NOFE,L,IV)/PTIME
      IF (AMOD(ADD,1.).EQ.0.0) GO TO 12
      IADD=ADD
      ADD=IADD+1.0
```

```
   12 TIME(NOFE,L,IV)=ALJ+ADD
      KTIME=TIME(NOFE,L,IV)
      IF (KTIME.LE.K2(NOFE)) GO TO 13
      WRITE (6,17) IP1,IP2,NOFE
      STOP
   13 CONTINUE
C             NDATE IS USED FOR PRINTING OUT THE DATE OF STORM
   14 I3=I4+LJ/NDELS
      NDATE(NOFE,1)=IPL(I3)
      NDATE(NOFE,2)=KOUT(I3)
      NDATE(NOFE,3)=TESTNO(I3)
      NOFE=NOFE+1
C             CHECK FOR MORE STORMS IN SET OF EVENTS
      IF (NFI1.GT.0) GO TO 5
C             CHECK TO SEE IF ALL EVENTS HAVE BEEN ANALYZED
      IF (NUDD.GE.I1) GO TO 2
      NOFE=NOFE-1
      READ (5,19) (KOUT(I),I=1,NOFE)
      WRITE (6,25) (KOUT(I),I=1,NOFE)
      KNN=0
      READ (5,19) (IPL(I),I=1,NOFE)
      WRITE (6,26) (IPL(I),I=1,NOFE)
      DO 15 I=1,NOFE
      TESTNO(I)=1
      KNN=KNN+1
   15 CONTINUE
      IF (IMODE.EQ.3) WRITE (6,30) JRECQW
      IF (IMODE.LT.3) JRECQW=0
      IF (IMODE.NE.2) RETURN
      DO 16 I=1,NOFE
      KOUT(I)=0
      IPL(I)=0
      DO 16 J=1,NWQ
   16 NWQM(I,J)=0.0
      JCONC=0
      RETURN
C
   17 FORMAT (1H ,18HSAMPLING TIME FOR ,2A3,10H FOR STORM,I4,29H IS AFTE
     1R KE ON CARD GROUP 14)
   18 FORMAT (2I4,I2,4F5.2)
   19 FORMAT (40I2)
   20 FORMAT (1H1,9HTHERE ARE,I4,32H STORM EVENTS GROUPED AS FOLLOWS,10I
     16/5(46X,10I6/))
   21 FORMAT (1H ,2A3,27H WETFALL CONCENTRATIONS ARE,F7.3,3H MI,A3,11HGR
     1AMS/LITER)
   22 FORMAT (1H0,9HSTORM NO.,I3,22H STARTS AT TIME PERIOD,I5,12H AND EN
     1DS AT,I5)
   23 FORMAT (2A3,I2)
   24 FORMAT (10F8.2)
   25 FORMAT (1H0,27HDETAILED OUTPUT FOR STORMS ,30I3)
```

SUBROUTINE INPUT2

```
26 FORMAT (1H0,29HTHE STORM EVENTS PLOTTED ARE ,30I3)                  R1510
27 FORMAT (1H ,27HNO. OF QW SAMPLES FOR STORM,I3,11H EXCEEDS 24)       R1520
28 FORMAT (1H0,25HNUMBER OF DT'S FOR STORM ,I2,8H EXCEEDS,I5)          R1530
29 FORMAT (1H0,12HKE FOR STORM,I3,18H SHOULD NOT EXCEED,I5)            R1540
30 FORMAT (1H0,50HNO. OF RECORDS REQUIRED FOR DIRECT ACCESS FILE 27=,  R1550
  1I7)                                                                 R1560
   END                                                                 R1570-
```

SUBROUTINE INPUT3

```
      SUBROUTINE INPUT3
      COMMON /I1/ NLU,LUSE(3,4),BK(5,4,4),ISSFRQ,NPAGE
      COMMON /ST3/ IPL(180),K1(60),K2(60),KOUT(180),IPA(4),IPB(4),NOPT
      COMMON /UNIT/ NUDD,NWQ,UD(2881),OP(7310),RODYS,DA,ISS,PTIME,JPR
      DIMENSION Z(5)
      NWQNLU=NWQ*NLU
      IF (NWQNLU.GT.1) WRITE (6,11)
      DO 5 K=1,NWQNLU
      READ (5,6) IP1,IP2,IP3,IP4,IP5,(Z(I),I=1,5)
      DO 2 L=1,NLU
      IF (LUSE(1,L).NE.IP1) GO TO 1
      IF (LUSE(2,L).NE.IP2) GO TO 1
      IF (LUSE(3,L).NE.IP3) GO TO 1
      LU=L
      GO TO 3
    1 IF (L.LT.NLU) GO TO 2
      WRITE (6,10) IP1,IP2,IP3
      STOP
    2 CONTINUE
    3 CONTINUE
      CALL ID(IP4,IP5,NWQ,IWQ)
      IF (Z(5).LE.0.001) Z(5)=Z(3)
      DO 4 I=1,5
    4 BK(I,IWQ,LU)=Z(I)
      WRITE (6,7) IPA(IWQ),IPB(IWQ),(LUSE(LV,LU),LV=1,3),(Z(I),I=1,3)
      WRITE (6,8) Z(5)
      IF (Z(4).GT.0.0001) WRITE (6,9) Z(4)
    5 CONTINUE
      RETURN
C
    6 FORMAT (3A3,2A3,6F8.3)
    7 FORMAT (//1H0,21HMODEL PARAMETERS FOR ,2A3,22H ON AREAS OF LAND US
     1E ,3A3,9H   :  K1=,F8.3/65X,3HK2=,F8.3/65X,3HK3=,F8.3)
    8 FORMAT (59X,9HDAILY K3=,F8.3)
    9 FORMAT (65X,3HK4=,F8.3)
   10 FORMAT (1H0,39HERROR IN SPECIFICATION OF LAND-USE TYPE,2X,3A3)
   11 FORMAT (1H1)
      END
```

SUBROUTINE CTCHMT

```
      SUBROUTINE CTCHMT
C           THIS SUBROUTINE IS USED TO READ IN CATCHMENT DATA
      REAL ISEG,IUP,ILAT
      COMMON /C1/ NSEG,ISEG(99),JUP(99,3),JLAT(99,4),ITYPE(99),EFF(51)
      COMMON /C2/ IPR(99),FLGTH(99),PARAM(99,4),LAND(99),NCAT,NRES
      COMMON /C3/ SSEFF(4),SSMIN(4),SSAREA(51),AKA(4),AKB(4)
      COMMON /F1/ TCT,Q(1442),R(1442),QMX,I1,DELTAT,DELPLG,NRV(10)
      COMMON /F3/ IFILE,IFILED,IFILEQ,JRECDS,IRECD,IRECQ,NSTRMS,IMODE
      COMMON /I1/ NLU,LUSE(3,4),BK(5,4,4),ISSFRQ,NPAGE
      COMMON /ST3/ IPL(180),K1(60),K2(60),KOUT(180),IPA(4),IPB(4),NOPT
      COMMON /UNIT/ NUDD,NWQ,UD(2881),DP(7310),RODYS,DA,ISS,PTIME,JPR
      COMMON /WQ1/ AT,ATO(4,51),XO(4,51),CON(4,1442),NSSDAY(3,40),ISWEEP
      COMMON /Z1/ IUD,ILOAD,I2CFSP,QWINT,AIMP,DAE,DT,DTS,NDTS
      DIMENSION ILAT(99,4), IUP(99,3)
      NCAT=1
      LAND(1)=1
      NRES=0
      READ (5,18) DAE,AIMP,SSAREA(1)
      WRITE (6,19) DAE,AIMP
      EFF(1)=DAE
      READ (5,17) NLU,(LUSE(1,I),LUSE(2,I),LUSE(3,I),I=1,NLU)
      WRITE (6,20) NLU,(LUSE(1,I),LUSE(2,I),LUSE(3,I),I=1,NLU)
      IF (ISS.EQ.0) GO TO 3
      READ (5,18) (SSEFF(J),J=1,NWQ)
      READ (5,18) (SSMIN(J),J=1,NWQ)
      WRITE (6,31)
      DO 2 J=1,NWQ
      IF (SSEFF(J).LE.1.0) GO TO 1
      WRITE (6,34) SSEFF(J)
      STOP
    1 WRITE (6,32) IPA(J),IPB(J),SSEFF(J),SSMIN(J)
    2 CONTINUE
      IF (IMODE.EQ.1) WRITE (6,33) SSAREA(1)
    3 IF (JRECDS.EQ.0) RETURN
      READ (5,21) NSEG1,(AKA(J),AKB(J),J=1,NWQ)
      IF (NSEG1.EQ.NSEG) GO TO 4
      WRITE (6,30)
      STOP
    4 NCAT=0
      WRITE (6,22) NSEG,DT
      IF (IMODE.EQ.1) RETURN
      IF (AKA(1).EQ.0.0.AND.AKB(1).EQ.0.0) GO TO 6
      WRITE (6,23)
      DO 5 J=1,NWQ
      WRITE (6,24) IPA(J),IPB(J),AKA(J),AKB(J)
    5 CONTINUE
    6 WRITE (6,26)
      DO 10 I=1,NSEG
      READ (5,27) ISEG(I),(IUP(I,J),J=1,3),(ILAT(I,J),J=1,4),ITYPE(I),IP
     1R(I),FLGTH(I),(PARAM(I,J),J=1,4),IP1,IP2,IP3
```

```
      IF (MOD(I,50).EQ.0) WRITE (6,26)
      WRITE (6,28) ISEG(I),(IUP(I,J),J=1,3),(ILAT(I,J),J=1,4),ITYPE(I),I
     1PR(I),FLGTH(I),(PARAM(I,J),J=1,4),IP1,IP2,IP3
      IF (ITYPE(I).EQ.3) NRES=NRES+1
      IF (ITYPE(I).EQ.3) NRV(NRES)=I
      IF (ITYPE(I).NE.1) GO TO 9
      DO 8 LU=1,NLU
      IF (IP1.NE.LUSE(1,LU)) GO TO 7
      IF (IP2.NE.LUSE(2,LU)) GO TO 7
      IF (IP3.NE.LUSE(3,LU)) GO TO 7
      NCAT=NCAT+1
      LAND(NCAT)=LU
      GO TO 9
    7 IF (LU.LT.NLU) GO TO 8
      WRITE (6,25) IP1,IP2,IP3
      STOP
    8 CONTINUE
    9 CONTINUE
   10 CONTINUE
      IPR(NSEG)=0
      IF (NCAT.LT.52) GO TO 11
      WRITE (6,29)
      STOP
   11 DTS=DT*60.
C              NUMBER CONTRIBUTING SEGMENTS (IUP,ILAT) USING SUBROUTINE
C              ITRAN WHICH GIVES THE CONTRIBUTING SEGMENTS THE SAME
C              NUMBER AS THE ORDER OF THE SEGMENTS (I.E., I)
      DO 13 I=1,NSEG
      DO 12 J=1,3
      X=IUP(I,J)
      JUP(I,J)=ITRAN(X)
   12 CONTINUE
      DO 13 J=1,4
      X=ILAT(I,J)
      JLAT(I,J)=ITRAN(X)
   13 CONTINUE
C              COMPUTE EFF. IMPERVIOUS AREA CONTRIBUTING TO EACH CHANNEL
      DAEC=0.0
      NC=0
      DO 16 I=1,NSEG
      IF (ITYPE(I).NE.1) GO TO 16
      NC=NC+1
      EFF(NC)=0.0
      SSAREA(NC)=0.0
      TOP=0.0
      DO 15 J=1,4
      IF (JLAT(I,J)) 15,15,14
   14 JJ=JLAT(I,J)
      IF (PARAM(JJ,2).LE.0.0) GO TO 15
      TERM=PARAM(JJ,2)*FLGTH(JJ)*FLGTH(I)/43560.
```

SUBROUTINE CTCHMT

