1401 Subroutine #4
E. Melichar
Revised March 1, 1962

DISK OPERATION AND ERROR SUBROUTINE

Full Track, Move Mode

The Disk Operation and Error Subroutine, Full Track, Move Mode performs all disk and disk check operations. It is efficient both in speed of operation and in use of core memory. Unlike Subroutine #1, however, it can accommodate only full track operations in the Move Mode.

FUNCTION

In using the Subroutine, the programmer must establish a 1,000-digit field into which data from the Ramac will be read or from which data will be read onto the Ramac. This field must be both immediately preceded and immediately followed in core by a Group Mark Word Mark. The first Group Mark Word Mark must be immediately preceded in core by an 8-digit disk address, the low order position of which must be labeled RA. This label corresponds to practice of the programming group of the Bank.

To perform a disk operation, one of the five branch instructions listed below is given. The computer will branch to the Subroutine, execute the specified operation, and return to the main routine in the next sequential instruction following the branch instruction used to enter the Subroutine. The branch instructions with which the Subroutine may be entered, and the disk operations performed by each, are as follows:

B SEEKFT	Seek Disk
B READFT	Read Disk Full Track
B WRITFT	Write Disk Full Track and Write Disk Check
B SKRDFT	Seek Disk and Read Disk Full Track
B SKWTFT	Seek Disk, Write Disk Full Track, and Write Disk Check

After executing each of the above operations the Subroutine performs a check for disk errors by testing the Access Inoperable Indicator after a Seek Disk operation and the Any Disk-Unit Error Condition indicator after a Write Disk, Read Disk, or Write Disk Check operation. If either indicator is on at any of these tests, the appropriate operation or operations are reexecuted up to 8 times. If the error condition is not corrected after 8 tries, the computer executes a programmed halt in the Subroutine.

CORE SPACE REQUIREMENTS

The complete Subroutine occupies 197 digits of core memory. Since it replaces disk operation instructions and indicator tests in the main program with much shorter branch instructions, it effects considerable savings in core memory in most programs. In addition, since the Subroutine utilizes the original disk operation instructions in reseeking, rereading, and rewriting after a disk error condition, it uses no additional core over what would have been used by an error routine only.

If, as is likely, an individual program does not require the Subroutine to perform all five alternative operations listed above, core memory can be conserved by making one or more of the easy modifications listed below:

MODIFICATIONS

If a disk error condition is not corrected, the computer will execute the Halt instruction on line 120 of the Subroutine. This instruction can be modified to a Branch, Halt and Branch, or a Halt with specific indications in the A- and B- address registers by following the general procedure outlined under Modifications for Subroutine #1.

To conserve core memory, the Subroutine can be shortened by removing cards containing the instructions for each of the five alternative operations that are not used in a specific program. The following table lists the cards that can be removed for each Branch operation that is not used, and the number of digits by which the program will be shortened in each case:

Operation not used	Remove cards	Shortened by
B SEEKFT	010	4 digits
B READFT	130,135	8 digits
B WRITFT	155,157	11 digits
B SKRDFT	280,290,300	12 digits
B SKWTFT	310,320,330	15 digits
B READFT and B SKRDFT	130;135;140,153 280,290,300	32 digits
B WRITFT and B SKWTFT	080;085;155;157 159;160;161;162 163;164,170,310 320,330	83 digits

MAJOR ADDITION TO SUBROUTINE, May 1, 1962

The Subroutine has been modified to test that the Ramac address it is given prior to a Write Disk operation falls within limits which are specified in each program in which the Subroutine is used. The purpose of this change is to guard against writing information in the wrong area of the Ramac and thereby perhaps destroying valuable information stored in that area. It is thought that this check is particularly appropriate when a new program is being tested on a Ramac that already holds permanent files, and also appropriate in programs in which Ramac addresses are part of the input data, and thus could conceivably be in error at any time the program is run.

