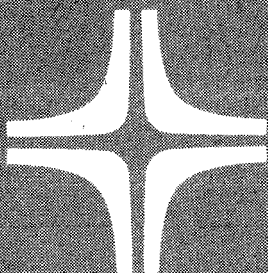


ESCORT

OS/3



Summary

This document contains the latest information available at the time of preparation. Therefore, it may contain descriptions of functions not implemented at manual distribution time. To ensure that you have the latest information regarding levels of implementation and functional availability, please consult the appropriate release documentation or contact your local Sperry Univac representative.

Sperry Univac reserves the right to modify or revise the content of this document. No contractual obligation by Sperry Univac regarding level, scope, or timing of functional implementation is either expressed or implied in this document. It is further understood that in consideration of the receipt or purchase of this document, the recipient or purchaser agrees not to reproduce or copy it by any means whatsoever, nor to permit such action by others, for any purpose without prior written permission from Sperry Univac.

Sperry Univac is a division of the Sperry Corporation.

FASTRAND, SPERRY UNIVAC, UNISCOPE, UNISERVO, and UNIVAC are registered trademarks of the Sperry Corporation. ESCORT, PAGEWRITER, PIXIE, and UNIS are additional trademarks of the Sperry Corporation.

This document was prepared by Systems Publications using the SPERRY UNIVAC UTS 400 Text Editor. It was printed and distributed by the Customer Information Distribution Center (CIDC), 555 Henderson Rd., King of Prussia, Pa., 19406.

Contents

STATEMENT FORMAT CONVENTIONS AND LANGUAGE RULES	1
RESERVED WORDS	2
ESCORT COMMANDS LISTING	3
PROGRAM STATEMENT FORMATS	4
Change Data Program Statement	4
Delete Data Program Statement	6
Enter Data Program Statement	7
Select Data Program Statement	10
Sort Data Program Statement	13
PROGRAM STATEMENT CLAUSE CROSS-REFERENCE	14
CLAUSE FORMATS	16
ARITHMETIC OPERATORS AND RULES	19
RELATIONAL OPERATORS	20
CHARACTER STRING OPERATORS	21
LOGICAL OPERATORS	22
SYSTEM UTILITY FIELDS	23
STRUCTURE DESIGNATIONS	25
EDIT CODES	26
FUNCTION KEY TABLE	28



The conventions and rules used to present statement formats are:

In the direct entry method, key in uppercase words exactly as shown. In the screen menu method, uppercase words designate words generated by ESCORT.

Lowercase words designate names, such as file or structure names, that you supply.

Brackets [] enclose optional parameters.

Braces { } indicate alternative parameter choices. When you use the parameter, you must pick one of the choices enclosed by braces.

Reserved words consist of all words used in ESCORT commands, statements, and clauses. You can't use these as variable field or file names, but you can use them in data you input or output or as alphanumeric constants.

ESCORT allows alphanumeric, numeric, and decimal constants.

ESCORT programs can be written in upper or lowercase, but when you're comparing or sorting alphanumeric data fields, lowercase words have greater value than their uppercase counterparts.

AND	DISPLAY	EXTENDING	NOT	STARTS	WORKAREA
BY	DOES	FIRSTIME	OF	SUBTOTALING	WS
CHANGE	ELSE	FROM	OR	SUBTOTALS	\$DATE
CLEARING	END	GE	OUTPUT	TO	\$DAY
COMPUTING	ENDOFILE	GT	PRINT	TOTALING	\$LINE
CONTAIN	ENDS	IF	PRINTER	TOTALS	\$MONTH
CONTAINS	ENTER	LE	SELECT	UPDATING	\$PAGE
CREATING	EQ	LT	SORT	USING	\$TIME
DATA	EXIT	NE	START	WHILE	\$TITLE
DELETE				WITH	\$YEAR

Program Commands	Structure Commands	Job Commands
CHANGE PROGRAM	CHANGE STRUCTURE	CHANGE JOB
DISPLAY PROGRAM	DISPLAY STRUCTURE	DISPLAY JOB
DISPLAY PROGRAM NAMES	DISPLAY STRUCTURE NAMES	DISPLAY JOB NAMES
ENTER PROGRAM	ENTER STRUCTURE	ENTER JOB
PRINT PROGRAM	PRINT STRUCTURE	PRINT JOB
PRINT PROGRAM NAMES	PRINT STRUCTURE NAMES	PRINT JOB NAMES
RUN PROGRAM	SAVE STRUCTURE	RUN JOB
SAVE PROGRAM		SAVE JOB

