

:=

There are three possible uses of the delimiter  
:= as follows:

- (i) in a switch declaration, e.g.  
switch S := .....
- (ii) in a for clause, e.g.  
for v := 1 step .....
- (iii) in an assignment statement.

The first two are dealt with under switch and for respectively, and this routine merely deals with assignment statements. After checking that we are not assigning to a constant or an expression and that we are not in a declaration, for clause or procedure call, DECSTA and SV set to /o o and TAKE deals with the variable (simple or subscripted) which precedes the delimiter.

As we require all left hand elements in an assignment to be the same type the top of the stack is then inspected to see if it is a :=. If not LHTYPE is set to TYPBOX and the delimiter is stacked, otherwise LHTYPE and TYPBOX are tested for equality and the multiple store function (STA) is stacked.

#### ERRORS

FAIL 28 ; := preceded by constant or used inside an expression.

FAIL 52 ; := must not appear in actual parameter list, or in a type or array declaration.

FAIL 7 ; := appears in a for statement and other than in assignment to controlled variable.

FAIL 112; Different types on Left hand side of an assignment.