

M E R G E R O U T I N E

1.0 C O N T E N T S

	<u>Page</u>
1.0 CONTENTS	
2.0 INTRODUCTION	1
2.1 HARDWARE REQUIREMENTS	1
2.2 DATA	1
3.0 ROUTINE CALLS	2
4.0 MACRO INSTRUCTIONS	3
4.1 INITIATING MACRO	3
4.2 SELECT MACRO	3
4.3 TERMINATING MACRO	3
4.4 NOTES	4
5.0 ESTIMATED STORAGE REQUIREMENTS	5
6.0 ESTIMATED EXECUTION TIMES	5
	ILLUSTRATIONS
SAMPLE PROGRAM: TWO-WAY MERGE	6-8

This document is preliminary in nature and is intended as a vehicle for meeting immediate needs with regard to system familiarization and orientation. UNIVAC® Division of Sperry Rand Corporation reserves the right to change and/or modify such information contained herein as may be required by subsequent system developments.

2.0 I N T R O D U C T I O N

The 1050 MERGE generator is designed to merge 2 to 12 ordered input files and to produce one output file of data sequentially ordered in accordance with the specified merge keys. A merge process is necessary if the data sorted by the 1050 SORT exceeded the maximum, which is determined by the amount of data which can be contained on one single reel of a collation phase tape.

The insertion method has been applied for merging.

Each merge program has to be written by the programmer as any other program. By implementing procedure calls the MERGE subroutines will be incorporated in the user program which has to communicate with the MERGE coding by means of the macro instructions. This supplies the user with the maximum flexibility in his own program. His program also has to perform the appropriate calls for the TAPE1 routine, or any other tape routine controlling tape input and output.

2.1 HARDWARE REQUIREMENTS

When using the Operating System and TAPE1, the following hardware requirements apply:

- UNISERVO Tape Units available

- Type: as allowed in TAPE1.

- Number of tape units: as required by TAPE1.

- All tape units must be of the same type.

- Memory Size

- Minimum: 8K

- Maximum: 32K

2.2 DATA

- Record Size

- This must be fixed (see TAPE1).

- Key

- The MERGE key consists of 1 to 10 fields occupying the same position in every record.

- Each field consists of 1 to 16 contiguous characters.

3.0 R O U T I N E C A L L S

The following calls have to be implemented.

3.1	LABEL	OP'N	OPERANDS
	*STD		

This is for standard label definitions.

3.2	LABEL	OP'N	OPERANDS
	SELCT		$p_1, p_2, p_3, \dots, p_n$

This causes the generation of the MERGE subroutines according to the specified parameters where

p_1 = number of input files (2-12)

p_2 = number of fields in the key (1-10)

p_3 = type and sequence desired for field 1

p_n = type and sequence desired for field n-2 and may be

B for ascending binary

D for ascending decimal

DB for descending binary

DD for descending decimal

3.3	LABEL	OP'N	OPERANDS
	KEYn		p_1, p_2

This is for a key definition where

n = field number (1-10) for an n-field key

p_1 = length of field (1-16) in characters

p_2 = position number of rightmost character of the field in the record (relative to the number one)

3.4 These comprise appropriate TAPE1 and FILE1 calls and AREA directives.

4.0 M A C R O I N S T R U C T I O N S

4.1 INITIATING MACRO

LABEL	OP*N	OPERANDS
	INIT	

This macro initiates one input file, sets initial conditions in the MERGE routine, and selects one record. The INIT macro has to be performed for each file and therefore has to follow each OPEN1 macro.

■ Entrance Condition

The absolute address of the first record is in the four least significant characters of AR1.

■ Exit Conditions

- a) The absolute address of the selected record is in the four least significant characters of AR1.
- b) The binary file number of this record is in the least significant character of AR2.

The file number can be binary 1 to 12 and corresponds to the call sequence in the user's program.

If EOF (end of file) sentinel has been found when executing the OPEN1 macro ("no data tape"), a TERM call has to follow for this file instead of an INIT call and the next file has to be opened.

The number of input files can be decreased at object time by replacing the INIT call by a TERM call.

4.2 SELECT MACRO

LABEL	OP*N	OPERANDS
	MERGE	

The MERGE macro selects the next record. It has to be performed after the last selected record has been transferred to the output area, and the next record of that file has been accessed.

■ Entrance Condition

The absolute address of the next record is in the four least significant characters of AR1.

■ Exit Conditions

- a) The absolute address of the selected record is in the four least significant characters of AR1.
- b) The binary file number of this record is in the least significant character of AR2.

4.3 TERMINATING MACRO

LABEL	OP*N	OPERANDS
	TERM	

The TERM macro has to be executed for each input file after an EOF sentinel has been found or the file has been closed by CLOS1. The MERGE will terminate this file and then select a record of the remaining input files.

■ Entrance Condition

None.

■ Exit Conditions

- a) The absolute address of selected record is in the four least significant characters of AR1.
- b) The binary file number of this record is the least significant character of AR2.
- c) If this is the TERM call for the last input file, control is given to the instruction labeled ZEND in the user's program, where the output file has to be closed and the program completion stop has to be executed.

