

Appendix B

Compute Functions

Appendix B - Compute Functions

Index of Functions

		<u>Page</u>
ABS	Absolute function	3
ACC	Running accumulation	3
AMREG	Apply multiple linear regression equation.	3
CONIC	Conic interpolation from elevation/area table	4
CORR	Compute correlation coefficients	4
COS	Cosine trigonometric function	5
COUNT	Count the number of valid and missing data values	5
CMA	Centered moving average smoothing	5
DDT	Differences per unit time	6
DECPAR	Decaying basin wetness parameter	6
DIFF	Successive differences	7
ESTLIN	Estimate values for missing data	7
ESTPPT	Estimate values for missing precipitation data	7
FMA	Forward moving average	7
GENTSR	Generate a regular interval time series	8
INT	Truncate to whole numbers	8
LAST	Last valid value in a time series	8
LOG	Natural log base "e"	8
LOG10	Log base 10	9
MATE	Generate data pairs from two time-series	9
MAX	Maximum value in a time series	9
MEAN	Mean value in a time series	9
MIN	Minimum value in a time series	10
MREG	Multiple linear regression function	10
MRG	Merge two time series	10
MRGP	Merge two paired data series	11
MUSK	Muskingum hydrologic routing	11
NINT	Round to nearest whole number	11
OLY	Olympic smoothing	11
PERCON	Period constants	12
POLY	Polynomial transformation	12
POLY2	Polynomial transformation with integral	13
PULS	Modified Puls or Working R&D routing function	13
QAC	Flow accumulator gage processor	13
RND	Round off	14
RTABLE	Rating table interpolation	14
RTABLR	Reverse rating table interpolation	14
RTABL2	Two-variable rating table interpolation	15
SCRN1	Screen for possible erroneous values based on maximum/minimum range	15

SCRN2	Screen for possible erroneous values	15
SDEV	Compute the standard deviation of one independent variable.	16
SELECT	Extract time series data at unique time specification.	16
SHIFT	Shift adjustment	17
SIN	Sine trigonometric function	17
SKEW	Compute the skew coefficient of one independent variable	17
SQRT	Square root function	18
SS	Straddle Stagger routing function	18
SSW	Wilmington District Straddle Stagger routing function	18
TAN	Tangent trigonometric function	19
TS1	Interpolated data at regular intervals	19
TS2	Period averages at regular intervals	19
TS3	Period mins or maxs at regular intervals	20
TS4	Interpolated data at irregular intervals	20
TSCYCL	Time series cyclic analysis	20
TSHIFT	Shift time series in time	21
TSNAP	Snap Irregular times to nearest Regular period	21
TTSR	Transform time series to regular	21
TTSI	Transform time series to irregular	22
1/X	Inverse function	22

Name: ABS Absolute function

Use: CO TY=ABS(TX)

Description: Compute the absolute value of TX and store in TY. TX and TY can be the same variable. TX can be scalar, time series or paired data. If a value of TX is undefined resulting values of TY are undefined. If a concurrent IF condition is not satisfied, TY is unchanged. If TY is a time series, its units and type are set equal to the units and type of TX.

Name: ACC Running accumulation

Use: CO TY=ACC(TX)

Description: Compute a running accumulation of the values in TX and store the result in TY. TX and TY can be the same variable. If a TX value is undefined or a concurrent IF condition is not satisfied, the value of TX is not added to the accumulation and the corresponding TY remains the same as the previous TY. Units of TY are the same as those of TX. If TX is typed as PER-AVER or PER-CUM, TY will be typed INST-VAL or INST-CUM, respectively.

Name: AMREG Apply multiple linear regression equation.

Use: CO TY=MREG(PF, TX1, TX2, ..., TXn, min, max)

Description: This function is used to calculate a regular time series (TY) based on a linear regression equation of the general form ($Y = B_0 + B_1 \cdot X_1 + B_2 \cdot X_2 + B_n \cdot X_n$). Where Y is the dependent variable and the X1, X2, and Xn are independent variables. The AMREG function requires that the function parameters TY, TX1, TX2, and TXn be regular time series variables and that they all have the same time interval. It is further required that the time variables be retrieved in the same sequential order as specified in the function parameter list. In other words TX1 is retrieved from DSS first followed by TX2 and so on. The number of independent time series variables that can be specified is limited to one less the maximum number of time series variables that the program allows. On the DOS PC this value will be 4 independent variables. The linear regression coefficients (B0 - Bn) are stored in a paired function variable (PF) which is normally retrieved from DSS. Two optional function parameters "min max" are available to specify a range of values that control which values in the dependent variable are to be accepted as valid computed values. These two parameters must be specified as the last two parameters and if specified, both must be

specified as real numbers. Units of TY are the same as those of TX1.
An IF condition has no effect.

