SET

SET *scalar* = *algebraic expression* ;
or
subscripted variable = *algebraic expression* ;
or
SET *frml name new variable name* ;

Function:

SET performs computations on scalar variables and single elements of time series or matrices.

Usage:

SET consists of the name of a scalar variable or an element of a series or matrix followed by an equal (=) sign and an arbitrary algebraic expression. The expression must follow the usual TSP rules for formulas.

The formula is evaluated using the current values of all the variables and the result is stored in the variable on the left hand side of the equation. If the left hand side is an element in a series and matrix, only that particular element is changed; the remainder of the variable is unchanged (whether or not the REPL mode is on).

Legal forms of scalar variables in TSP are the following (these forms can be used wherever scalars are allowed):

1. A simple variable name such as BETA, or D1. This could be already defined by a CONST or PARAM, but this is not required.

2. A subscripted series, such as GNP(I), GNP(72:1), GNP(72) or YOUNG(40). If the frequency is NONE, you can use a simple subscript in the same units as your sample. If the frequency is QUARTERLY (4), or MONTHLY (12), use a valid TSP date as the subscript. A variable (4 characters or less) can also be used on dated and undated series. Since variables do not take date values (except for annual frequency), a variable subscript is always relative to the start of the current SMPL. For example, the following loop fills the series X throughout the current SMPL:

```
SMPL 48:1,86:2;
DO I=1,@NOB;
   SET X(I) = ... ;
ENDDO;
```

3. A subscripted matrix. A matrix may have a single numeric or variable subscript, which is computed by the following formula:

   subscript = (j-1)*NROW + i

   where i is the row number of the element and j is the column number.

Matrices may also be doubly subscripted, but any variable subscripts must be 2 characters or less. Here are some examples of legal matrix elements:

XL(I,J) XL(2,6) MAT(345) MAT(I) XL(II,247) A(K1,LL)

SET

These are illegal:

```
MAT(VARSUB)   XL(I20,2)    A(K,L+1)   GNP(I-1)
```

The first and second are illegal because subscript names are limited to 4 or 2 characters. The third and fourth are illegal because expressions are not allowed as subscripts.

SET is not recommended for creating a series or matrix. READ, GENR, TREND, DUMMY, etc. should be used to create series. READ, MFORM, MMAKE, or COPY should be used to create matrices. SET can be used to update series and matrices, or to retrieve particular elements from them.

Examples:

```
SET VALUE = X(1,1) ;
SET SE = @S ;
MATRIX(2,3) = A+B**2 / (LOG(LABW)) ;

FREQ A;
SMPL 1983 1990 ;
GENR PFOR = 100 ;
DO I = 1984 TO 1990 ;
    SET I1 = I-1 ;
    SET PFOR(I) = PFOR(I1)*EXP(1.0+DELTA) ;
ENDD ;
```

This last example shows how SET can be used to compute a series in which each observation is a dynamic function of a previous observation. This can be done more efficiently, however, with a dynamic GENR:

```
FREQ A;
SMPL 83,90;
PFOR = 100;
SMPL 84,90;
PFOR = PFOR(-1)*EXP(1+DELTA);
```

Output:

SET produces no printed output. A scalar is stored in data storage, or a series or matrix is updated.