4.4 NOTES

All macro instructions alter the contents of AR1, AR2, and index registers 1 and 2.

If equality of records is found, the record with the smallest file number will be selected.

The MERGE routine is restartable, therefore a multicycle merge can be performed.

5.0 ESTIMATED STORAGE REQUIREMENTS

The SELECT coding requires:

$165 + 65f + 20n$ characters

where f = number of input files

n = number of fields in the key

INIT: 5 characters

MERGE: 10 characters

TERM: 5 characters

6.0 ESTIMATED EXECUTION TIMES

For a 1-field key (16 characters) and 3 input files,

INIT 2146.5 microseconds

MERGE 1930.5 microseconds

TERM 2205 microseconds

PROGRAM TWO-WAY MERGE

PROGRAMMER _____ DATE _____ PAGE _____ OF _____ PAGES

For BEGIN only.

SEQUENCE	LABEL	OPERATION	OPERANDS	COMMENTS	PROGRAM-ID
1	3	13	30	50	75
001	0.1	SMPL	BEGIN 020000		
	0.2	*STD		STANDARD LABEL DEFINITIONS	
	0.3	SELT	2, 1, D	2 INPUT FILES, 1 KEY, 1 DECIMAL ASCENDING	
	0.4	KEY1	5, 1, 5		
	0.5	TAPE1	2, STDBY, AR, 1, STDBY, TRF, 1, 2		
	0.6	A	FILE1 IN, 'FIRST', '000', A, 4, 1, 1, IMPA, INPA2, E0FA, AB, 60, 2, 248		
	0.7	B	FILE1 IN, 'SECOND', '000', A, 4, 2, 2, 2, INPB, INPB2, E0FA, AR, 60, 2, 248		
	0.8	C	FILE1 OUT, 'OUT', '000', A, 5, 3, 3, OUTC, OUTC2, TRF, 60, 2, 248		
	0.9	INPA	AREA 248		
	1.0	INPA2	AREA 248		
	1.1	INPB	AREA 248		
	1.2	INPB2	AREA 248		
	1.3	OUTC	AREA 248		
	1.4	OUTC2	AREA 248		
	1.5	IA	EQU 0		
	1.6	ORIG	0		
	1.7	ITEMA	AREA 60, 3	DUMMY AREA	
	1.8	ORIG	IA		
	1.9	START	OPENI C	OPEN OUTPUT FILE	
	2.0		OPENI A	OPEN FIRST INPUT FILE	

PROGRAM TWO-WAY MERGE PROGRAMMER _____ DATE _____ PAGE _____ OF _____ PAGES

For BEGIN only.

SEQUENCE PAGE	LABEL LINE	OPERATION 13	OPERANDS			COMMENTS	PROGRAM-ID
			18	19	30		
02	0.1	SC	E.O.F.A	9,32		SET CONNECTOR	
	0.2	INIT				INITIATE FIRST INPUT FILE	
	0.3	OPEN	B			OPEN SECOND INPUT FILE	
	0.4	INIT				INITIATE SECOND INPUT FILE	
	0.5	RTRN	BT	A,1,X3		ADDR. OF SELECTED RECORD TO IAS	
	0.6	CC	3,1,1			FILE A SELECTED?	
	0.7	JE	FLA				
	0.8	PUT	I,EMA,C			RECORD FILE B TO OUTPUT AREA	
	0.9	GET	B			NEXT RECORD FROM FILE B	
	1.0	J	Mrg				
	1.1	FLA	PUT	I,EMA,C			
	1.2	GET	A			NEXT RECORD FROM FILE A	
	1.3	Mrg	MERGE			SELECT ONE RECORD	
	1.4	J	RTRN				
	1.5	E.O.F.A	TERM			TERMINATE FILE A, SELECT RECORD	
	1.6	JC	O.P.B, 00			JUMP ONLY IF NO DATA ON FILE A	
	1.7	J	RTRN				
	1.8	E.O.F.B	TERM			TERMINATE FILE B, SELECT RECORD	
	1.9	J	RTRN				
	2.0	ZEND	C.LOSI	C, LOCK		ALL INPUT FILES TERMINATED	

PROGRAM TWO WAY MERGE PROGRAMMER _____ DATE _____ PAGE _____ OF _____ PAGES

For BEGIN only.

SEQUENCE	LABEL	OPERATION	OPERANDS	COMMENTS	PROGRAM-ID
PAGE 1 LINES 3 4 5 6 7	11	13 18 19	30 40 45 46	50 60 70	75 80
003	01	JR	0736	STOP SECTION OF EXECUTIVE	
	02	JD	077700	FINAL STOP	
	03	JR	0700	PROGRAM RELEASE	
	04	END	START		
	05				
	06				
	07				
	08				
	09				
	10				
	11				
	12				
	13				
	14				
	15				
	16				
	17				
	18				
	19				
	20				