Name: CONIC Conic interpolation from elevation/area table

Use: CO TY=CONIC(TX,TB,input,output,[scale])

Description: Interpolate values for TX using conic interpolation table TB and store the result in TY. TX and TY variables must be regular or irregular time series data. TB is a paired data variable that can be created using program DSSPD. The first paired data values contain the initial conic depth in feet or meters "x(1)" and storage in acre-feet or cubic meters below the first elevation "y(1)". The rest of the paired values "x(2-),y(2-)" contain the elevation in ft-msl or m-msl and area in acres or sq. meters. If the initial conic depth is undefined, the function will calculate one. The third and fourth parameters are used to specify the type of input (TX) and output (TY) data desired. The input type has to be either STORAGE or ELEVATION. The output type has to be either STORAGE, ELEVATION, or AREA. An optional fifth parameter can be used to specify a scale value to use with the input and output storage values. The default scale value is 1.0. An IF condition has no effect. The units and type must be set by the use of the "SD" command.

Name: CORR Compute correlation coefficients

Use: CO SY=CORR(TX1, TX2, INDEX)

Description: Compute the number of valid pairs for correlation, regression constant, regression coefficient, determination coefficients, standard errors of regression, determination coefficient adjusted for degrees of freedom, and standard error adjusted for degrees of freedom between the two variables TX1 and TX2. TX1 and TX2 must be of the same type (either uniform time series or irregular time series) and contain the same number of rows or ordinates. An IF condition has no effect. Variable INDEX is used to set the scalar variable SY as follows:

Index	Description
1	Number of valid pairs for correlation.
2	Regression constant.
3	Regression coefficient.
4	Determination coefficient.
5	Standard error of regression.
6	Determination coefficient adjusted for degrees of freedom.
7	Standard error adjusted for degrees of freedom.

Name: COS Cosine trigonometric function

Use: CO TY=COS(TX)

Description: Compute the cosine of TX in radians and store in TY. TX and TY can be the same variable. TX can be scalar, time series or paired data. If a value of TX is undefined resulting values of TY are undefined. If a concurrent IF condition is not satisfied, TY is unchanged. The units and type of TY are set equal to the units and type of TX

Name: COUNT Count the number of valid and missing data values

Use: CO SY=COUNT(TX, TYPE)

Description: This function is used to count the valid and missing values in time series variable TX and stores it in scalar variable SY. Possible values for TYPE are: "VALID" and "MISSING". If a concurrent IF condition is not satisfied, then the number of valid values is not incremented and the value is assumed to be missing and the missing counter incremented.

Name: CMA Centered moving average smoothing

Use: CO TY=CMA(TX,NP[,CLEVEL])

Description: Compute a centered, moving average of NP values in TX and store in TY. NP must be odd and greater than 2. The default CLEVEL is "LEVEL3" and NP/2 values at the beginning and end of TY will be undefined. If a value in TX is undefined or a concurrent IF condition is not satisfied, then the resulting TY values are the averages of one less value. TX must be a regular-interval time-series and TY and TX cannot be the same variable. Useful for filtering instantaneous data containing high-frequency variations. Units and type are set equal to TX values. The following levels are available:

LEVEL1 Only valid values will be averaged. Valid values that do not have all valid values within the averaging period number will be missing. Values at the start and end of the data that do not have enough valid values to average will be assigned missing values.

LEVEL2 Only valid values will be averaged. Valid values that do not have all valid values within the averaging period number will be missing. Valid values at the start and end of the data that do not have enough valid values to average will be averaged over a reduced number of values.

LEVEL3 (Default setting) All values will be averaged based on valid values within the averaging period number. If all values are missing for the period number, the value will be missing. Values at the start and end of the data that do not have enough valid values to average will be assigned missing values.

LEVEL4 All values will be averaged based on valid values within the averaging period number. If all values are missing for the period number, the value will be missing. Values at the start and end of the data that do not have valid values to average will be averaged based on a reduced number of values.

Note: Questionable and estimated flagged values are used in the computations.

Name: DDT Differences per unit time

Use: CO TY=DDT(TX)

Description: Computes the successive differences in TX per time period:

$$TY(t)=(TX(t)-TX(t-1))/DT$$

where DT is the time difference in days between t and $t-1$. If a value of TX is undefined, TY is undefined. If a concurrent IF condition is not satisfied, TY is unchanged. TY and TX cannot be the same variable. TY is assigned the type 'PER-AVER.' The units of TY are undefined and should be set by the user with the SD command. An example of the use of DDT is the computation of reservoir inflow from outflow and the change in storage, where the change in storage is transformed to average flow volume per day by the function. The conversion factor for storage to flow is 0.50416 when the storage change is in $ac\text{-}ft$ day and 1.1574×10^{-5} when it is in m^3 day.

Name: $DECPAR$ Decaying basin wetness parameter

Use: $CO\ TY=DECPAR(TY,TZ,R)$

Description: Compute a time-series of parameters TY as a function of TZ and R :

$$TY(t)=R*TY(t-1)+TZ(t)$$

where R is the decay rate and TZ is precipitation. R is less than 1. The function extends TY : the first value in the series TY is used as the starting value, and any other TY are computed in the sequence. If the first value in TY is undefined, it is assumed to be zero. The first TZ value is ignored. TY and TZ must be regular-interval time series with identical times. R should be appropriate for the interval. The type and units of TY are unchanged. An IF condition has no effect.