```
      EFF(NC)=EFF(NC)+TERM
      TOP=TOP+PARAM(JJ,1)*TERM
      SSAREA(NC)=TOP/EFF(NC)
   15 CONTINUE
      DAEC=DAEC+EFF(NC)
   16 CONTINUE
      WRITE (6,35) DAEC
      RETURN
C
   17 FORMAT (I5,12A3)
   18 FORMAT (5F5.2)
   19 FORMAT (///20X,34HTOTAL EFFECTIVE IMPERVIOUS AREA = ,F7.2,6H ACRES
     1/20X,23HIMPERVIOUS RETENTION = ,F4.2,7H INCHES)
   20 FORMAT (20X,9HTHERE ARE,I2,18H LAND-USE TYPES:   ,3A3/3(49X,3A3/))
   21 FORMAT (I5,8F5.0)
   22 FORMAT (1H0,19X,26HTOTAL NUMBER OF SEGMENTS =,I3/20X,5HDT = ,F7.3,
     18H MINUTES)
   23 FORMAT (//1H0,24HPERVIOUS AREA PARAMETERS/1H0,11HCONSTITUENT,5X,3H
     1AKA,5X,3HAKB)
   24 FORMAT (1H ,2X,2A3,F11.2,F8.2)
   25 FORMAT (1H0,39HERROR IN SPECIFICATION OF LAND-USE TYPE,2X,3A3)
   26 FORMAT (1H1,55X,6HLENGTH/1H ,7HSEGMENT,1X,17HUPSTREAM SEGMENTS,3X,
     118HADJACENT SEGMENTS ,4HTYPE,4H IPR,1X,6H(FEET),8X,16HOTHER PARAME
     2TERS,8X,9H LAND USE)
   27 FORMAT (8A4,2I2,F7.0,4F5.0,3A3)
   28 FORMAT (2X,A4,3X,3(1X,A4),3X,4(1X,A4),I3,I4,F8.1,4F8.3,3A3)
   29 FORMAT (1H0,21HNCAT CANNOT EXCEED 51)
   30 FORMAT (1H0,46HNSEG DOES NOT MATCH BETWEEN DR3M AND DR3M-QUAL)
   31 FORMAT (1H0,20HSTREET SWEEPING DATA/1H0,25HCONSTITUENT  SSEFF  SSM
     1IN)
   32 FORMAT (1H ,2X,2A3,5X,F5.2,F7.2)
   33 FORMAT (1H0,8HSSAREA =,F5.2)
   34 FORMAT (1H0,41HSWEEPING EFFICIENCY SHOULD NOT EXCEED 1.0,F15.3)
   35 FORMAT (/1H0,50HEFFECTIVE IMPERVIOUS AREA BASED ON SEGMENT DATA IS
     1,F8.2,6H ACRES)
      END
```

SUBROUTINE SSDAY(YR,MO,CN,KK,NSS)

```
      SUBROUTINE SSDAY(YR,MO,CN,KK,NSS)
      INTEGER YR,CN,RODYS
C           THIS SUBROUTINE IDENTIFIES STREET SWEEPING DAYS
      COMMON /UNIT/ NUDD,NWQ,UO(2881),OP(7310),RODYS,DA,ISS,PTIME,JPR
      COMMON /WQ1/ AT,ATO(4,51),XO(4,51),CON(4,1442),NSSDAY(3,40),ISWEEP
    1 IF (MO.NE.NSSDAY(1,NSS)) RETURN
      IF (YR.NE.NSSDAY(3,NSS)) RETURN
      IF (CN.EQ.1.AND.NSSDAY(2,NSS).LE.16) GO TO 2
      IF (CN.EQ.2.AND.NSSDAY(2,NSS).GE.17) GO TO 3
      RETURN
    2 NSSDAY(1,NSS)=KK+NSSDAY(2,NSS)-1
      GO TO 4
    3 NSSDAY(1,NSS)=KK+NSSDAY(2,NSS)-17
    4 NSS=NSS+1
      IF (NSS.GT.ISS) RETURN
      GO TO 1
      END
```

```
      SUBROUTINE OUTPT(ICNT,IDEL,JJ,JOUTPT)
C             THIS SUBROUTINE OUTPUTS SIMULATED CONCENTRATION DATA
C             AND SETS UP ARRAYS FOR QWLOAD
      REAL ISEG
      COMMON /C1/ NSEG,ISEG(99),JUP(99,3),JLAT(99,4),ITYPE(99),EFF(51)
      INTEGER RODYS,TESTNO(180)
      COMMON /F1/ ICT,Q(1442),R(1442),QMX,I1,DELTAT,DELPLG,NRV(10)
      COMMON /ST2/ NOFE,NF(60),NQU(4),TESTNO,KNN,CF1(4),IND(4)
      COMMON /ST3/ IPL(180),K1(60),K2(60),KOUT(180),IPA(4),IPB(4),NOPT
      COMMON /TIME/ NDELS,NOUD(180),INDP(20),NDATE(60,3)
      COMMON /UNIT/ NUDD,NWQ,UD(2881),DP(7310),RODYS,DA,ISS,PTIME,JPR
      COMMON /WQ1/ AT,ATO(4,51),XO(4,51),CON(4,1442),NSSDAY(3,40),ISWEEP
      DIMENSION RT(5), IHR(5), TMN(5), TOUT(5), MOUT(5)
C             JOUTPT=2 IF SIMULATED CONCS. ARE LISTED
C                    =1 IF SIM. CONCS. ARE PLOTTED BUT NOT LISTED
C                    =0 IF SIM. CONCS. ARE NEITHER LISTED NOR PLOTTED
C                    =3 IF SIM. CONCS. ARE NOT AT WATERSHED OUTLET
      IF (JOUTPT.LT.2) GO TO 1
      WRITE (6,10) I1,(NDATE(I1,III),III=1,3)
      IF (JOUTPT.EQ.3) WRITE (6,11) ISEG(ICNT)
      IF (JOUTPT.NE.3) WRITE (6,12)
      WRITE (6,6) IPA(JJ),IPB(JJ)
      IF (CF1(JJ).GT.1.0E-6) WRITE (6,7)
      IF (CF1(JJ).LT.1.0E-6) WRITE (6,8)
    1 LJ=K1(I1)
      LK=K2(I1)
      QMX=0.0
      I5=0
      ICT=0
      ICNT=0
      DO 5 I=LJ,LK
      ICT=ICT+IDEL
      I5=I5+1
      RT(I5)=CON(JJ,ICT)
      IF (RT(I5).GT.QMX) QMX=RT(I5)
      IF (JOUTPT.EQ.0) GO TO 2
      MOUT(I5)=I/NDELS
      IRV=MOUT(I5)*NDELS
      TOUT(I5)=((I-IRV)*PTIME)/60.
      IHR(I5)=INT(TOUT(I5))
      TMN(I5)=AMOD(TOUT(I5),1.)*60.
    2 IF (I5.LT.5.AND.I.LT.LK) GO TO 5
      IF (JOUTPT.LT.2) GO TO 3
      WRITE (6,9) (IHR(IV),TMN(IV),RT(IV),IV=1,I5)
    3 DO 4 J=1,I5
      ICNT=ICNT+1
      R(ICNT)=RT(J)
      IF (JOUTPT.EQ.0) GO TO 4
      IF (JOUTPT.EQ.3) GO TO 4
      Q(ICNT)=TOUT(J)+MOUT(J)*24.
```

SUBROUTINE OUTPT(ICNT,IDEL,JJ,JOUTPT)

```
    4 CONTINUE
      I5=0
    5 CONTINUE
      RETURN
C
    6 FORMAT (1H ,52X,9HDATA FOR ,2A3/)
    7 FORMAT (5(7X,4HTIME,5X,5HCONC.,1X)/5(7X,5H(HRS),4X,6H(MG/L)))
    8 FORMAT (5(7X,4HTIME,5X,5HCONC.,1X)/5(7X,5H(HRS),4X,6H(UG/L)))
    9 FORMAT (5(6X,I2,1H:,F3.0,F10.3))
   10 FORMAT (1H0,26HSTORM-RUNOFF EVENT NUMBER ,I3,7H  DATED,I3,1H/,I2,1
     1H/,I2)
   11 FORMAT (1H ,52X,8HSEGMENT ,A4)
   12 FORMAT (1H ,52X,19HAT WATERSHED OUTLET)
      END
```

SUBROUTINE RITE1

```
      SUBROUTINE RITE1
C              THIS SUBROUTINE OUTPUTS A SUMMARY OF THE MEASURED DATA
      INTEGER RODYS,TESTNO(180),O1
      COMMON /QWD/ CO(60,4,24),TIME(60,4,24),NWQM(60,4),RVLC(60,4)
      COMMON /ST1/ FPK(60),FVOL(60),FLD(60,4),CMX(60,4),WFALL(4,60),NWF
      COMMON /ST2/ NOFE,NF(60),NQU(4),TESTNO,KNN,CF1(4),IND(4)
      COMMON /ST3/ IPL(180),K1(60),K2(60),KOUT(180),IPA(4),IPB(4),NOPT
      COMMON /TIME/ NDELS,NOUD(180),INDP(20),NDATE(60,3)
      COMMON /UNIT/ NUDD,NWQ,UD(2881),DP(7310),RODYS,DA,ISS,PTIME,JPR
      COMMON /WQ2/ EAT(60,4),XOS(60,4),SFLD(60,4),BFL(60),RVB(60,4)
      O1=0
      WRITE (6,4)
      DO 3 I=1,NOFE
      WRITE (6,5) I,(NDATE(I,III),III=1,3)
      WRITE (6,6) FVOL(I)
      WRITE (6,7) FPK(I)
      IF (NWQ.EQ.0) GO TO 3
      DO 2 J=1,NWQ
      IF (NWQM(I,J).EQ.0) GO TO 2
      IF (O1.GT.0) GO TO 1
      WRITE (6,8)
      WRITE (6,9)
      WRITE (6,10)
      O1=1
    1 PER=RVLC(I,J)/FVOL(I)*100.
      WRITE (6,11) IPA(J),IPB(J),RVB(I,J),RVLC(I,J),PER,NWQM(I,J),FLD(I,
     1J)
    2 CONTINUE
      O1=0
    3 CONTINUE
      RETURN
C
    4 FORMAT (1H1,45X,24HSUMMARY OF MEASURED DATA/1H ,25X,60H*RUNOFF VOL
     1UMES IN INCHES ARE BASED ON TOTAL WATERSHED AREA*)
    5 FORMAT (1H0/1H ,26HSTORM-RUNOFF EVENT NUMBER ,I3,7H  DATED,I3,1H/,
     1I2,1H/,I2)
    6 FORMAT (1H ,24HMEASURED DIRECT RUNOFF =,F9.3,7H INCHES)
    7 FORMAT (1H ,25HMEASURED PEAK DISCHARGE =,F9.2,4H CFS)
    8 FORMAT (32X,13HRUNOFF VOLUME,2X,13HRUNOFF VOLUME,2X,10HPERCENTAGE,
     15X,3HNO.,5X,8HMEASURED)
    9 FORMAT (17X,13HWATER-QUALITY,2X,13HBEFORE FIRST ,2X,12HUSED IN LOA
     1D,3X,8HOF TOTAL,7X,2HOF,6X,7HLOAD IN)
   10 FORMAT (19X,9HPARAMETER,7X,6HSAMPLE,6X,12HCOMPUTATIONS,3X,11HRUNOF
     1F VOL.,2X,7HSAMPLES,4X,6HPOUNDS)
   11 FORMAT (20X,2A3,6X,F8.3,7X,F8.3,8X,F5.1,I11,6X,F8.3)
      END
```

