COMPILER OPTIONAL OUTPUT

The following describes the meaning of the optional output from the Compiler, in particular the storage and label allocation.

(1) Procedure Identifiers

Identifier	Procedure Number	Enclosing Procedure Number	Procedure Type	Function Location
e.g.				
COSIN@	5	1	RP	3

The Identifier holds the first 8 characters of the procedure identifier (e.g. COSINE).

The Procedure Number specifies the USERCODE label number allocated to describe that procedure body (e.g. 5) and has a value between 2 and 95 inclusive.

The Enclosing Procedure Number specifies the procedure in which the procedure is declared (e.g. 1) and has a value between 1 and 95 inclusive;; 1 denoting the Main Program.

The Procedure Type specifies the kind of procedure declared as follows:-

BP - boolean
IP - integer
RP - real

If none of the above, 'P' is printed and the next column is left blank.

The Function Location specifies, for TYPE procedures, the stack location which is used to store the answer assigned to the function, relative to M1.

(2) Labels

Procedure Number	Block No.	Block Nesting Number	L	Identifier	USERCODE Reference

Appendix 5 (cont.)

(2) Labels

Procedure Number	Block No.	Block Nesting Number	L	Identifier	USERCODE Reference
e.g.					
5	6	2	L	COSFAIL	205

The Procedure Number specifies the USERCODE Label number allocated to the procedure in which the label is declared (e.g. 5) and has a value between 1 and 95 inclusive.

The Block Number is the block number counted sequentially from the start of the procedure in which the label is declared (e.g. 6).

The Block Nesting Number specifies the depth of the block in which the label is declared (e.g. 2).

'L' denotes a label description.

The Identifier holds the first 8 characters of the label identifier (e.g. COSFAIL).

(3) Begin/end

BEGIN 'C' /END 'C'	Block Number	Block Nesting Number
e.g.		
MAIN PROG		
BEGIN B	1	1
BEGIN P	2	1
END P	2	1
END B	1	1
ENDPROG		

Appendix 5 (cont.)

The BEGIN 'C' /END 'C' specifies the 'class' of BEGIN or END as follows:-

P denotes Procedure
F denotes a For Statement
B denotes an ordinary block

The Block Number is the block number counted sequentially from the start of the program.

The Block Nesting Number specifies the depths of the block containing the BEGIN or END.

Note. The first BEGIN (Block No. 1, Block Nesting No. 1) is preceded by MAINPROG and the last END (Block 1, Block Nesting No. 1) ENDPROG.

(4) The optional output given under the heading 'Storage Allocation and Reference Correspondence', by setting D4 of Z38, is given below.

Variables.

Marker 'O' 'N'	Type	Identifier	Allocated Address
e.g.			
N	R	A	2
O	IA	C	9
	RA	B	11

Marker 'O' 'N'. If the variable is an actual variable and is 'own', then O is printed, otherwise it is omitted.

Also, if the variable is formal, called by name, then 'N' is printed, otherwise it is omitted.

Type. This specifies the declared type as follows:-

A not a type array (i.e. implicitly real)
B boolean
I integer
R real
BA boolean array
IA integer array
RA real array
SW switch

The Identifier holds the first eight characters of the variable identifier.

The Allocated Address is the stack location given to the variable, and is relative to M1.