Name: $DIFF$ Successive differences

Use: $CO\ TY=DIFF(TX)$

Description: Compute the successive differences of TX and store in TY . TX and TY cannot be the same variable. TX must be of type $INST\text{-}VAL$ or $INST\text{-}CUM$. If a value of TX is undefined, resulting values of TY are undefined. If a concurrent IF condition is not satisfied, TY is unchanged. The type and units of TY are undefined and should be set using the SD command..

Name: ESTLIN Estimate values for missing data

Use: CO TY=ESTLIN(TX,NMAX,QFLAG)

Description: Linearly interpolate estimates for values in TX with flags equal or lesser in quality to QFLAG and place the results in TY. Do not estimate more than NMAX continuous missing values. Possible flags, in order of decreasing quality, are: Q = questionable and M = missing. An IF condition has no effect. TX must be a time-series, and TX and TY may be the same variable. Type and units of TY are the same as TX.

Name: ESTPPT Estimate values for missing precipitation data

Use: CO TY=ESTPPT(TX,NMAX,QFLAG)

Description: Linearly interpolate estimates for cumulative precipitation values in TX with quality flags equal to or lesser in quality to QFLAG and place the results in TY. Possible flags, in order of decreasing quality, are: Q = questionable and M = missing. If the values bracketing the missing period are increasing with time, do not estimate more than NMAX continuous missing values. If the values bracketing the missing period are equal, then estimate any number of missing values. If the values bracketing the missing period are decreasing with time, do not estimate any missing values. TX must be a data type of INST-CUM. An IF condition has no effect. The units and type of TY are set equal to the units and type of TX.

Name: FMA Forward moving average

Use: CO TY=FMA(TX,NP)

Description: Compute a moving average of the last NP values in TX and store in TY. NP must be greater than 2. NP values at the beginning of TY will be missing. If a value in TX is missing, the value is not used for computing TY, and the average is over one less value. At least 2 values of TX must be defined, else TY is missing. Useful for computing flow durations. Also, may be used to determine parameters for use in the SCR2 screening function. The units and type of TY are set equal to the units and type of TX. An IF condition has no effect.

Name: GENTSR Generate a regular interval time series

Use: CO TY=GENTSR(DT,TOFF,Y0,QFLAG)

Description: Generate a new, regular-interval time series TY with a time interval of DT, a time interval offset of TOFF, a constant value Y0, and all values internally flagged with QFLAG. TOFF is time from the beginning of the standard interval to the actual time of the data. DT and TOFF are expressed as units of time as nnT, where T is M for minutes, H for hours, and D for days. Possible flags are: N = none, M = missing, E = estimated, and Q = questionable. If Y0 is -901., then the flag is automatically M. Units and type must be set independently using the SD command. An IF condition has no effect.

Name: INT Truncate to whole numbers

Use: CO TY=INT(TX)

Description: Truncate to a whole number TX and store in TY. TX and TY can be the same variable. TX can be scalar, time series or paired data. If a value of TX is undefined, resulting values of TY are undefined. If a concurrent IF condition is not satisfied, TY is unchanged. The units and type of TY are set equal to the units and type of TX.

Name: LAST Last valid value in a time series

Use: CO SY=LAST(TX)

Description: Find the last valid value in time-series TX and place it in scalar SY. Ignores missing values or values concurrent with an unsatisfied IF condition.

Name: LOG Natural log base "e"

Use: CO TY=LOG(TX)

Description: Compute the natural log base e of TX and store in TY. TX and TY can be the same variable. TX can be scalar, time series or paired data. If a value of TX is undefined resulting values of TY are undefined. If a concurrent IF condition is not satisfied, TY is unchanged. If a value in TX is 0 or negative, the value in TY will be set to missing. The units and type of TY are set equal to the units and type of TX.

Name: LOG10 Log base 10

Use: CO TY=LOG10(TX)

Description: Compute the log base 10 of TX and store in TY. TX and TY can be the same variable. TX can be scalar, time series or paired data. If a value of TX is undefined, resulting values of TY are undefined. If a concurrent IF condition is not satisfied, TY is unchanged. If a value in TX is 0 or negative, the value in TY will be set to missing. The units and type of TY are set equal to the units and type of TX.

Name: MATE Generate data pairs from two time-series

Use: CO PF=MATE(TX,TY,SORT/NOSORT)

Description: Derive a paired-function PF by pairing values in the time series TX and TY. TX and TY values must have identical times. (Functions TS1 and TS4 may be useful in conjunction with MATE). The paired data are sorted into ascending order of TX if SORT is specified. The units of PF are respectively the same as those of TX and TY. The types and label of PF are undefined and must be set with the SD command. An IF condition has no effect.

Name: MAX Maximum value in a time series

Use: CO SY=MAX(TX)

Description: Find the maximum value in time-series TX and place it in scalar SY. Ignores missing values or values concurrent with an unsatisfied IF condition.

Name: MEAN Mean value in a time series

Use: CO SY=MEAN(TX)

Description: Compute the mean value in time-series TX and place the result in scalar SY. Ignores missing values or values concurrent with an unsatisfied IF condition.