```
      SUBROUTINE RITE2(N1,N2)
C           THIS SUBROUTINE PRINTS OUT A SUMMARY OF
C              SIMULATED AND MEASURED LOADS
      INTEGER TESTNO(180),RODYS
      COMMON /F2/ IYR,IMO,IDY,NDYS,ICK(60),JCONC,JLOAD,JRECQW
      COMMON /F3/ IFILE,IFILED,IFILEQ,JRECDS,IRECD,IRECQ,NSTRMS,IMODE
      COMMON /I1/ NLU,LUSE(3,4),BK(5,4,4),ISSFRQ,NPAGE
      COMMON /QWD/ CO(60,4,24),TIME(60,4,24),NWQM(60,4),RVLC(60,4)
      COMMON /ST1/ FPK(60),FVOL(60),FLD(60,4),CMX(60,4),WFALL(4,60),NWF
      COMMON /ST2/ NOFE,NF(60),NQU(4),TESTNO,KNN,CF1(4),IND(4)
      COMMON /ST3/ IPL(180),K1(60),K2(60),KOUT(180),IPA(4),IPB(4),NOPT
      COMMON /UNIT/ NUDD,NWQ,UD(2881),DP(7310),RODYS,DA,ISS,PTIME,JPR
      COMMON /TIME/ NDELS,NOUD(180),INDP(20),NDATE(60,3)
      COMMON /WQ2/ EAT(60,4),XOS(60,4),SFLD(60,4),BFL(60),RVB(60,4)
      COMMON /Z1/ IUD,ILOAD,I2CFSP,QWINT,AIMP,DAE,DT,DTS,NDTS
      WRITE (6,8)
      IF (JCONC.EQ.1.AND.JLOAD.EQ.0) WRITE (6,9)
      DO 3 J=N1,N2
      WRITE (6,6) IPA(J),IPB(J)
      WRITE (6,10)
      DO 2 I=1,NOFE
      IF (FLD(I,J).EQ.0.0) GO TO 1
      IF (SFLD(I,J).EQ.0.0) GO TO 1
      VR=ALOG(FLD(I,J)/SFLD(I,J))**2
      WRITE (6,7) I,(NDATE(I,III),III=1,3),SFLD(I,J),FLD(I,J),VR
      GO TO 2
    1 WRITE (6,7) I,(NDATE(I,III),III=1,3),SFLD(I,J)
    2 CONTINUE
    3 CONTINUE
      IF (IMODE.GT.1) RETURN
      WRITE (6,8)
C          OUTPUT EAT AND ALS
      DAR=(DA*640.)/DAE
      DO 5 J=N1,N2
      IF (BK(3,J,1).LE.0.0) GO TO 5
      IF (BK(4,J,1).GT.1.E-5) GO TO 5
      AK3=BK(3,J,1)
      WRITE (6,6) IPA(J),IPB(J)
      WRITE (6,11)
      DO 5 I=1,NOFE
      IF (FLD(I,J).LE.0.0) GO TO 4
      RV1=RVLC(I,J)*DAR
      RV2=RVB(I,J)*DAR
      DIV=1.-EXP(-AK3*RV1)
      NWET=NDATE(I,1)
      IF (NWF.EQ.2) NWET=1
      PLOAD=0.277*WFALL(J,NWET)*DAE*RV1
      IF (NQU(J).EQ.2) PLOAD=PLOAD/1000.
      RLOAD=FLD(I,J)-PLOAD
      ALS=RLOAD/(EXP(-AK3*RV2)*DIV)
```

SUBROUTINE RITE2(N1,N2)

```
      ALS=ALS/DAE
      WRITE (6,12) I,EAT(I,J),ALS
      GO TO 5
    4 WRITE (6,12) I,EAT(I,J)
    5 CONTINUE
      RETURN
C
    6 FORMAT (//1H0,9HDATA FOR ,2A3/)
    7 FORMAT (1H ,I5,5X,I3,1H/,I2,1H/,I2,F12.3,2F16.3)
    8 FORMAT (1H1)
    9 FORMAT (1H ,78HMEASURED AND SIMULATED LOADS ARE BASED ON RUNOFF BE
     1TWEEN FIRST AND LAST SAMPLE)
   10 FORMAT (1H ,22X,9HSIMULATED,7X,8HMEASURED,7X,12HCONTRIBUTION/1H ,6
     1H STORM,15X,11HRUNOFF LOAD,5X,11HRUNOFF LOAD,5X,12HTO OBJECTIVE/1H
     2 ,7H NUMBER,6X,4HDATE,6X,8H(POUNDS),8X,8H(POUNDS),8X,8H FCT. 1 )
   11 FORMAT (1H ,8X,10HEQUIVALENT,4X,8HEXPECTED/1H ,7X,12HACCUMULATION,
     13X,7HINITIAL/1H ,7X,13HTIME AT START,2X,12HIMP. SURFACE/1H ,5HSTOR
     2M,4X,8HOF STORM,5X,12HLOAD (LB/AC))
   12 FORMAT (1H ,I4,5X,F9.2,4X,F10.4,3X,F10.4)
      END
```

SUBROUTINE STORM(I1)

```
      SUBROUTINE STORM(I1)
C               * STORM ANALYSIS *
C               --PEAKS,VOLUMES,RUNOFF LOADS
      INTEGER RODYS,TESTNO(180),01
      REAL I2CFSP
      COMMON /QWD/ CO(60,4,24),TIME(60,4,24),NWQM(60,4),RVLC(60,4)
      COMMON /ST1/ FPK(60),FVOL(60),FLD(60,4),CMX(60,4),WFALL(4,60),NWF
      COMMON /ST2/ NOFE,NF(60),NQU(4),TESTNO,KNN,CF1(4),IND(4)
      COMMON /ST3/ IPL(180),K1(60),K2(60),KOUT(180),IPA(4),IPB(4),NOPT
      COMMON /TIME/ NDELS,NOUD(180),INDP(20),NDATE(60,3)
      COMMON /UNIT/ NUDD,NWQ,UD(2881),DP(7310),RODYS,DA,ISS,PTIME,JPR
      COMMON /WQ2/ EAT(60,4),XOS(60,4),SFLD(60,4),BFL(60),RVB(60,4)
      COMMON /Z1/ IUD,ILOAD,I2CFSP,QWINT,AIMP,DAE,DT,DTS,NDTS
      NOFT=NOFE
      I4=I1
      NOFE=I1
C               BEGIN ANALYSIS OF A SET OF EVENTS
      DO 1 I=I1,NUDD
      IF (NOUD(I+1).NE.(NOUD(I)+1)) GO TO 2
    1 CONTINUE
    2 NFI1=NF(NOFE)
      I1=I+1
C               BEGIN ANALYSIS OF A STORM
C               FIND PEAK DISCHARGE
    3 QR=0.0
      QMX=0.0
      SRV=0.0
      LJ=K1(NOFE)
      LM=K2(NOFE)
      NFI1=NFI1-1
      DO 4 K=LJ,LM
      Q3=UD(K)
      IF (Q3.LE.QMX) GO TO 4
      QMX=Q3
    4 CONTINUE
C               FIND RUNOFF VOLUME
      DO 5 L=LJ,LM
      SRV=SRV+UD(L)
    5 CONTINUE
      FVOL(NOFE)=SRV/I2CFSP
      FPK(NOFE)=QMX
      IF (NOPT.EQ.1) WRITE (6,12) NOFE,(NDATE(NOFE,III),III=1,3)
C           COMPUTE STORM-RUNOFF LOADS
      DO 11 L=1,NWQ
      IF (NWQM(NOFE,L).EQ.0) GO TO 11
      IV=NWQM(NOFE,L)
      LB=TIME(NOFE,L,1)
      LE=TIME(NOFE,L,IV)
      IF (NOPT.NE.1) GO TO 6
      WRITE (6,13) IPA(L),IPB(L)
```

SUBROUTINE STORM(I1)

```
      IF (CF1(L).GT.1.0E-7) WRITE (6,14) IND(3)
      IF (CF1(L).LT.1.0E-7) WRITE (6,14) IND(4)
    6 J=1
      DO 9 K=L3,LE
      IF (K.EQ.L3) GO TO 7
      RVLC(NOFE,L)=RVLC(NOFE,L)+UD(K)
    7 AK=K
      CALL QWTAB(NOFE,L,AK,F2,J,IV)
      F2N=F2*UD(K)*CF1(L)
      IF (K.EQ.L3) GO TO 8
      FLD(NOFE,L)=FLD(NOFE,L)+(F2N+F20)*PTIME*30.
    8 CONTINUE
      F20=F2N
      IF (NOPT.NE.1) GO TO 9
      MOUT=K/NDELS
      MOUT=MOUT*NDELS
      TOUT=((K-MOUT)*PTIME)/60.
      IHR=TOUT
      TMN=AMOD(TOUT,1.)*60.
      WRITE (6,15) IHR,TMN,UD(K),F2,IND(J)
      J=2
    9 CONTINUE
      DO 10 K=LJ,LB
   10 RVB(NOFE,L)=RVB(NOFE,L)+UD(K)
      RVB(NOFE,L)=RVB(NOFE,L)/I2CFSP
      RVLC(NOFE,L)=RVLC(NOFE,L)/I2CFSP
   11 CONTINUE
      NOFE=NOFE+1
C               CHECK FOR MORE STORMS IN SET OF EVENTS
      IF (NFI1.GT.0) GO TO 3
      NOFE=NOFT
      I1=I4
      RETURN

   12 FORMAT (1H1/1H ,26HSTORM-RUNOFF EVENT NUMBER ,I3,7H  DATED,I3,1H/,
     1I2,1H/,I2/14X,46HTHE FOLLOWING DATA ARE USED IN COMPUTING LOADS/15
     2X,46H(MEASURED CONCENTRATIONS ARE MARKED WITH AN *))
   13 FORMAT (1H0,7X,2A3,6X,4HTIME,9X,9HDISCHARGE,4X,13HCONCENTRATION)
   14 FORMAT (1H ,19X,4H(HR),11X,5H(CFS),5X,3H(MI,A3,12HGRAMS/LITER))
   15 FORMAT (18X,I2,1H:,F3.0,2(8X,F8.2),A1)
      END
```

SUBROUTINE QWTAB(K,L,F1,F2,J,IV)

```
      SUBROUTINE QWTAB(K,L,F1,F2,J,IV)
C          THIS SUBROUTINE PERFORMS LINEAR INTERPOLATION
      COMMON /QWD/ CO(60,4,24),TIME(60,4,24),NWQM(60,4),RVLC(60,4)
      IF (F1.LT.TIME(K,L,1)) GO TO 3
      IF (F1.GE.TIME(K,L,IV)) GO TO 4
      DO 1 I=2,24
      IF (F1.GT.TIME(K,L,I)) GO TO 1
      IF (F1.EQ.TIME(K,L,I)) J=1
      GO TO 2
    1 CONTINUE
      I=24
    2 F2=CO(K,L,I-1)+(CO(K,L,I)-CO(K,L,I-1))/(TIME(K,L,I)-TIME(K,L,I-1))
     1*(F1-TIME(K,L,I-1))
      RETURN
    3 F2=CO(K,L,1)
      J=2
      RETURN
    4 F2=CO(K,L,IV)
      RETURN
      END
```

FUNCTION ITRAN(X)

```
      FUNCTION ITRAN(X)
C              THIS FUNCTION NUMBERS LATERAL AND UPSTREAM INFLOW
C              SEGMENTS TO CORRESPOND TO THE ISEG'S
      REAL ISEG
      COMMON /C1/ NSEG,ISEG(99),JUP(99,3),JLAT(99,4),ITYPE(99),EFF(51)
      I=1
    1 IF (X-ISEG(I)) 3,2,3
    2 ITRAN=I
      RETURN
    3 I=I+1
      IF (I-NSEG) 1,1,4
    4 ITRAN=0
      RETURN
      END
```

SUBROUTINE ID(IP1,IP2,NWQ,J)