4 Change Data Program Statement

CHANGE DATA OF master-file [structure] $\left[\begin{array}{l} \text{FROM} \{ \text{WS} \\ \text{input-file} \} \left[\left\{ \begin{array}{l} \text{structure} \\ \text{structure,form} \end{array} \right\} \right] \end{array} \right]$

[USING input-file-1,...,input-file-5]

WORKAREA structure [data replacement statements]

SUBTOTALING fieldnames ["literal expressions"]

TOTALING fieldnames ["literal expressions"]

IF $\left\{ \begin{array}{l} \text{conditions} \\ \text{FIRSTIME} \\ \text{ENDOFIE} \end{array} \right\}$ clauses [ELSE clauses] END

COMPUTING data replacement statements

WHILE conditions clauses END

CLEARING fieldnames

PRINT { fieldnames ["literal expressions"]
SUBTOTALS
TOTALS
structure }

DISPLAY { fieldnames ["literal expressions"]
SUBTOTALS
TOTALS
structure }

OUTPUT TO { WS } { structure
PRINTER } { structure, form }

EXIT

9 Delete Data Program Statement

```
DELETE DATA OF master-file { FROM { WS { structure }  
                             { input-file } { structure, form } }  
                             [IF conditions] }
```

Enter Data Program Statement

ENTER DATA FROM WS { structure
 { structure, form } }

WORKAREA structure [data replacement statements]

SUBTOTALING fieldnames ["literal expressions"]

TOTALING fieldnames ["literal expressions"]

IF { conditions } clauses [ELSE clauses] END
 { FIRSTIME }
 { ENDOFILE }

∞ Enter Data Program Statement (cont)

COMPUTING data replacement statements

WHILE conditions clauses END

CLEARING fieldnames

CREATING output-file [structure]

EXTENDING output-file [structure]

PRINT { fieldnames ["literal expressions"]
SUBTOTALS
TOTALS
structure }

DISPLAY { fieldnames ["literal expressions"]
SUBTOTALS
TOTALS
structure }

OUTPUT TO { WS {structure}
PRINTER } {structure,form} }

EXIT

10 Select Data Program Statement

SELECT DATA OF master-file $\left[\text{FROM} \left\{ \begin{array}{l} \text{WS} \\ \text{input-file} \end{array} \right\} \left[\left\{ \begin{array}{l} \text{structure} \\ \text{structure, form} \end{array} \right\} \right] \right]$

[USING input-file-1,..., input-file-5]

WORKAREA structure [data replacement statements]

SUBTOTALING fieldnames ["literal expressions"]

TOTALING fieldnames ["literal expressions"]

IF $\left\{ \begin{array}{l} \text{conditions} \\ \text{FIRSTIME} \\ \text{ENDOFIE} \end{array} \right\}$ clauses [ELSE clauses] END

COMPUTING data replacement statements

WHILE conditions clauses END

CLEARING fieldnames

CREATING output-file [structure]

EXTENDING output-file [structure]

UPDATING output-file [structure]

OUTPUT TO { WS } { structure }
 { PRINTER } { structure, form }

12

Select Data Program Statement (cont)

PRINT { fieldnames ["literal expressions"]
SUBTOTALS
TOTALS
structure }

DISPLAY { fieldnames ["literal expressions"]
SUBTOTALS
TOTALS
structure }

EXIT

Sort Data Program Statement

```
SORT DATA OF master-file [structure] [BY field1,....,field6]  
      CREATING output-file [structure]
```

PROGRAM STATEMENT AND CLAUSE
CROSS-REFERENCE

Clauses	Enter Data	Change Data	Select Data	Sort Data	Delete Data
BY				X	
CLEARING	X	X	X		
COMPUTING	X	X	X		
CREATING	X		X	X	
DISPLAY	X	X	X		
EXIT	X	X	X		
EXTENDING	X		X		

PROGRAM STATEMENT AND CLAUSE
CROSS-REFERENCE (cont)

IF	X	X	X		X
OUTPUT TO	X	X	X		
PRINT	X	X	X		
SUBTOTALING	X	X	X		
TOTALING	X	X	X		
UPDATING			X		
USING		X	X		
WHILE	X	X	X		
WORKAREA	X	X	X		