Name: MIN Minimum value in a time series

Use: CO SY=MIN(TX)

Description: Find the minimum value in time-series TX and place the result in scalar SY. Ignores missing values or values concurrent with an unsatisfied IF condition.

Name: MREG Multiple linear regression function

Use: CO PF=MREG(TY, TX1, TX2, ..., TXn, min, max)

Description: This function is used to determine the coefficients (B_n) of a linear regression equation of the general form ($Y = B_0 + B_1 \cdot X_1 + B_2 \cdot X_2 + B_n \cdot X_n$). Where Y is the dependent variable and the X1, X2, and Xn are independent variables. The MREG function requires that the function parameters TY, TX1, TX2, and TXn be regular time series variables and that they all have the same time interval. It is further required that the time variables be retrieved in the same sequential order as specified in the function parameter list. In other words TY is retrieved from DSS first followed by TX1 and so on. The number of independent time series variables that can be specified is limited to one less the maximum number of time series variables that the program allows. On the DOS PC this value will be 4 independent variables. The calculated linear regression coefficients will be stored in a paired function variable specified by the user (PF). The user should store this variable to DSS with the PUT command if it is to be used later with the AMREG function. Two optional function parameters "min, max" are available to specify a range of values that control which values in the dependent variable are to be used for calculating the regression coefficients. These two parameters must be specified as the last two parameters and if specified, both must be specified as real numbers. The statistics output is written to file "MREG.REP" in addition to being printed out in the regular output.

Name: MRG Merge two time series

Use: CO TY=MRG(TX, TZ, QFLAG)

Description: Merge TX with TZ. TY includes all values in TX and TZ, except where TX and TZ occur at the same time. TX and TY cannot be the same variable. In that case, the value for TX is used unless it is flagged with a quality equal to or less than QFLAG and TZ is flagged with a quality greater than QFLAG. Possible flags, in order of decreasing quality, are:

N = no flag, E = estimated, Q = questionable and M = missing. An IF condition has no effect. The type and units of TY are undefined and must be set by the the SD command.

Name: MRGP Merge two paired data series

Use: CO PY=MRGP(PX,PZ,NCURVE)

Description: Merge PZ with PX. PY includes all values in PX and the curve specified by the index curve number NCURVE in PZ. It is required that PX and PZ have the same number of data values and that the X axis values for both PX and PZ are identical. NCURVE is used to specify the curve number data in PZ that is to be merged with PX. A value of 0 for NCURVE signifies that all curves in PZ are to be merged with PX.

Name: MUSK Muskingum hydrologic routing

Use: CO TY=MUSK(TX,NR,K,X)

Description: Route the uniform time series variable TX by the Muskingum hydrologic routing method and store it in variable TY. TX and TY can be the same variable. The "IF" compute options has no effect on this function. The "K" parameter is the Muskingum "k" in hours, "X" parameter is the Muskingum "x" (range between 0 and .5), and "NR" is the number of routing subreaches. The units and type of TY are set equal to the units and type of TX.

Name: NINT Round to nearest whole number

Use: CO TY=NINT(TX)

Description: Round to nearest whole number TX and store in TY. TX and TY can be the same variable. TX can be scalar, time series or paired data. If a value of TX is undefined resulting values of TY are undefined. If a concurrent IF condition is not satisfied, TY is unchanged. The units and type of TY are set equal to the units and type of TX.

Name: OLY Olympic smoothing

Use: CO TY=OLY(TX,NP[,CLEVEL])

Description: Compute a smoothed time-series from TX using the Olympic smoothing

scheme: same as centered, moving average except the minimum and maximum values in the span NP are ignored. Place the result in TY. The units and type of TY are set equal to the units and type of TX. Variables TY and TX cannot be the same variable. The units and type of TY are set equal to the units and type of TX. The following levels are available:

LEVEL1 Only valid values will be averaged. Valid values that do not have all valid values within the averaging period number will be missing. Values at the start and end of the data that do not have enough valid values to average will be assigned missing values.

LEVEL2 Only valid values will be averaged. Valid values that do not have all valid values within the averaging period number will be missing. Valid values at the start and end of the data that do not have enough valid values to average will be averaged over a reduced number of values.

LEVEL3 (Default setting) All values will be averaged based on valid values within the averaging period number. If all values are missing for the period number, the value will be missing. Values at the start and end of the data that do not have enough valid values to average will be assigned missing values.

LEVEL4 All values will be averaged based on valid values within the averaging period number. If all values are missing for the period number, the value will be missing. Values at the start and end of the data that do not have valid values to average will be averaged based on a reduced number of values.

Note: Questionable and estimated flagged values are used in the computations.

Name: PERCON Period constants

Use: CO TY=PERCON(TS,TX)

Description: Generates a series TY at times concurrent with TX and with values equal to the previous chronological TS value. Variables TX and TY cannot be the same variable. If no previous TS is present, then TY values are undefined. TS and TX may be irregular- or regular-interval time series. An IF condition has no effect. The units and type of TY are undefined and should be set with the SD command.