```
      SUBROUTINE ID(IP1,IP2,NWQ,J)
      COMMON /ST3/ IPL(180),K1(60),K2(60),KOUT(180),IPA(4),IPB(4),NOPT
      DO 2 L=1,NWQ
      IF (IPA(L).NE.IP1) GO TO 1
      IF (IPB(L).NE.IP2) GO TO 1
      J=L
      GO TO 3
    1 IF (L.LT.NWQ) GO TO 2
      WRITE (6,4) IP1,IP2
      STOP
    2 CONTINUE
    3 CONTINUE
      RETURN
C
    4 FORMAT (1H ,26HERROR IN SPECIFICATION OF ,48HALPHANUMERIC LABEL FO
     1R WATER-QUALITY CONSTITUENT,2X,2A4)
      END
```

SUBROUTINE HEADR

```
      SUBROUTINE HEADR
      REAL ISEG
      INTEGER HEAD1(120),HEAD2(60,2),HEAD3(60),BD(3),ED(3)
      COMMON /C1/ NSEG,ISEG(99),JUP(99,3),JLAT(99,4),ITYPE(99),EFF(51)
      COMMON /F3/ IFILE,IFILED,IFILEQ,JRECDS,IRECD,IRECQ,NSTRMS,IMODE
      COMMON /F5/ HEAD1,HEAD2,HEAD3
      COMMON /Z1/ IUD,ILOAD,I2CFSP,QWINT,AIMP,DAE,DT,DTS,NDTS
      READ (IFILE'1) HEAD1,HEAD2,HEAD3
      BD(3)=HEAD1(20)/10000
      BD(1)=(HEAD1(20)/100)-(BD(3)*100)
      BD(2)=HEAD1(20)-(BD(3)*10000)-(BD(1)*100)
      ED(3)=HEAD1(21)/10000
      ED(1)=(HEAD1(21)/100)-(ED(3)*100)
      ED(2)=HEAD1(21)-(ED(3)*10000)-(ED(1)*100)
      K=HEAD1(18)
      KK=HEAD1(17)
      WRITE (6,4) (HEAD1(I),I=1,19),(BD(I),I=1,3),(ED(I),I=1,3)
      N=K+21
      WRITE (6,5) (HEAD1(J),J=22,N)
      WRITE (6,6)
      DO 1 I=1,KK
      BD(3)=HEAD3(I)/10000
      BD(1)=(HEAD3(I)/100)-(BD(3)*100)
      BD(2)=HEAD3(I)-(BD(3)*10000)-(BD(1)*100)
    1 WRITE (6,7) I,BD,HEAD2(I,1),HEAD2(I,2)
      NSEG=HEAD1(18)
      DTS=HEAD1(19)
      DT=DTS/60
      IF (HEAD1(16).GT.JRECDS) GO TO 2
      RETURN
    2 WRITE (6,3) HEAD1(16)
      STOP
C
    3 FORMAT (1H0,29HJRECDS SHOULD EQUAL OR EXCEED,I6)
    4 FORMAT (1H1,44H ***** HEADER RECORDS FROM RUNOFF FILE *****/1H0,27
     1HSTREAMFLOW STATION NUMBER -,2A4,/,17H  STATION NAME - ,13A4,/,27H
     2  NO. OF RECORDS IN FILE = ,I6,/,24H  NO. OF STORM EVENTS = ,I3,/,
     323H  NUMBER OF SEGMENTS = ,I3,/,18H  DT IN SECONDS = ,I4,/,34H  BE
     4GINNING DATE OF SIMULATION =  ,I2,1H/,I2,1H/,I2,/,30H  ENDING DATE
     5 OF SIMULATION = ,I2,1H/,I2,1H/,I2,//,13H  SEGMENT ID )
    5 FORMAT (5X,A4)
    6 FORMAT (1H1,//,72H  STORM NUMBER        DATE      STARTING RECORD NUM
     1BER     NUMBER OF VALUES )
    7 FORMAT (/,7X,I2,8X,I2,1H/,I2,1H/,I2,11X,I5,16X,I7)
      END
```

BLOCK DATA

```
BLOCK DATA
INTEGER TESTNO(180)
COMMON /ST2/ NOFE,NF(60),NQU(4),TESTNO,KNN,CF1(4),IND(4)
DATA IND/1H*,1H ,3HLLI,3HCRO/
END
```

SUBROUTINE FILES

```
      SUBROUTINE FILES
C              CREATES SPACE ON THE DIRECT ACCESS FILE FOR THE
C              NUMBER OF RECORDS REQUESTED (JRECDS).
      COMMON /F3/ IFILE,IFILED,IFILEQ,JRECDS,IRECD,IRECQ,NSTRMS,IMODE
      IFILE=25
      IF (JRECDS.LE.50) GO TO 2
      IF (JRECDS.LE.100) GO TO 3
      IF (JRECDS-500) 1,7,8
    1 IGO=JRECDS/100+1-(1-MIN0(1,MOD(JRECDS,100)))
      GO TO (3,4,5,6,7), IGO
C
    2 CONTINUE
      DEFINE FILE 25(50,480,L,IRECD)
      GO TO 28
    3 CONTINUE
      DEFINE FILE 25(100,480,L,IRECD)
      GO TO 28
    4 CONTINUE
      DEFINE FILE 25(200,480,L,IRECD)
      GO TO 28
    5 CONTINUE
      DEFINE FILE 25(300,480,L,IRECD)
      GO TO 28
    6 CONTINUE
      DEFINE FILE 25(400,480,L,IRECD)
      GO TO 28
    7 CONTINUE
      DEFINE FILE 25(500,480,L,IRECD)
      GO TO 28
    8 IGO=JRECDS/500-(1-MIN0(I,MOD(JRECDS,500)))
      IF (JRECDS.GT.10000) GO TO 29
      GO TO (9,10,11,12,13,14,15,16,17,18,19,20,21,22,23,24,25,26,27), I
     1GO
C
    9 CONTINUE
      DEFINE FILE 25(1000,480,L,IRECD)
      GO TO 28
   10 CONTINUE
      DEFINE FILE 25(1500,480,L,IRECD)
      GO TO 28
   11 CONTINUE
      DEFINE FILE 25(2000,480,L,IRECD)
      GO TO 28
   12 CONTINUE
      DEFINE FILE 25(2500,480,L,IRECD)
      GO TO 28
   13 CONTINUE
      DEFINE FILE 25(3000,480,L,IRECD)
      GO TO 28
   14 CONTINUE
```

```
      DEFINE FILE 25(3500,480,L,IRECD)
      GO TO 28
   15 CONTINUE
      DEFINE FILE 25(4000,480,L,IRECD)
      GO TO 28
   16 CONTINUE
      DEFINE FILE 25(4500,480,L,IRECD)
      GO TO 28
   17 CONTINUE
      DEFINE FILE 25(5000,480,L,IRECD)
      GO TO 28
   18 CONTINUE
      DEFINE FILE 25(5500,480,L,IRECD)
      GO TO 28
   19 CONTINUE
      DEFINE FILE 25(6000,480,L,IRECD)
      GO TO 28
   20 CONTINUE
      DEFINE FILE 25(6500,480,L,IRECD)
      GO TO 28
   21 CONTINUE
      DEFINE FILE 25(7000,480,L,IRECD)
      GO TO 28
   22 CONTINUE
      DEFINE FILE 25(7500,480,L,IRECD)
      GO TO 28
   23 CONTINUE
      DEFINE FILE 25(8000,480,L,IRECD)
      GO TO 28
   24 CONTINUE
      DEFINE FILE 25(8500,480,L,IRECD)
      GO TO 28
   25 CONTINUE
      DEFINE FILE 25(9000,480,L,IRECD)
      GO TO 28
   26 CONTINUE
      DEFINE FILE 25(9500,480,L,IRECD)
      GO TO 28
   27 CONTINUE
      DEFINE FILE 25(10000,480,L,IRECD)
   28 CONTINUE
      RETURN
   29 WRITE (6,30)
      STOP
C
   30 FORMAT (1H0,45H*****ERROR--JRECDS IS GREATER THAN 10000*****)
      END
```

SUBROUTINE FILE27

```
      SUBROUTINE FILE27
C              CREATES SPACE ON THE DIRECT ACCESS FILE FOR THE
C              NUMBER OF RECORDS REQUESTED (JRECQW).
      COMMON /F2/ IYR,IMO,IDY,NDYS,ICK(60),JCONC,JLOAD,JRECQW
      COMMON /F3/ IFILE,IFILED,IFILEQ,JRECDS,IRECD,IRECQ,NSTRMS,IMODE
      IFILEQ=27
      IF (JRECQW.LE.50) GO TO 2
      IF (JRECQW.LE.100) GO TO 3
      IF (JRECQW-500) 1,7,8
    1 IGO=JRECQW/100+1-(1-MIN0(1,MOD(JRECQW,100)))
      GO TO (3,4,5,6,7), IGO
C
    2 CONTINUE
      DEFINE FILE 27(50,480,L,IRECD)
      GO TO 18
    3 CONTINUE
      DEFINE FILE 27(100,480,L,IRECD)
      GO TO 18
    4 CONTINUE
      DEFINE FILE 27(200,480,L,IRECD)
      GO TO 18
    5 CONTINUE
      DEFINE FILE 27(300,480,L,IRECD)
      GO TO 18
    6 CONTINUE
      DEFINE FILE 27(400,480,L,IRECD)
      GO TO 18
    7 CONTINUE
      DEFINE FILE 27(500,480,L,IRECD)
      GO TO 18
    8 IGO=JRECQW/500-(1-MIN0(1,MOD(JRECQW,500)))
      IF (JRECQW.GT.5000) GO TO 19
      GO TO (9,10,11,12,13,14,15,16,17), IGO
C
    9 CONTINUE
      DEFINE FILE 27(1000,480,L,IRECD)
      GO TO 18
   10 CONTINUE
      DEFINE FILE 27(1500,480,L,IRECD)
      GO TO 18
   11 CONTINUE
      DEFINE FILE 27(2000,480,L,IRECD)
      GO TO 18
   12 CONTINUE
      DEFINE FILE 27(2500,480,L,IRECD)
      GO TO 18
   13 CONTINUE
      DEFINE FILE 27(3000,480,L,IRECD)
      GO TO 18
   14 CONTINUE
```

SUBROUTINE FILE27

```
      DEFINE FILE 27(3500,480,L,IRECD)
      GO TO 18
   15 CONTINUE
      DEFINE FILE 27(4000,480,L,IRECD)
      GO TO 18
   16 CONTINUE
      DEFINE FILE 27(4500,480,L,IRECD)
      GO TO 18
   17 CONTINJE
      DEFINE FILE 27(5000,480,L,IRECD)
   18 CONTINUE
      RETURN
   19 WRITE (6,20)
      STOP

   20 FORMAT (1H0,45H*****ERROR--JRECQW IS GREATER THAN  5000*****)
      END
```

SUBROUTINE PLT(Q,R,ICNT,IEND,YMAX,I1,J,JK)

