TAKE IDENTIFIER

This subroutine is used to find the namelist entry for the current identifier and to compile the correct object program Take Address or Take Value operation.

The subroutine SEARCH fails if the identifier is not found in the Namelist and leaves I pointing to the entry if found. An extra check is made that an identifier occurring in an array declaration is non local. The type of the identifier is then inspected to determine which object program operation should be compiled.

(i) <u>switch</u> e.g. <u>go</u> <u>to</u> S [n]

No operation is required until the switch subscript has been processed.

(ii) <u>label</u> e.g. <u>go</u> <u>to</u> LABEL

GTF or GT is stacked depending on whether the label is a formal parameter or not, together with the label address from the namelist. This will be compiled in UNSTAK at the end of the statement.

(iii) <u>array</u> e.g. a + arrayname [b]

Take Address or Take Formal Address is compiled with the address of the array map.

(iv) <u>type procedure</u>

e.g. FUN :=

A:= a + FUN

In the former case, this is an assignment to a function designator shown by P = 0. If FD = 3 this assignment is outside the procedure body; otherwise FD is set to 1 to show the assignment has been made. IFUN or RFUN is compiled depending on whether this is an integer procedure or real procedure with no parameters.

In the latter case, this is a function call shown by P = 1. An operation to reserve space for the function designator is compiled, followed by CFF or CF, depending on whether this procedure is a formal parameter or not, together with the object program address of the procedure block.

(v) <u>scalar</u> e.g. REAL := int + REAL

The relevant Take Address or Take Value operation is compiled depending on whether P = 0 or 1 and a further distinction is made when the variable is a formal parameter, which may be called by name or value (see TABLE 1).

TABLE 1

P	f[I]	v[I]	type	Object Program Operation	
0	0	-	Integer	Take Integer Address	TIA
0	0	-	Real	Take Real Address	TRA
1	0	-	Integer	Take Integer Value	TIR
1	0	-	Real	Take Real Value	TRR
0	1	0	Integer	Take Formal Address	TFA *
0	1	0	Real		
0	1	1	Integer	Take Stack Integer Address	IFUN
0	1	1	Real	Take Stack Real Address	RFUN
1	1	0	Integer	Take Formal Value Indirect	TRCN
1	1	0	Real	(Take Result Call by Name)	
1	1	1	Integer	Take Formal Value	TFV *
1	1	1	Real		

ERRORS

FAIL 41 ; indentifier in bound pair is local

FAIL 22 ; incorrect use of label, go to obscured or missing.

FAIL 111 ; type procedure zero declared with parameters.

FAIL 46 ; assignment to function designator is outside procedure body, or assignment to formal procedure

FAIL 31 ; assignment to switch

FAIL 25 ; non type procedure as function designator

* TFA and TFV are the same interpreter primitive.