Name: POLY Polynomial transformation

Use: CO TY=POLY(TX,TP)

Description: Compute a polynomial transformation of TX using the polynomial coefficients TP. Store the result in TY. Variables TX and TY cannot be the same variable. If a TX value is missing TY is undefined. If a concurrent IF condition is not satisfied, TY is unchanged. Units and type of TY are defined by TP. TP can be created with the utility DSSPD.

Name: POLY2 Polynomial transformation with integral

Use: CO TY=POLY2(TX,TP)

Description: Compute a polynomial transformation of TX using the integral of the polynomial defined by coefficients TP. Variables TX and TY cannot be the same variable. Store the result in TY. If a TX value is missing TY is undefined. If a concurrent IF condition is not satisfied, TY is unchanged. Units and type are defined by TP. TP can be created with the utility DSSPD.

Name: PULS Modified Puls or Working R&D routing function

Use: CO TY=PULS(TX,PF,X,NR,STOR1,Q01)

Description: Route the uniform time series variable TX by the Modified Puls or Working R&D hydrologic routing method and store it in variable TY. Variables TY and TX may be the same variable. The variable PF must have been previously defined in a GET command. It references a storage-discharge paired function relationship for the reach. Storage must be the first variable, discharge the second variable, and each variable must repeat only once. Variable X is the wedge coefficient (Muskingum X) for use in working R&D. Use 0.0 value to route by Modified Puls method. Variable NR is the number of routing reaches. STOR1 and Q01 are initial storage and flow respectively. Use a value of -1 for both STOR1 and Q01 in order to use the first flow value in TX and interpolate a corresponding storage value from PF. The IF compute command has no effect on this function. The units and type of TY are set equal to the units and type of TX.

Name: QAC Flow accumulator gage processor

Use: CO TY=QAC(TX,TC)

Description: Compute period-average flows from a flow accumulator type gage:

$$TY(t) = (TX(t)-TX(t-1))/(TC(t)-TC(t-1))$$

TX and TC are respectively, time series of the accumulated flow and the count. TX and TC values must occur at the same times. Variables TY and TX cannot be the same variable. If corresponding values for TX and TC decrease from the previous period, then the accumulation is assumed to have reset to zero at the beginning of the interval. An IF condition has no effect. TY is assigned the type 'PER-AVER' and the units are set equal to the units of TX.

Name: RND Round off

Use: CO TY=RND(TX,NDIG,IPLAC)

Description: Round off values to NDIG or IPLAC, whichever controls. NDIG is the number of digits to round to and can range from 1 to 8. IPLAC is a magnitude of 10 to which to round to: for example, -1 specifies rounding to one-tenth (0.1). The number of digits shown is never less than 1, however. The following example illustrates the effects on rounding if NDIG=3 and IPLACE=0 (ie., round to ones place):

	Rounds to	
1445.1	"	1450.
144.51	"	145.
14.451	"	14.
1.4451	"	1.0

If TX is undefined, TY is undefined. If a concurrent IF condition is unsatisfied, TY is unchanged. Variable TY and TX can be the same variable.

Name: RTABLE Rating table interpolation

Use: CO TY=RTABLE(TX,TB)

Description: Interpolate values for TX using table TB and store the result in TY. Variables TY and TX cannot be the same variable. TB must be created using the program DSSPD (use /R option) with specific information in the header: type of interpolation, offset, shift, and datum. Program DSSUTL can be used to specify or modify the paired data header information. If the type of interpolation is LOGLOG, table x values are adjusted by subtracting the offset. The shift is added to and the datum subtracted from all incoming TX values. The header information in TB is used to define the units of TY. The type of TY should be set using the SD command. An IF statement has no effect on computation.

Name: RTABLR Reverse rating table interpolation

Use: CO TY=RTABLR(TX,TB)

Description: Interpolate values for TX using the reverse of table TB. Variables TY and TX cannot be the same variable. TB must be created using the program DSSPD (option /R) with specific information in the header: type of interpolation, offset, shift, and datum. Program DSSUTL can be used to specify or modify the paired data header information. If the type of interpolation is LOGLOG, table x values are adjusted by subtracting the offset. The shift is subtracted from and the datum added to all resulting TX values. The header information in TB also defines the units of TY. The type of TY should be set using the SD command. An IF statement has no effect on computation.

Name: RTABL2 Two-variable rating table interpolation

Use: CO TY=RTABL2(TX,TZ,TB)

Description: TY is a function of two independent variables TX and TZ. Variable TY and TX cannot be the same variable. The functional relationship is specified by the table TB, which consists of sets of TX/TY pairs, each set corresponding to a TZ value. TB is created with the program DSSPD (option /R) with values for TZ specified as labels for the sets of TX/TY pairs. RTABL2 interpolates linearly in table TB. No extrapolation is done: if the TX or TY value are outside the range bounded by TB, TY is set to missing. TX and TZ must be concurrent time series. The header information in TB also defines the units of TY. The units and type of TY must be set using the SD command. An IF statement has no effect on computation.