```
      SUBROUTINE PLT(Q,R,ICNT,IEND,YMAX,I1,J,JK)
C              THIS SUBROUTINE SETS UP FOR LINE PRINTER PLOTTING
      INTEGER TESTNO(180)
      COMMON /ST2/ NOFE,NF(60),NQU(4),TESTNO,KNN,CF1(4),IND(4)
      COMMON /ST3/ IPL(180),K1(60),K2(60),KOUT(180),IPA(4),IPB(4),NOPT
      LOGICAL *1IMAGE(5200)
      DIMENSION Q(ICNT), R(ICNT)
      GO TO (1,3,4), IEND
    1 IX=Q(1)
      IY=Q(ICNT)
      XMIN=IX
      XMAX=IY+1
      IF (JK.EQ.3) XMAX=1.0
      DIV=10.
      IF (YMAX.LT.10.) DIV=0.1
      IF (YMAX.LT.0.1) DIV=0.01
      IF (YMAX.LT.0.01) DIV=0.001
      AJ=YMAX/DIV
      IAJ=AJ
      YMAX=(IAJ+1.)*DIV
      IF (JK.EQ.4) XMIN=0.0
      IF (JK.EQ.4) XMAX=YMAX
      WRITE (6,7)
      IF (JK.EQ.4) GO TO 2
      WRITE (6,5) I1
    2 IF (J.GT.0) WRITE (6,6) IPA(J),IPB(J)
      CALL PLOT2(IMAGE,XMAX,XMIN,YMAX,0.0,6)
      CALL PLOT3(1HC,Q,R,ICNT)
      RETURN
    3 CALL PLOT3(1H0,Q,R,ICNT)
      RETURN
    4 CONTINUE
      IF (JK.EQ.4) CALL PLOT4(11,11HSIM. LOADS )
      IF (JK.EQ.2.AND.CF1(J).GT.1.0E-6) CALL PLOT4(12,12HCON. IN MG/L)
      IF (JK.EQ.2.AND.CF1(J).LT.1.0E-6) CALL PLOT4(12,12HCON. IN UG/L)
      IF (JK.EQ.3) CALL PLOT4(11,11HCHAR. CURVE)
      RETURN
C
    5 FORMAT (33X,12HSTORM NUMBER,I3)
    6 FORMAT (1H ,32X,2A3)
    7 FORMAT (1H1)
      END
```

SUBROUTINE PRPLOT

```
      SUBROUTINE PRPLOT
      IMPLICIT LOGICAL*1(W),LOGICAL*1(K)
      DIMENSION NSCALE(5), ABNOS(25), X(1), Y(1)
      LOGICAL *1NOS(10)/'0','1','2','3','4','5','6','7','8','9'/
      LOGICAL *1IMAGE(1),CH,LABEL(1),ERR1,ERR3,ERR5
      LOGICAL *1VC,HC,FOR1(19),FOR2(15),FOR3(19),NC,BL,HF,HF1
      REAL *8FOX1(3),FOX2(2),FOX3(3)
      INTEGER *2VCR
      EQUIVALENCE (FOR1(1),FOX1(1)), (FOR2(1),FOX2(1)), (FOR3(1),FOX3(1)
     1), (VC,VCR)
      INTEGER FILE
      DATA HC/'-'/,NC/'+'/,BL/' '/,HF/'F'/,HF1/'.'/
      DATA FOX1/'(1XA1,F9','.2,  121','A1)      '/
      DATA FOX2/'(1XA1, 9','X121A1) '/
      DATA FOX3/'(1H0F   .','  ,  F   ','. )      '/
      DATA VCR/Z4F00/
      DATA KPLOT1/.FALSE./,KPLOT2/.FALSE./
      DATA KABSC,KORD,KBOTGL/3*.FALSE./
C
      ENTRY PLOT1(NSCALE,NHL,NSBH,NVL,NSBV)
      IFL=FILE
      ERR1=.FALSE.
      ERR3=.FALSE.
      ERR5=.FALSE.
      KPLOT1=.TRUE.
      KPLOT2=.FALSE.
      NH=IABS(NHL)
      NSH=IABS(NSBH)
      NV=IABS(NVL)
      NSV=IABS(NSBV)
      NSCL=NSCALE(1)
      IF (NH*NSH*NV*NSV.NE.0) GO TO 1
      KPLOT=.FALSE.
      ERR1=.TRUE.
      RETURN
    1 KPLOT=.TRUE.
      IF (NV.LE.25) GO TO 2
      KPLOT=.FALSE.
      ERR3=.TRUE.
      RETURN
    2 CONTINUE
      NVM=NV-1
      NVP=NV+1
      NDH=NH*NSH
      NDHP=NDH+1
      NDV=NV*NSV
      NDVP=NDV+1
      NIMG=(NDHP*NDVP)
      IF (NDV.LE.120) GO TO 3
      KPLOT=.FALSE.
```

```
      ERR5=.TRUE.
      RETURN
    3 CONTINUE
      IF (NSCL.EQ.0) GO TO 4
      FSY=10.**NSCALE(2)
      FSX=10.**NSCALE(4)
      IY=MIN0(IABS(NSCALE(3)),7)+1
      IX=MIN0(IABS(NSCALE(5)),9)+1
      GO TO 5
    4 FSY=1.
      FSX=1.
      IY=4
      IX=4
    5 FOR1(10)=NOS(IY)
      NA=MIN0(IX,NSV)-1
      NS=NA-MIN0(NA,120-NDV)
      NB=11-NS+NA
      I1=NB/10
      I2=NB-I1*10
      FOR3(6)=NOS(I1+1)
      FOR3(7)=NOS(I2+1)
      FOR3(9)=NOS(NA+1)
      IF (NV.GT.0) GO TO 7
      DO 6 J=11,18
    6 FOR3(J)=BL
      GO TO 8
    7 I1=NV/10
      I2=NV-I1*10
      FOR3(11)=NOS(I1+1)
      FOR3(12)=NOS(I2+1)
      FOR3(13)=HF
      I1=NSV/100
      I3=NSV-I1*100
      I2=I3/10
      I3=I3-I2*10
      FOR3(14)=NOS(I1+1)
      FOR3(15)=NOS(I2+1)
      FOR3(16)=NOS(I3+1)
      FOR3(17)=HF1
      FOR3(18)=FOR3(9)
    8 IF (KPLOT1) RETURN
      KPLOT1=.TRUE.
C
      ENTRY PLOT2(IMAGE,XMAX,XMIN,YMAX,YMIN,FILE)
      IFL=FILE
      KPLOT2=.TRUE.
      IF (KPLOT1) GO TO 9
      NSCL=0
      NH=5
      NSH=10
```

SUBROUTINE PRPLOT

```
      NV=10
      NSV=10
      GO TO 1
    9 CONTINUE
      IF (KPLOT) GO TO 10
      IF (ERR1) WRITE (IFL,30)
      IF (ERR3) WRITE (IFL,31)
      IF (ERR5) WRITE (IFL,32)
      RETURN
   10 YMX=YMAX
      DH=(YMAX-YMIN)/FLOAT(NDH)
      DV=(XMAX-XMIN)/FLOAT(NDV)
      DO 11 I=1,NVP
   11 ABNOS(I)=(XMIN+FLOAT((I-1)*NSV)*DV)*FSX
      DO 12 I=1,NIMG
   12 IMAGE(I)=BL
      DO 16 I=1,NDHP
      I2=I*NDVP
      I1=I2-NDV
      KNHOR=MOD(I-1,NSH).NE.0
      IF (KNHOR) GO TO 14
      DO 13 J=I1,I2
   13 IMAGE(J)=HC
   14 CONTINUE
      DO 16 J=I1,I2,NSV
      IF (KNHOR) GO TO 15
      IMAGE(J)=NC
      GO TO 16
   15 IMAGE(J)=VC
   16 CONTINUE
      XMIN1=XMIN-DV/2.
      YMIN1=YMIN-DH/2.
      RETURN
C
      ENTRY PLOT3(CH,X,Y,N3)
      IF (KPLOT2) GO TO 18
   17 WRITE (IFL,33)
   18 CONTINUE
      IF (.NOT.KPLOT) RETURN
      IF (N3.GT.0) GO TO 19
      KPLOT=.FALSE.
      WRITE (IFL,34)
      RETURN
   19 DO 26 I=1,N3
      IF (DV) 21,20,21
   20 DUM1=0
      GO TO 22
   21 CONTINUE
      DUM1=(X(I)-XMIN1)/DV
   22 IF (DH) 24,23,24
```

```
   23 DUM2=0
      GO TO 25
   24 CONTINUE
      DUM2=(Y(I)-YMIN1)/DH
   25 CONTINUE
      IF (DUM1.LT.0..OR.DUM2.LT.0.) GO TO 26
      IF (DUM1.GE.NDVP.OR.DUM2.GE.NDHP) GO TO 26
      NX=1+INT(DUM1)
      NY=1+INT(DUM2)
      J=(NDHP-NY)*NDVP+NX
      IMAGE(J)=CH
   26 CONTINUE
      RETURN
C
      ENTRY PLOT4(NL,LABEL)
      ENTRY FPLOT4(NL,LABEL)
      IF (.NOT.KPLOT) RETURN
      IF (.NOT.KPLOT2) GO TO 17
      DO 28 I=1,NDHP
      IF (I.EQ.NDHP.AND.KBOTSL) GO TO 28
      WL=BL
      IF (I.LE.NL) WL=LABEL(I)
      I2=I*NDVP
      I1=I2-NDV
      IF (MOD(I-1,NSH).EQ.0.AND..NOT.KORD) GO TO 27
      WRITE (IFL,FOR2) WL,(IMAGE(J),J=I1,I2)
      GO TO 28
   27 CONTINUE
      ORDNO=(YMX-FLOAT(I-1)*DH)*FSY
      IF (I.EQ.NDHP) ORDNO=YMIN
      WRITE (IFL,FOR1) WL,ORDNO,(IMAGE(J),J=I1,I2)
   28 CONTINUE
      IF (KABSC) GO TO 29
      WRITE (IFL,FOR3) (ABNOS(J),J=1,NVP)
   29 RETURN
C
      ENTRY OMIT(LSW)
      KABSC=MOD(LSW,2).EQ.1
      KORD=MOD(LSW,4).GE.2
      KBOTSL=LSW.GE.4
      RETURN
C
C
C
   30 FORMAT (T5,'SOME PLOT1 ARG. ILLEGALLY 0')
   31 FORMAT (T5,'NO. OF VERTICAL LINES >25')
   32 FORMAT (T5,'WIDTH OF GRAPH >121')
   33 FORMAT (T5,'PLOT2 MUST BE CALLED')
   34 FORMAT (T5,'PLOT3, ARG2 ) 0')
      END
```

ATTACHMENT G

SAMPLE RUNS

Two example computer runs of DR_3M-QUAL are shown on the following pages. For each of these runs the input data deck is listed followed by the output from the program. The first run is a lumped-parameter simulation of suspended solids in runoff from a drainage basin in Denver, Colorado. Measured water-quality and runoff data are included in the input data. Four periods of storm runoff are simulated. Concentration-versus-time and load characteristic plots are output for the first storm.

The second run is a distributed parameter run and has segments that represent pervious-area runoff and detention storage. The model reads segment flow files previously stored on disk by DR_3M. The computer run of DR_3M used to create these flow files is shown in the sample runs section of the DR_3M manual (Alley and Smith, 1982).

INPUT DATA ----- PROGRAM D33M Q   PAGE 1

```
         1         2         3         4         5         6         7         8
12345678901234567890123456789012345678901234567890123456789012345678901234567890

    1    3        13
5 SLDS
 1
22222222                                          0.2
06714310
                     71 06 01     72 05 16
  5.0
2222222271 725
2222222272 6 4
2222222272 6 5
                                                                              9
0671431071 61    0   0   0   0   0   0   2   2   1   0   0   0   0   0   0   0 3
0671431071 62    0   0   0   0   0   0   0   0   0   0   0   0   0   9         3
0671431071 71    0   0   0   0   0   0   0   0   1   0   0   0   0   0   0   0 3
0671431071 72    0   2   9   0   0   0  17   0 -70   09999   0   0   0   0
0671431072 41    09999   8   0   0   0   0   0   0   0   0   0   7  14   0   0 3
0671431072 42    0   0   5   5   1   0   0   0   0  57  39   1   5   1
0671431072 51    0   0   0   0   0   2   0   0   8   0  24   1   2   0   0   0 3
0671431072 52    0   0   2   0   0   0   0   0   0   0  10   1   0   0   0     3
0671431072 61    0   0 105 -54 -56  46   0   0   0   0  77  39   1   1   1  56
                                                                              9
 2 1 1
 219 252
 253 300
 1 0
 0 0
 57.6  .05
    1ONE
    5 0.0  1.0
FP02                          2 0 2323.        .60
IP01                          2 0 2323.        .30
PP01                          2 0 2323.        0.0
CH01             FP02IP01PP01 1 1 1200.   5.0  1.3  4.0  0.6ONE
DE01CH01                      3 1
    5
  0.0     31.     125.     500.     1000.
    1     1     5
  0.0      30.     60.      80.   100.
 0.0     3.0     5.0     7.0     9.0
 0.0     0.20    0.40    0.60    0.75
 0.0     10.     20.     30.      50.
ONE       S SLDS 15.     .20     4.5