Clause	Format
BY	BY field-1,..., field-6
CLEARING	CLEARING fieldnames
COMPUTING	COMPUTING data replacement statements
CREATING	CREATING output-file [structure]
DISPLAY	DISPLAY { <ul style="list-style-type: none"> fieldnames ["literal expressions"] SUBTOTALS TOTALS structure }

EXIT

EXTENDING

IF

OUTPUT TO

PRINT

EXIT

EXTENDING output-file [structure]

IF {conditions} clauses [ELSE clauses] END
 {FIRSTIME}
 {ENDOFILE}

OUTPUT TO {WS } {structure
 {PRINTER}} {structure,form }

PRINT { fieldnames ["literal expressions"] }
 { SUBTOTALS }
 { TOTALS }
 { structure }

Clause	Format
SUBTOTALING	SUBTOTALING fieldnames ["literal expressions"]
TOTALING	TOTALING fieldnames ["literal expressions"]
UPDATING	UPDATING input-file-n
USING	USING input-file-1,...,input-file-5
WHILE	WHILE conditions clauses END
WORKAREA	WORKAREA structure [data replacement statements]

Arithmetic Operators	
+	Addition
-	Subtraction
*	Multiplication
/	Division
:	Modulo (a division operation where the result is the remainder portion)

In arithmetic expressions, multiplication, division, and modulo are performed before addition and subtraction.

The sequence for performing operations of the same level is left to right.

Expressions enclosed within parentheses are evaluated first. When using nested parentheses, evaluation begins with the innermost set and works outward.

Relational Operators

Alphabetic	Symbolic	Meaning
EQ	=	Equal to
NE	<>	Not equal to
GT	>	Greater than
LT	<	Less than
GE	>=	Greater than or equal to
LE	<=	Less than or equal to

Character String Operators

STARTS WITH "string"
ENDS WITH "string"
CONTAINS "string"
DOES NOT START WITH "string"
DOES NOT END WITH "string"
DOES NOT CONTAIN "string"

In addition to the relational and special character string operators, there are two logical operators, OR and AND that are used to connect one or more conditionals.

ESCORT provides eight system utility fields you can specify in any ESCORT program. These fields are always prefixed by a dollar sign (\$).

Field	Length	Function
\$DATE	8A	Gives you the current system date (mm/dd/yy)
\$DAY	2N	Gives you the current system day
\$LINE	3N	Sets the system printer to a specified line number before printing
\$MONTH	2N	Gives you the current system month
\$PAGE	4N	Sets the system page number to a specific page number

Field	Length	Function
\$TIME	5A	Gives you the current system time (hh:mm)
\$TITLE	40A	Allows you to provide a title line for printed reports
\$YEAR	2N	Gives you the current system year

In the LENGTH/TYPE parameter, the following symbols are used to designate field type:

- A — alphanumeric
- N — numeric
- V — assumed decimal
- P — packed decimal
- B — binary

In the KEYS parameter, the letter K, followed by a number 1 through 6 is used to designate key values.

Symbol	Function
Z	Suppress leading zeros
*	Replace leading zeros with asterisks
,	Insert commas where applicable
—	Insert sign after least significant digit. If the number is: Positive — the sign inserted is a blank character Negative — the sign inserted is a minus symbol

/	<p>Insert slash character where applicable in date field.</p> <p>In four digit date field, one slash is inserted.</p> <p>In six digit date field, two slashes are inserted.</p>
*\$	<p>Fixed dollar sign is assumed when the asterisk is used in the first position of the field.</p>
\$	<p>Floating dollar sign is assumed when the asterisk is omitted.</p>

NOTE:

The slash, sign, comma codes, and the decimal point are characters that are added to the field and should be allowed for in your output.

Key	Structure Processor	Program Mode	Tutorial Mode	Run-time Processor
F1	End of input, return to master menu	Not used	Not used	Not used
F2	Cancel display output	Not used	Not used	Not used
F3	Not used	Cancel current screen and return to previous screen	Cancel current screen and return to previous screen	Not used

F4	Abort structure and return to menu	Terminate free-form input and return to previous menu	Not used	Terminate data input and return to caller
----	------------------------------------	---	----------	---

NOTE:

In the run-time F4 key description, caller refers to the ESCORT processor that called for a run-time function. The processor can be either program mode or tutorial mode. Structures do not use run-time functions.