Name: SCRN1 Screen for possible erroneous values based on maximum/minimum range

Use: CO TY=SCRN1(TX,XMIN,XMAX,MAXDEL,QFLAG)

Description: Flag any value in TX that falls outside the range XMIN - XMAX or exceeds the maximum change MAXDEL from the previous value. Variables TY and TX can be the same variable. The maximum change comparison is done only when the consecutive values are not flagged. Possible values for QFLAG are: M = missing data or Q = questionable. The result is placed in TY. The units and type of TY are set equal to the units and type of TX. An IF statement has no effect on computation.

Name: SCRN2 Screen for possible erroneous values based on forward moving average maximum

Use: CO TY=SCRN2(TX,NPTS,MAXDEL,QFLAG)

Description: Flag any value in TX that exceeds the maximum change MAXDEL from the forward moving average of NPTS ending at the previous value. Missing values in TX are not counted in the moving average and the divisor of the average is less one for every missing value. Values which fail the screen are not counted either. At least 2 values must be defined else the moving average is undefined and the screen passes the relevant TX value. Possible flags are: M = missing data or Q = questionable. Variables TY and TX cannot be the same variable. The result is placed in TY. SCRN2 is useful for detecting and removing spikes. The FMA function may be useful for determining appropriate NPTS and MAXDEL parameters. The units and type of TY are set equal to the units and type of TX. An IF statement has no effect on computation.

Name: SDEV Compute the standard deviation of one independent variable.

Use: CO SY=SDEV(TX)

Description: This function is used to compute the standard deviation of one independent variable. TX can be either regular or irregular time series. SY is a scalar and is set to the computed standard deviation. This function requires a minimum of three valid values to compute.

Name: SELECT Extract time series data at unique time specification.

Use: CO TS= SELECT(TX,LEVEL,RANGE,FLAG,TWINDO,TFLAG)

Description: This function is used to extract data from any regular or irregular time series, based on any number of unique time specifications. The extraction process can take place on any of five different selection LEVELs which can be mutually inclusive or exclusive as defined by the FLAG and RANGE parameters. Each LEVEL has a RANGE parameter that is either a specific value or a range of values. The extracted data is defaulted to be irregular time series, but the user has the option to redefine it as regular by the TFLAG parameter. CAUTION: Defining the extracted data to be regular will cause it to be stored in a consecutive manner at regular time interval specified on the E pathname part without regard to its actual irregular date and time. Variables TS and TX cannot be the same variable. If a concurrent IF condition is not satisfied, TX is not selected. The units and type of TS are set equal to the units and type of TX.

<u>Parameter</u>	<u>Description</u>
TX	Input variable name of regular or irregular time series.
LEVEL	Level of time specific selection.
RANGE	Beginning and ending values of LEVEL. If only one value is specified, then the beginning and ending values are assumed to be the same. The following table shows all the valid LEVEL and RANGE values.

LEVEL	RANGE	Example
YEAR	Numeric: Four digit year value or a range of values. #### or ####-####	1938 or 1938-1945
MONTH	Alpha: First three characters of a month or a range of months. aaa or aaa-aaa	JAN or OCT-FEB
DAYMON	Numeric/Alpha: One or two digit day value or a range of values or the key word "LASTDAY" which specifies the last day of the month.	15 or 1-15 or LASTDAY or 15-LASTDAY

DAYWEE	Alpha: First three characters of a week day or a range of week days. aaa or aaa-aaa Note: Sunday is day 1 and Saturday is 7.	MON or SUN-SAT
TIME	Numeric: Four digit military style 24 hour clock consisting of a single time or a range of time. #### or ####-####	2300 or 0300-0600

Parameter

Description

FLAG The valid entries for the processing FLAG are "INCLUDE" (default) or "EXCLUDE". This entry controls the inclusion or exclusion of all data specified by LEVEL and RANGE.

TWINDO This parameter is only used with the TIME component of the LEVEL parameter and is expressed in minutes before and after the time of day within which the data will be extracted. The window is assumed to be symmetric about the time specified in RANGE. At the time boundaries of 0000 and 2400, the time window will not cross into the previous or next day.

TFLAG The valid entries for the TFLAG are "IRREGULAR" (default) or "REGULAR". This parameter allows the user to control how DSSMATH treats the data extracted as it relates to its internal definition of the type of time series data it is. This allows the user to override the protection in the program to write irregular data as regular.

Name: SHIF S h i f t a d j u s t m e n t

Use: CO TY=SHIFT(TS, TX)

Description: Generate a time series of shift adjustments TY with times at TX and actual shifts TS, possibly at other times. TX and TY cannot be the same variable. TS values are interpolated at TX times. The interpolation is linear between TS values, except when TX time is greater than last TS time. Then, the last value of TS is held constant for the remaining TX times. If no previous TS value brackets TY, then TY is set to zero. An IF condition has no effect. The units and type of TY are set equal to the units and type of TS.

Name: SIN Sine trigonometric function

Use: CO TY=SIN(TX)

Description: Compute the sine of TX in radians and store in TY. TX and TY can be the same variable. TX can be scalar, time series or paired data. If a value of TX is undefined resulting values of TY are undefined. If a concurrent IF condition is not satisfied, TY is unchanged. The units and type of TY are set equal to the units and type of TX.