12345678901234567890123456789012345678901234567890123456789012345678901234567890
         1         2         3         4         5         6         7         8
```

```
**************************************************
*                 U.S. GEOLOGICAL SURVEY         *
*DISTRIBUTED ROUTING RAINFALL-RUNOFF-QUALITY MODEL*
*                  VERSION  8/11/82              *
**************************************************

DISCHARGE STATION      22222222
DAILY PRECIP. STATION 06714310
DRAINAGE AREA=  0.20 SQ. MI.
UNIT DATA ARE IN     5.000 MINUTE INCREMENTS
THE PERIOD OF RECORD IS FROM  5- 1-71 (DAY=  25719) TO  6-15-72 (DAY=  26100)

JRECDS =     13
IUDATA = 0
IMODE = 3
JPUN =  5

THE FOLLOWING WATER-QUALITY CONSTITUENTS ARE SIMULATED: S SLDS IN MILLIGRAMS/LITER
```

```
***** HEADER RECORDS FROM RUNOFF FILE *****

STREAMFLOW STATION NUMBER -22222222
 STATION NAME -  HYPOTHETICAL EXAMPLE FOR LINK WITH DR3M-QUAL
 NO. OF RECORDS IN FILE =      13
 NO. OF STORM EVENTS =   2
 NUMBER OF SEGMENTS =   5
 DT IN SECONDS =  150
 BEGINNING DATE OF SIMULATION =   5/ 1/71
 ENDING DATE OF SIMULATION =  5/16/72

 SEGMENT ID
    FP02
    IP01
    PP01
    CH01
    DE01
```

STORM NUMBER	DATE	STARTING RECORD NUMBER	NUMBER OF VALUES
1	7/25/71	4	68
2	6/ 4/72	9	76

```
THERE ARE   2 STORM EVENTS GROUPED AS FOLLOWS      1      1

STORM NO.  1 STARTS AT TIME PERIOD  219 AND ENDS AT  252

STORM NO.  2 STARTS AT TIME PERIOD  263 AND ENDS AT  300

DETAILED OUTPUT FOR STORMS   1  0

THE STORM EVENTS PLOTTED ARE    0  0

NO. OF RECORDS REQUIRED FOR DIRECT ACCESS FILE 27=      5


                         TOTAL EFFECTIVE IMPERVIOUS AREA =   57.60 ACRES
                         IMPERVIOUS RETENTION = 0.05 INCHES
                         THERE ARE 1 LAND-USE TYPES:  ONE


                         TOTAL NUMBER OF SEGMENTS =  5
                         DT =   2.500 MINUTES


PERVIOUS AREA PARAMETERS

CONSTITUENT     AKA      AKB
  S SLDS        0.0      1.00
```

SEGMENT	UPSTREAM SEGMENTS	ADJACENT SEGMENTS	TYPE	IPR	LENGTH (FEET)	OTHER PARAMETERS				LAND USE
FP02			2	0	2323.0	0.0	0.600	0.0	0.0	
IP01			2	0	2323.0	0.0	0.300	0.0	0.0	
PP01			2	0	2323.0	0.0	0.0	0.0	0.0	
CH01		FP02 IP01 PP01	1	1	1200.0	5.000	1.300	4.000	0.500	ONE
OE01	CH01		3	1	0.0	0.0	0.0	0.0	0.0	

EFFECTIVE IMPERVIOUS AREA BASED ON SEGMENT DATA IS 57.60 ACRES

RESERVOIR DE01

```
NLAYER = 1
JFLOW = 1
DEAD =    0.0   ACRE-FT
N =  5
NS =  5
SG = 2.65
VISCOS = 0.0114
PERMANENT POOL CAPACITY =   0.0  ACRE-FT
```

***** BASIN GEOMETRY *****

STAGE (FT)	AREA (ACRES)	AVERAGE DEPTH (FT)	DISCHARGE (CFS)	CAPACITY (ACRE-FT)
0.0	0.0	0.0	0.0	0.0
3.0000	0.20000	1.50	10.00	0.30000
5.0000	0.40000	2.94	20.00	0.90000
7.0000	0.60000	4.24	30.00	1.90000
9.0000	0.75000	5.60	50.00	3.25000

***** PARTICLE SIZE DISTRIBUTION OF INFLOW *****

SIZE (MICRONS)	0.0	31.0	125.0	500.0	1000.0
% FINER(S SLDS)	0.0	30.0	60.0	80.0	100.0

MODEL PARAMETERS FOR S SLDS ON AREAS OF LAND USE ONE : K1= 15.000

K2= 0.200

K3= 4.600

DAILY K3= 4.600

STORM-RUNOFF EVENT NUMBER 1 DATED 7/25/71

SEGMENT CH01
DATA FOR S SLDS

TIME (HRS)	CONC. (MG/L)	TIME (HRS)	CONC. (MG/L)	TIME (HRS)	CONC. (MG/L)	TIME (HRS)	CONC. (MG/L)	TIME (HRS)	CONC. (MG/L)
18:15.	205.926	18:20.	181.407	18:25.	146.477	18:30.	100.262	18:35.	71.387
18:40.	59.582	18:45.	52.310	18:50.	50.806	18:55.	52.850	19: 0.	57.367
19: 5.	63.821	19:10.	71.929	19:15.	80.519	19:20.	91.372	19:25.	103.581
19:30.	117.109	19:35.	131.924	19:40.	148.009	19:45.	165.339	19:50.	183.898
19:55.	203.565	20: 0.	224.518	20: 5.	245.040	20:10.	268.184	20:15.	292.430
20:20.	317.744	20:25.	344.087	20:30.	371.419	20:35.	399.691	20:40.	428.858
20:45.	459.865	20:50.	489.557	20:55.	521.173	21: 0.	550.555		

STORM-RUNOFF EVENT NUMBER 1 DATED 7/25/71

AT WATERSHED OUTLET
DATA FOR S SLDS

TIME (HRS)	CONC. (MG/L)	TIME (HRS)	CONC. (MG/L)	TIME (HRS)	CONC. (MG/L)	TIME (HRS)	CONC. (MG/L)	TIME (HRS)	CONC. (MG/L)
18:15.	6.320	18:20.	23.577	18:25.	40.144	18:30.	43.874	18:35.	42.000
18:40.	37.784	18:45.	32.999	18:50.	28.605	18:55.	24.333	19: 0.	19.019
19: 5.	16.085	19:10.	13.940	19:15.	12.090	19:20.	10.643	19:25.	9.355
19:30.	8.485	19:35.	7.943	19:40.	7.676	19:45.	7.662	19:50.	7.881
19:55.	8.320	20: 0.	9.032	20: 5.	9.962	20:10.	11.105	20:15.	12.612
20:20.	14.097	20:25.	15.762	20:30.	17.359	20:35.	18.904	20:40.	20.354
20:45.	21.345	20:50.	22.300	20:55.	23.107	21: 0.	24.044		

DATA FOR S SLDS

STORM NUMBER	DATE	SIMULATED RUNOFF LOAD (POUNDS)	MEASURED RUNOFF LOAD (POUNDS)	CONTRIBUTION TO OBJECTIVE FCT. 1
1	7/25/71	164.268		
2	6/ 4/72	36.914		

```
         1         2         3         4         5         6         7         8
12345678901234567890123456789012345678901234567890123456789012345678901234567890

    11    1
~ SLDS
 1
067141O0 THIRTY-SIXTH STREET                                 3.5094
06714101 THIRTY-SIXTH STREET
                      76  4  1    75 10 10
  10.    1
0671410076042910ll                                      .20  .20  .20  .20  .20  3.1 2
067141007604291012  6.9  14.  29.  42.  55.  65.  70.  65.  60.  50.  42.  33. 2
067141007604301001  25.  21.  18.  15.  12.  10.  9.0  8.0  5.9  5.1  4.3  3.5 2
067141007604301002  3.0  3.0  3.0  3.5  3.8  4.0  4.5  4.2  4.0  4.3  4.9  4.5 2
067141007605241010                                                .15  .15  .16  .16 2
067141007605241011  .25  .43  .61 1.04 1.72  2.4  4.5  4.5  3.5  2.6 1.95 1.56 2
067141007605241012 1.25 1.09  .91  .76  .57  .57  .48  .43  .41  .39  .38       2
067141007608021007  1.1  4.5  40.  55.  55.  47.  42.  42.  43.  40.  26.  21. 2
067141007608021008  17.  13.                                                     2
067141007610061006            .28  .37  .37  .37  .37  1.0  18.  73. 163. 149. 2
067141007610061007  60.  40.  35.  42.  54.  65.  64.  50.  55.  38.  27. 18.3 2
067141007610061008  13.  11.  8.6  6.5  4.8  3.9 3.04                           2
                                                                                 9
067141017604l 0.0 0.0 0.0 0.0 0.0 .01 .01 0.0 0.0 0.0 0.0 0.0 0.0 0.0 .03 0.0  3
0671410176042 .43 0.0 0.0 .07 0.0 0.0 .02 0.0 0.0 .06 .36 0.0 -1. -1.           3
0671410176051 0.0 0.0 0.0 0.0 .02 .02 .01 0.0 0.0 0.0 0.0 0.0 0.0 0.0 .02 0.0  3
0671410176052 0.0 0.0 0.0 .02 .23 .23 0.0 -1. .08 0.0 0.0 0.0 0.0 0.0 0.0      3
0671410176061 0.0 0.0 0.0 0.0 0.0 .04 .02 0.0 0.0 0.0 0.0 0.0 0.0 0.0 0.0 0.0  3
0671410176062 .13 .38 0.0 0.0 0.0 0.0 .09 .01 0.0 .06 0.0 0.0 0.0 0.0          3
0671410176071 0.0 0.0 0.0 0.0 0.0 0.0 0.0 0.0 0.0 0.0 0.0 .01 0.0 0.0 0.0 0.0  3
0671410176072 0.0 0.0 .28 .34 .38 0.0 .05 0.0 1.3 0.0 0.0 0.0 0.0 .01 .01      3
0671410176081 .90 -1. 0.0 .27 0.0 0.0 .02 0.0 0.0 0.0 .01 .04 .03 0.0 0.0 0.0  3
0671410176082 .06 0.0 .02 0.0 0.0 0.0 0.0 .09 0.0 0.0  0.0 0.0 0.0 0. .07      3
0671410176091 0.0 0.0 0.0 0.0 0.0 0.0 0.0 0.0 0.0 0.0 0.0 0.0 .02 .23 .08 0.0  3
0671410176092 .02 0.0 .27 0.0 0.0 .15 0.0 0.0 .29 .39 .13 0.0 0.0 0.0          3
0671410176101 0.0 0.0 0.0 0.0 0.0 -1. 0.0 0.0 0.0 0.0                          3
                                                                                 9
  4 1 1 1 1
  130 170 1
5 SLDS15
  93.     127.     61.     50.     64.     110.     50.     72.     64.     51.
  52.     43.      45.     66.     31.
  20.     110.     120.    130.    140.    150.     170.    180.    190.    200.
 210.     220.     230.    240.    250.
  119 144 1
5 SLDS16
 203.     117.     96.     87.     79.     45.      46.     40.     45.     28.
  30.     36.      15.     23.     25.     24.

12345678901234567890123456789012345678901234567890123456789012345678901234567890
```

DATE 8/25/82

I_N_P_U_T D_A_T_A ----- P_R_O_G_R_A_M D_R_3_M__Q PAGE 2

```
           1         2         3         4         5         6         7         8
12345678901234567890123456789012345678901234567890123456789012345678901234567890

 85.      95.      105.     115.     125.     135.     145.     155.     175.     185.
 195.     205.     215.     225.     235.     245.
   71  87 1
S SLDS18
 48.      92.      164.     86.      183.     149.     125.     101.     102.     77.
 83.      80.      70.      73.      65.      68.      48.      48.
 30.      35.      40.      42.      45.      50.      55.      60.      65.      70.
  75.     80.      85.      95.      105.     115.     130.     145.
  53  92 1
S SLDS24
 142.     274.     908.     556.     335.     400.     372.     311.     264.     210.
 155.     137.     84.      111.     69.      75.      75.      114.     120.     137.
 99.      80.      71.      67.      43.      66.
 45.      50.      55.      60.      65.      70.      75.      80.      90.      95.
 100.     105.     110.     115.     125.     145.     155.     170.     180.     190.
 200.     215.     230.     245.     250.     285.
 1 0 0 0
 2 0 0 0
898. 0.02
    1MILE HIGH
MILE HIGHS SLDS14.43    .112     10.

12345678901234567890123456789012345678901234567890123456789012345678901234567890
           1         2         3         4         5         6         7         8
```

```
*****************************************************
*               U.S. GEOLOGICAL SURVEY              *
*DISTRIBUTED ROUTING RAINFALL-RUNOFF-QUALITY MODEL*
*                  VERSION  8/11/82                 *
*****************************************************

DISCHARGE STATION      06714100 THIRTY-SIXTH STREET
DAILY PRECIP. STATION 06714101 THIRTY-SIXTH STREET
DRAINAGE AREA=  3.51 SQ. MI.
UNIT DATA ARE IN   10.000 MINUTE INCREMENTS
THE PERIOD OF RECORD IS FROM  4- 1-76 (DAY=  27485) TO 10-10-76 (DAY=  27677)

JRECDS =      0
IUDATA = 1
IMODE = 1
JPJN =  5

THE FOLLOWING WATER-QUALITY CONSTITUENTS ARE SIMULATED: S SLDS IN MILLIGRAMS/LITER
```

```
THERE ARE    4 STORM EVENTS GROUPED AS FOLLOWS      1      1      1      1

STORM NO.  1 STARTS AT TIME PERIOD  130 AND ENDS AT  170

STORM NO.  2 STARTS AT TIME PERIOD  119 AND ENDS AT  144

STORM NO.  3 STARTS AT TIME PERIOD   71 AND ENDS AT   87

STORM NO.  4 STARTS AT TIME PERIOD   63 AND ENDS AT   92

DETAILED OUTPUT FOR STORMS   1  0  0  0

THE STORM EVENTS PLOTTED ARE   2  0  0  0


                   TOTAL EFFECTIVE IMPERVIOUS AREA =  898.00 ACRES
                   IMPERVIOUS RETENTION = 0.02 INCHES
                   THERE ARE 1 LAND-USE TYPES:  MILE HIGH