Name: SKEW Compute the skew coefficient of one independent variable

Use: CO SY=SKEW(TX)

Description: This function is used to compute the skew coefficient of one independent variable. TX can be either regular or irregular time series. SY is a scalar and is set to the computed skew coefficient. This function requires a minimum of three valid values to compute.

Name: SQRT Square root function

Use: CO TY=SQRT(TX)

Description: Compute the square root of TX and store in TY. TX and TY can be the same variable. TX can be scalar, time series or paired data. If a value of TX is undefined, resulting values of TY are undefined. If a concurrent IF condition is not satisfied, TY is unchanged. If a value in TX is negative, the value in TY will be set to missing. The units and type of TY are set equal to the units and type of TX.

Name: SS Straddle Stagger routing function

Use: CO TY=SS(TX,NAVG,LAG,NR)

Description: Route the uniform time series variable TX by the Straddle Stagger hydrologic routing method and store it in variable TY. Note: Variables TY and TX cannot be the same variable and variable TX is permanently modified by being lagged by the SS function. Variable NAVG is the number of ordinates to average. Variable LAG is the number of ordinates to lag hydrograph. Variable NR is the number of subreaches to use. The "IF" compute options has no effect on this function. The units and type of TY are set equal to the units and type of TX.

Name: SSW Wilmington District Straddle Stagger routing function

Use: CO TY=SSW(TX,NAVG,LAG,NR)

Description: Route the uniform time series variable TX by the Straddle Stagger hydrologic routing method and store it in variable TY. Note: Variables TY and TX cannot be the same variable. This function is similar to function SS except that LAG is usually zero and the result of averaging NAVG values is stored in the NAVGed value. Variable NAVG is the number of ordinates to average. Variable LAG is the number of ordinates to lag hydrograph. Variable NR is the number of subreaches to use. For example, if NAVG is 7, the results for ordinates 13 through 19 is stored in ordinate 19. For the same example, if LAG is set to 2, the result is stored in ordinate 21. The "IF" compute option has no effect on this function. The units and type of TY are set equal to the units and type of TX.

Name: TAN Tangent trigonometric function

Use: CO TY=TAN(TX)

Description: Compute the tangent of TX in radians and store in TY. TX and TY can be the same variable. TX can be scalar, time series or paired data. If a value of TX is undefined resulting values of TY are undefined. If a concurrent IF condition is not satisfied, TY is unchanged. The units and type of TY are set equal to the units and type of TX.

Name: TS1 Interpolated data at regular intervals

Use: CO TY=TS1(TX,DT,TOFF)

Description: Derive a new, regular-interval time series TY from TX. TY will have a time span defined by the current time window, a time interval of DT, and a time interval offset of TOFF. TOFF is time from the beginning of the standard interval to the actual time of the data. DT and TOFF are expressed as units of time as nnT, where T is M for minutes, H for hours, and D for days. TX may be a regular or irregular time series. Units and type of TY are set equal to the units and type of TX. TY and TX can not be the same variable. An IF condition has no effect.

Name: TS2 *Period averages at regular intervals*

Use: CO TY=TS2(TX,DT,TOFF)

Description: *Derive a new, regular-interval time-series TY from the instantaneous values in TX with a time span defined by the current time window, a time interval of DT, and an interval offset of TOFF. DT and TOFF are expressed as units of time as nnT, where T is M for minutes, H for hours, and D for days. TOFF is time from the beginning of the standard interval to the actual time of the data. TX may be an irregular or regular time series, but must be of type INST-VAL. TY will be typed PER-AVER. Example of use: deriving daily average flow from hourly observed values. TY and TX cannot be the same variable. An IF condition has no effect.*

Name: TS3 *Period mins or maxs at regular intervals*

Use: CO TY=TS3(TX,DT,TOFF,EXT)

Description: *Finds either the minima or maxima in TX at DT intervals, with interval offsets at TOFF, and puts the result in TY. EXT is used to indicate the extreme of interest: 'MIN' or 'MAX'. DT and TOFF are expressed as units of time as nnT, where T is M for minutes, H for hours, and D for days. TOFF is time from the beginning of the standard interval to the actual time of the data. TY is typed PER-EXTR. Example of use: finding daily minima and maxima from hourly instantaneous observations. TY and TX cannot be the same variable. An IF condition has no effect.*

Name: TS4 *Interpolated data at irregular intervals*

Use: CO TY=TS4(TX,TZ)

Description: *Derive a new, irregular-interval time series TY from TX at the times for TZ (ie., TZ provides the time pattern). Units and type are preserved for TY based on TX. TY and TX cannot be the same variable. An IF condition has no effect.*

Name: TSCYCL *Time series cyclic analysis*

Use: CO XX=TSCYCL(TX,TREAT,STATFILE,DSSFILE)

Description: *Derive a set of statistics from cyclic regular time series TX. TX must be regular data at a 1HOUR, 1DAY, or 1MONTH interval. The time series TX may be, for example, 30 years of 1DAY data. For each interval,*

(e.g., 1st day of the year, 2nd day of the year, ... , 365th day of the year), statistics would be determined. The 14 statistics determined for each interval are: maximum, minimum, average, standard deviation, 5%, 10%, 25%, 50% (median), 75%, 90%, 95% percentiles, date of maximum, date of minimum, and number of values processed. These results are written to a specified DSS file, and two text output files. The variable XX is a dummy and is ignored. An IF condition has no effect.