MODEL PARAMETERS FOR S SLDS ON AREAS OF LAND USE MILE HIGH   :  K1=  14.430
                                                                K2=   0.112
                                                                K3=  10.000
                                                          DAILY K3=  10.000
```

STORM-RUNOFF EVENT NUMBER 1 DATED 4/29/75

THE FOLLOWING DATA ARE USED IN COMPUTING LOADS
(MEASURED CONCENTRATIONS ARE MARKED WITH AN *)

S SLDS	TIME (HR)	DISCHARGE (CFS)	CONCENTRATION (MILLIGRAMS/LITER)
	22: 0.	3.10	93.00*
	22:10.	6.90	96.78
	22:20.	14.00	100.56
	22:30.	29.00	104.33
	22:40.	42.00	108.11
	22:50.	55.00	111.89
	23: 0.	65.00	115.67
	23:10.	70.00	119.44
	23:20.	65.00	123.22
	23:30.	60.00	127.00*
	23:40.	50.00	61.00*
	23:50.	42.00	50.00*
	0: 0.	33.00	64.00*
	0:10.	25.00	87.00
	0:20.	21.00	110.00*
	0:30.	18.00	50.00*
	0:40.	15.00	72.00*
	0:50.	12.00	64.00*
	1: 0.	10.00	51.00*
	1:10.	9.00	52.00*
	1:20.	8.00	43.00*
	1:30.	5.90	45.00*
	1:40.	5.10	56.00*
	1:50.	4.30	31.00

SUMMARY OF MEASURED DATA
RUNOFF VOLUMES IN INCHES ARE BASED ON TOTAL WATERSHED AREA

STORM-RUNOFF EVENT NUMBER 1 DATED 4/29/76
MEASURED DIRECT RUNOFF = 0.053 INCHES
MEASURED PEAK DISCHARGE = 70.00 CFS

WATER-QUALITY PARAMETER	RUNOFF VOLUME BEFORE FIRST SAMPLE	RUNOFF VOLUME USED IN LOAD COMPUTATIONS	PERCENTAGE OF TOTAL RUNOFF VOL.	NO. OF SAMPLES	MEASURED LOAD IN POUNDS
S SLDS	0.000	0.049	92.5	15	2394.857

STORM-RUNOFF EVENT NUMBER 2 DATED 5/24/76
MEASURED DIRECT RUNOFF = 0.002 INCHES
MEASURED PEAK DISCHARGE = 4.50 CFS

WATER-QUALITY PARAMETER	RUNOFF VOLUME BEFORE FIRST SAMPLE	RUNOFF VOLUME USED IN LOAD COMPUTATIONS	PERCENTAGE OF TOTAL RUNOFF VOL.	NO. OF SAMPLES	MEASURED LOAD IN POUNDS
S SLDS	0.001	0.002	65.6	16	86.855

STORM-RUNOFF EVENT NUMBER 3 DATED 8/ 2/76
MEASURED DIRECT RUNOFF = 0.033 INCHES
MEASURED PEAK DISCHARGE = 55.00 CFS

WATER-QUALITY PARAMETER	RUNOFF VOLUME BEFORE FIRST SAMPLE	RUNOFF VOLUME USED IN LOAD COMPUTATIONS	PERCENTAGE OF TOTAL RUNOFF VOL.	NO. OF SAMPLES	MEASURED LOAD IN POUNDS
S SLDS	0.000	0.032	98.7	18	1489.241

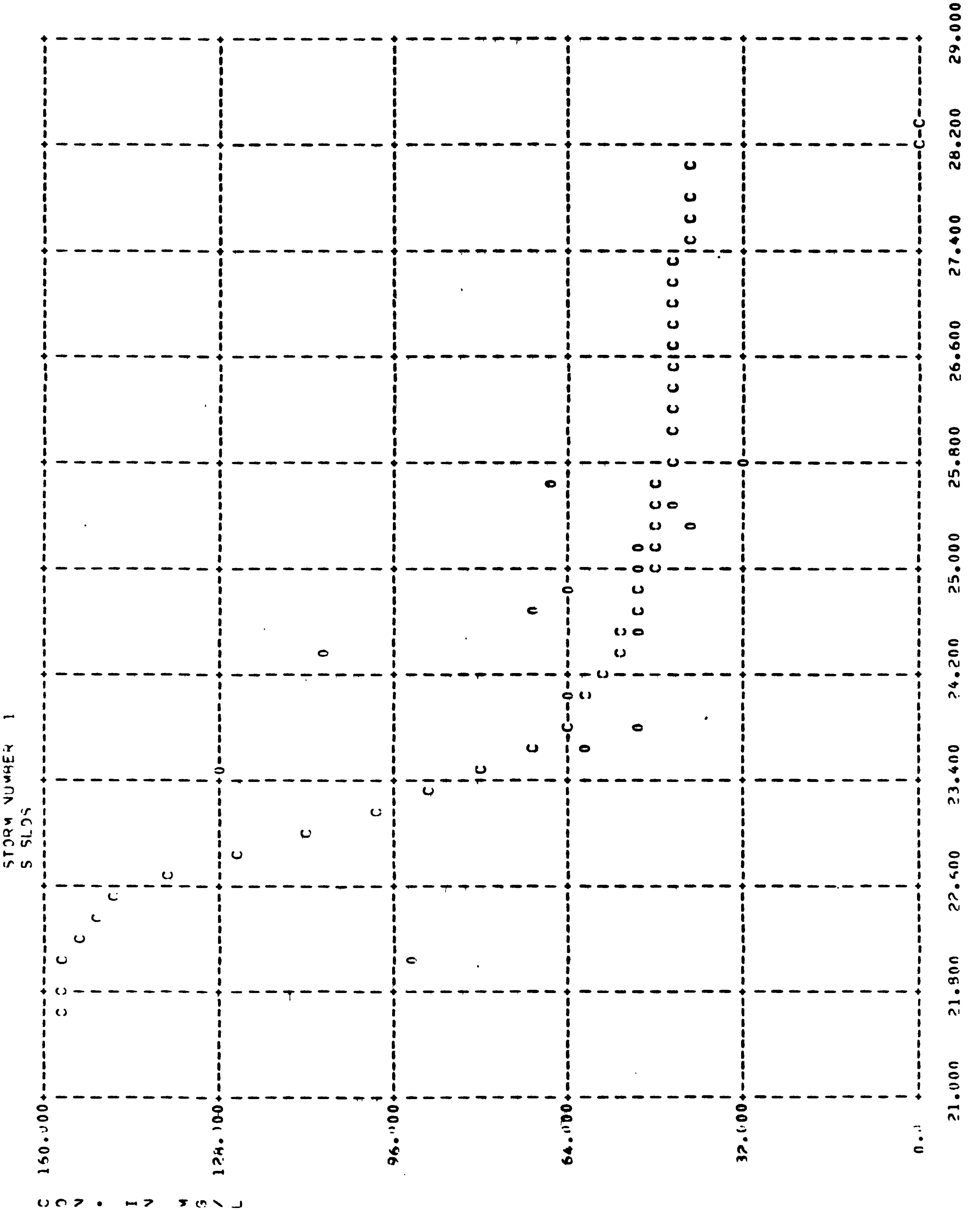
STORM-RUNOFF EVENT NUMBER 4 DATED 10/ 6/76
MEASURED DIRECT RUNOFF = 0.074 INCHES
MEASURED PEAK DISCHARGE = 163.00 CFS

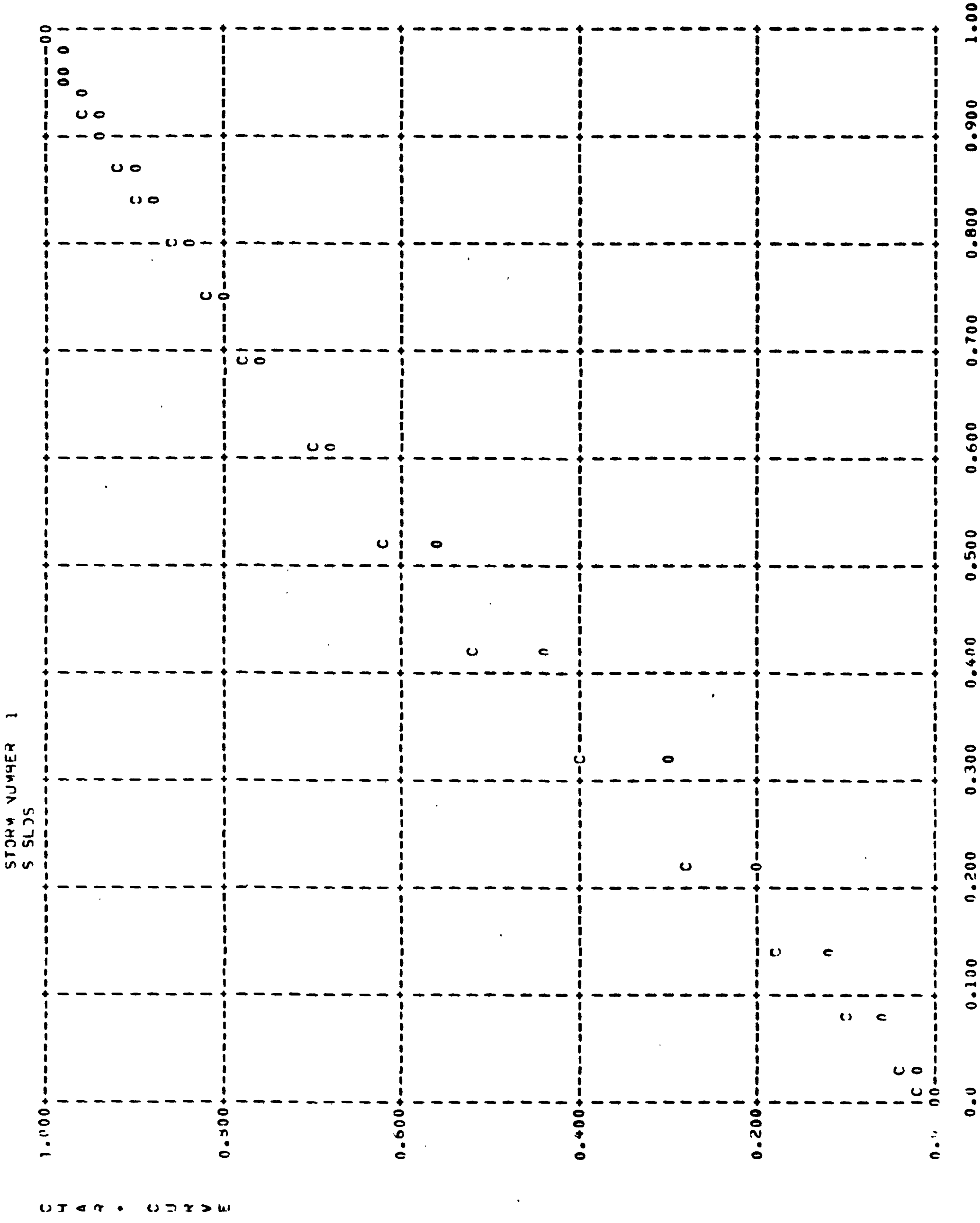
WATER-QUALITY PARAMETER	RUNOFF VOLUME BEFORE FIRST SAMPLE	RUNOFF VOLUME USED IN LOAD COMPUTATIONS	PERCENTAGE OF TOTAL RUNOFF VOL.	NO. OF SAMPLES	MEASURED LOAD IN POUNDS
S SLDS	0.000	0.073	98.7	24	7035.320

STORM-RUNOFF EVENT NUMBER 1 DATED 4/29/76

AT WATERSHED OUTLET
DATA FOR S SLDS

TIME (HRS)	CONC. (MG/L)	TIME (HRS)	CONC. (MG/L)	TIME (HRS)	CONC. (MG/L)	TIME (HRS)	CONC. (MG/L)	TIME (HRS)	CONC. (MG/L)
21:40.	157.555	21:50.	157.393	22: 0.	156.779	22:10.	155.197	22:20.	151.866
22:30.	145.822	22:40.	136.734	22:50.	125.225	23: 0.	112.410	23:10.	99.647
23:20.	88.192	23:30.	78.564	23:40.	71.103	23:50.	65.285	0: 0.	60.910
0:10.	57.563	0:20.	55.202	0:30.	53.240	0:40.	51.645	0:50.	50.363
1: 0.	49.330	1:10.	48.463	1:20.	47.722	1:30.	47.113	1:40.	46.624
1:50.	46.222	2: 0.	45.888	2:10.	45.605	2:20.	45.350	2:30.	45.095
2:40.	44.823	2:50.	44.526	3: 0.	44.205	3:10.	43.866	3:20.	43.523
3:30.	43.190	3:40.	42.853	3:50.	42.499	4: 0.	42.152	4:10.	0.0
4:20.	0.0								


STORM NUMBER 1
S SLDS
CONC. IN MG/L
160.000
128.000
96.000
64.000
32.000
0.0
21.000
21.800
22.600
23.400
24.200
25.000
25.800
26.600
27.400
28.200
29.000


STORM NUMBER 1
0.0
0.100
0.200
0.300
0.400
0.500
0.600
0.700
0.800
0.900
1.000
0.200
0.400
0.600
0.800
1.000

STORM-RUNOFF EVENT NUMBER 4 DATED 10/ 6/75
THE FOLLOWING DATA ARE USED IN COMPUTING LOADS
(MEASURED CONCENTRATIONS ARE MARKED WITH AN *)

SLDS	TIME (HR)	DISCHARGE (CFS)	CONCENTRATION (MILLIGRAMS/LITER)
	11:10.	0.37	142.00
	11:20.	1.00	274.00*
	11:30.	18.00	556.00*
	11:40.	73.00	400.00*
	11:50.	153.00	311.00*
	12: 0.	149.00	254.00*
	12:10.	50.00	165.00*
	12:20.	40.00	94.00*
	12:30.	35.00	90.00
	12:40.	42.00	70.75
	12:50.	54.00	74.25
	13: 0.	65.00	75.50
	13:10.	54.00	88.00
	13:20.	50.00	114.00*
	13:30.	55.00	120.00*
	13:40.	38.00	137.00*
	13:50.	27.00	99.00*
	14: 0.	18.30	86.33
	14:10.	13.00	77.00
	14:20.	11.00	71.00*
	14:30.	8.60	69.00
	14:40.	6.50	67.00

STORM-RUNOFF EVENT NUMBER 3 DATED 8/ 2/75

THE FOLLOWING DATA ARE USED IN COMPUTING LOADS

(MEASURED CONCENTRATIONS ARE MARKED WITH AN *)

S SLDS	TIME (HR)	DISCHARGE (CFS)	CONCENTRATION (MILLIGRAMS/LITER)
	12:20.	4.50	48.00*
	12:30.	40.00	154.00*
	12:40.	55.00	149.00*
	12:50.	55.00	101.00*
	13: 0.	47.00	77.00*
	13:10.	42.00	80.00*
	13:20.	42.00	71.50
	13:30.	43.00	59.00
	13:40.	40.00	56.50
	13:50.	26.00	51.33
	14: 0.	21.00	48.00*
	14:10.	17.00	48.00
	14:20.	13.00	48.00

STORM-RUNOFF EVENT NUMBER 2 DATED 5/24/75

THE FOLLOWING DATA ARE USED IN COMPUTING LOADS

(MEASURED CONCENTRATIONS ARE MARKED WITH AN *)

5 SLDS	TIME (HR)	DISCHARGE (CFS)	CONCENTRATION (MILLIGRAMS/LITER)
	21:10.	4.50	203.00
	21:20.	4.50	150.00
	21:30.	3.50	106.50
	21:40.	2.60	91.50
	21:50.	1.95	83.00
	22: 0.	1.66	52.00
	22:10.	1.25	45.50
	22:20.	1.09	44.50
	22:30.	0.91	41.50
	22:40.	0.75	42.50
	22:50.	0.57	36.50
	23: 0.	0.57	29.00
	23:10.	0.48	33.00
	23:20.	0.43	25.50
	23:30.	0.41	19.00
	23:40.	0.39	24.50
	23:50.	0.38	25.33
	0: 0.	0.0	24.00

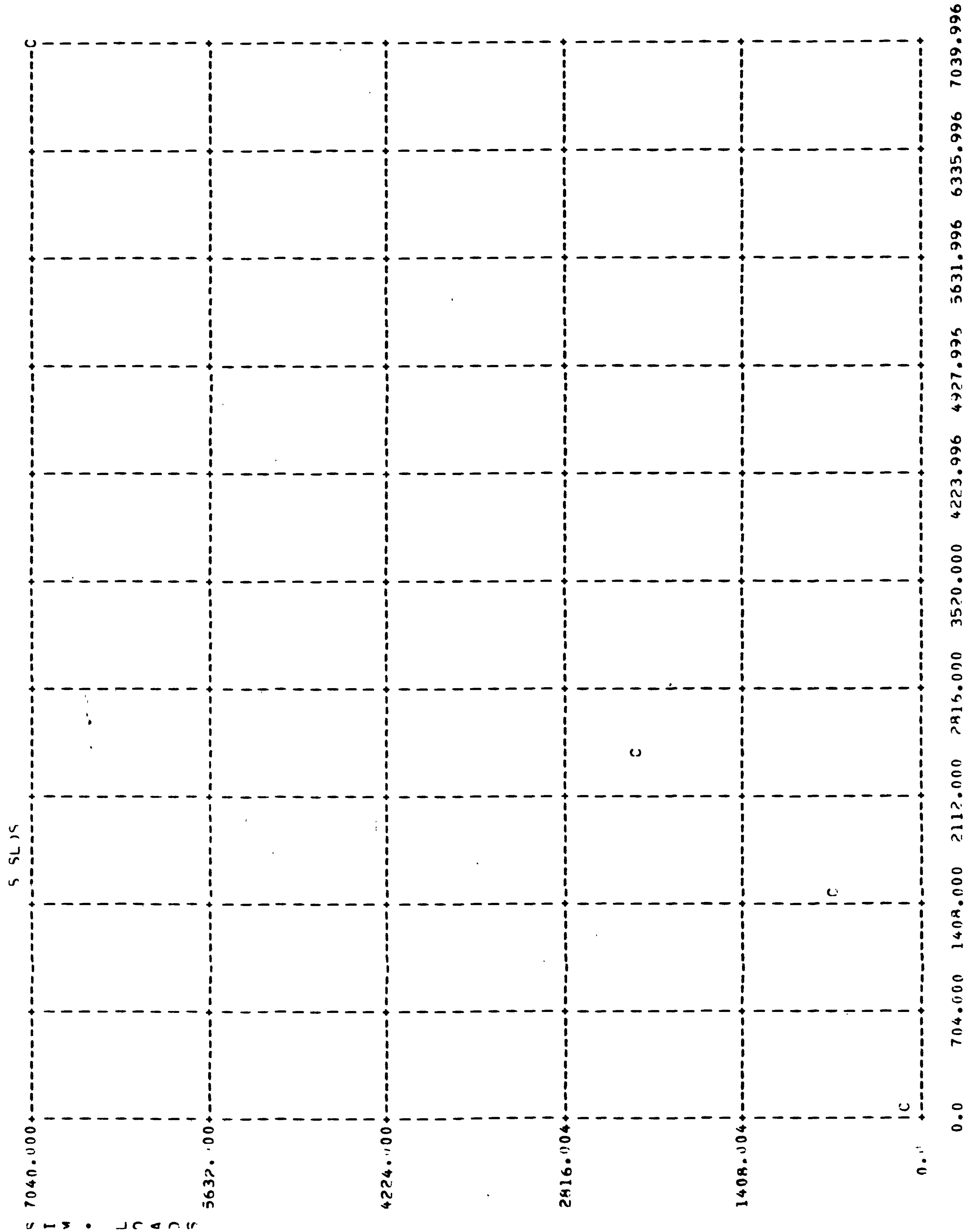
MEASURED AND SIMULATED LOADS ARE BASED ON RUNOFF BETWEEN FIRST AND LAST SAMPLE

DATA FOR S SLDS

STORM NUMBER	DATE	SIMULATED RUNOFF LOAD (POUNDS)	MEASURED RUNOFF LOAD (POUNDS)	CONTRIBUTION TO OBJECTIVE FCT. 1
1	4/29/75	2255.158	2394.857	0.004
2	5/24/75	135.397	86.855	0.197
3	8/ 2/75	749.408	1489.241	0.472
4	10/ 6/75	6989.500	7035.320	0.000

DATA FOR S SLDS

STORM	EQUIVALENT ACCUMULATION TIME AT START OF STORM	EXPECTED INITIAL IMP. SURFACE LOAD (LB/AC)
1	2.54	3.8015
2	2.55	2.5508
3	0.99	3.0141
4	9.27	9.3709


0.0
704.000
7039.996
6335.996
5631.996
4223.996
3520.000
4224.000