TREAT controls the treatment of newly generated statistic values when missing data occur in the interval. It must be expressed in either the form nn# or nn%. If nn# is used nn gives the number of missing data values in the interval that will be accepted in the computation of the new value. If nn% is used nn gives the percent of missing data values in the interval that will be accepted in the computation of the new value. If the criteria is met a data value will be generated, if not a missing value will occur.

STATFILE is the base name, not more than 6 characters, of two files that will be written containing the same data written to the DSS file. The first file will contain all of the statistics generated. It will be in the form of a wide table. Its name will be 'statfile.sts' or 'statfile.t' where 'statfile' is the portion of the name the user specifies. This file is wide, so it is hard to view on some systems. The second file is a subset of the first named 'statfile.sum', or 'statfile.s'. The second file is limited to 80 columns wide for easy viewing. The second file contains: maximums with their dates, minimums with their dates, average, and the 50% percentile(median) values.

DSSFILE is the DSS file into which the cyclic results will be written. The results will be, for example, a record of all the maximums for each day of the year over the 30 year period. Fourteen records will be added to the DSS file specified, each containing one of the derived statistics. The records will use the pseudo year 3000 for storing the statistics

Name: TSHIFT Shift time series in time

Use: CO TY=TSHIFT(TX,DT)

Description: Derive a time series TY by shifting times in TX by DT. DT is specified in the form nnT, where nn is the number of time units T and T may be M for minutes, H for hours, or D for days. Units and type of TY are set equal to the units and type of TX. An IF condition has no effect.

Name: TSNAP Snap Irregular times to nearest Regular period

Use: CO TY=TSNAP(TX,DT,TOFF,TBACK,TFORWARD)

Description: Derive a regular time series from existing time series TX. TX may be regular or irregular. The new time series will be at a time interval of DT, interval offset of TOFF, backward time window of TBACK, and forward time window of TFORWARD. Parameters: DT, TOFF, TBACK, and TFORWARD are expressed as units of time "nnT", where "nn" is a number and "T" is MIN (minutes), H (hours), D (days), W (weeks), MON(months), or Y (years). Units and type of TY are set equal to the units and type of TX. An IF condition has no effect.

Name: TTSR Transform time series to regular

Use: CO TY=TTSR(TX,DT,TOFF,FUNCT,TREAT,TYPE)

Description: Derive a new regular time series from existing time series TX. Variables TY and TX cannot be the same variable. An IF condition has no effect. TX may be regular or irregular. The new time series will be at a time interval of DT, and an interval offset of TOFF. DT and TOFF must each be expressed in the form nnT, where T may be MIN, HOUR, DAY, WEEK, TRI, MON or YEAR (e.g. 1DAY). The units and type of TY must be defined by the use of the SD command. FUNCT is one of the following:

- INT - Interpolation at end of interval
- MAX - Maximum over interval
- MIN - Minimum over interval
- AVE - Average over interval
- ACC - Accumulation over interval
- ITG - Integration over interval
- NUM - Number of valid data over interval

TREAT controls the treatment of the new generated data value when missing data occur in the interval. It must be expressed in either the form nn# or nn%. If nn# is used nn gives the number of missing data values in the interval that will be accepted in the computation of the new value. If nn% is used nn gives the percent of missing data values in the interval that will be accepted in the computation of the new value. If the criteria is met a data value will be generated, if not a missing value will occur.

TYPE provides the user control to override how the interpolation and processing of data occurs. TYPE must be one of the following: DEFAULT, INST-VAL, PER-AVER, or PER-CUM. If DEFAULT is used, processing depends on the data type stored in the DSS data record. Otherwise processing will be performed as if the data type were as

given by TYPE. Data type INST-VAL considers the data to change linearly from the previous data value to the current data value over the interval. Data type PER-AVER considers the data to be constant at the current data value over the interval. Data type PER-CUM considers the data to increase from zero (0.0) up to the current value over the interval. Note: It is required that valid values exist at the start of the data being used.

Name: TTSI Transform time series to irregular

Use: CO TY=TTSI(TX,TZ,FUNCT,TREAT,TYPE)

Description: Derive a new irregular time series from existing time series TX. TX may be regular or irregular. TZ is an existing time series at the desired new spacing of TY. All other parameters and restrictions are as shown for the function TTSR.

Name: 1/X Inverse function

Use: CO TY=1/X(TX)

Description: Divide 1 by variable TX and store in TY. TX and TY can be the same variable. TX can be scalar, time series or paired data. If a value of TX is undefined, resulting values of TY are undefined. If a concurrent IF condition is not satisfied, TY is unchanged. If a value in TX is zero, the value in TY will be set to missing. The units and type of TY are set equal to the units and type of TX.