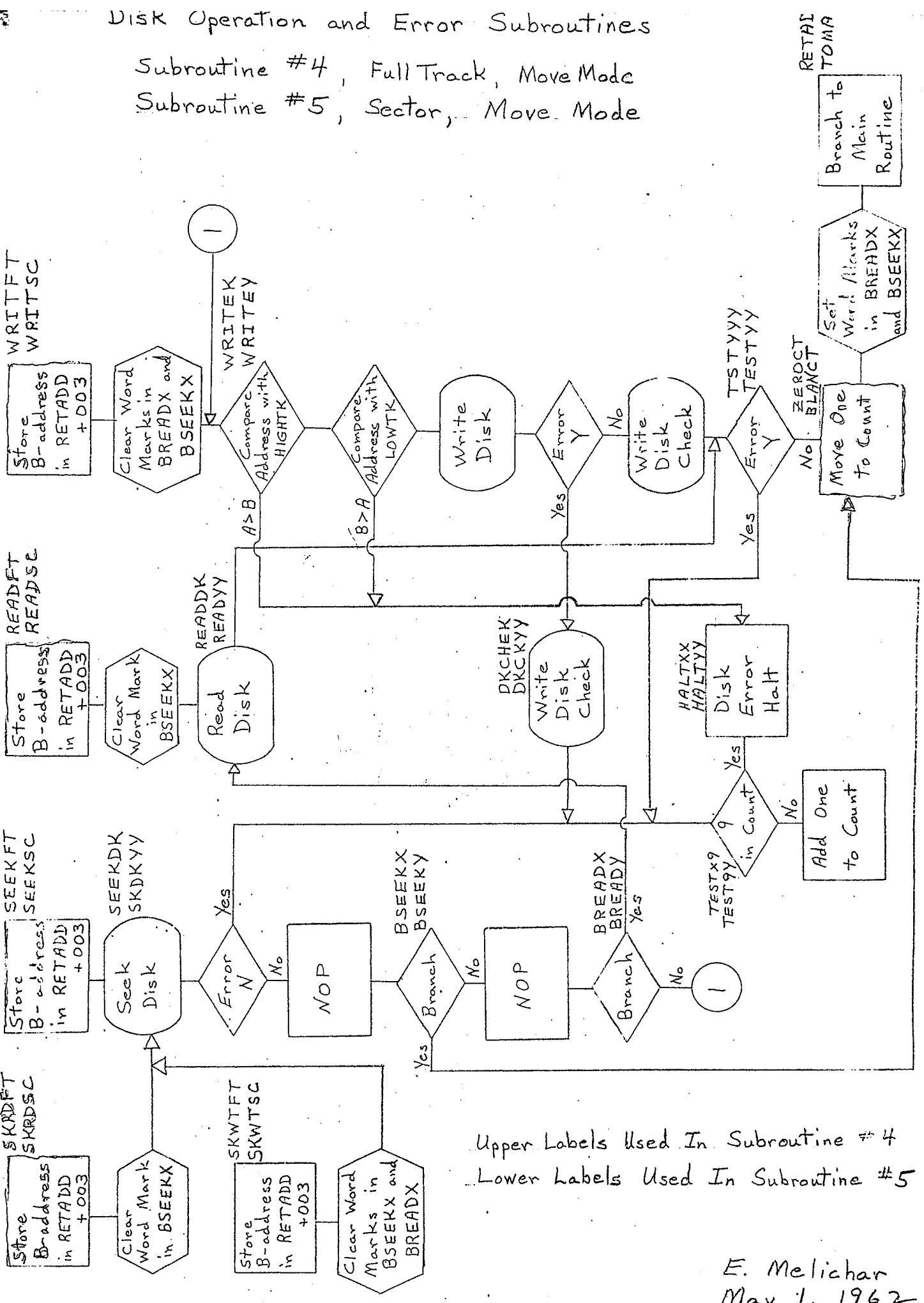
The programer must establish two 5-digit DCW fields in his program labeled LOWTK and HIGHTK, respectively. LOWTK must contain the lowest valid Ramac address and HIGHTK must contain the highest valid Ramac address used by the program in a Write Disk operation. If the Ramac address specified prior to a Write Disk operation falls outside these limits, the computer will execute a programmed halt in the Subroutine.

The 10 digits of core memory used by LOWTK and HIGHTK are included in the 197 digits used by the complete Subroutine, as specified on page 2. If this check for low and high valid Ramac addresses is not wanted, it can be removed from the Subroutine by removing cards 160, 161, 162, and 163 and labeling the instruction on card 164 "WRITEY." This modification will conserve 34 digits of core memory.

PG	LIN	CNT	LOC	LABEL	OP	A	ADJ X	B	ADJ XD	COM
1	000		0***		ORG	0333				
	10			SEEKFT	SBR	RETADD	≠003			DISK SUBROUTINE
	20	8	0333	SEEKDK	MCW	%F0		RA	-007 R	FULL TRACK
	30	5	0341		B	TESTX9			N	MOVE MODE
	40	1	0346		NOP					SUBROUTINE #4
	50	4	0347	BSEEKX	B	ZEROCT				
	60	1	0351		NOP					
	70	4	0352	BREADX	B	READDK				
	80	4	0356		B	WRITEK				
	85	8	0360	DKCHEK	MCW	%F3		RA	-007 W	
	90	8	0368	TESTX9	B	HALTXX		COUNTX	9	
	100	7	0376		A	ONEXXX		COUNTX		
	110	4	0383		B	SEEKDK				
	120	1	0387	HALTXX	H					DISK ERROR HALT
	130			READFT	SBR	RETADD	≠003			
	135	4	0388		CW	BSEEKX				
	140	8	0392	READDK	MCW	%F2		RA	-007 R	
	153	4	0400		B	TSTYYY				
	155			WRITFT	SBR	RETADD	≠003			
	157	.7	0404		CW	BREADX		BSEEKX		
3	159	7	0411	WRITEK	C	RA	-001	HIGHTK		
3	160	5	0418		B	HALTXX			T	
3	161	7	0423		C	RA	-001	LOWTK		
3	162	5	0430		B	HALTXX			U	
3	163	8	0435		MCW	%F2		RA	-007 W	
3	164	5	0443		B	DKCHEK			Y	
	170	8	0448		MCW	%F3		RA	-007 W	
	181	5	0456	TSTYYY	B	TESTX9			Y	
	190	7	0461	ZEROCT	MCW	ONEXXX		COUNTX		
	200				SW	BSEEKX		BREADX		
	210	4	0468	RETADD	B	0000				
	280			SKRDFT	SBR	RETADD	≠003			
	290	4	0472		CW	BSEEKX				
	300	4	0476		B	SEEKDK				
	310			SKWTFT	SBR	RETADD	≠003			
	320	7	0480		CW	BSEEKX		BREADX		
	330	4	0487		B	SEEKDK				
	340	4	0491	COUNTX	DCW	*				
	350	4	0495	ONEXXX	DCW	*	1			
	39	0166	0499	18						6 0000

Disk Operation and Error Subroutines

Subroutine #4, Full Track, Move Mode
Subroutine #5, Sector, Move Mode



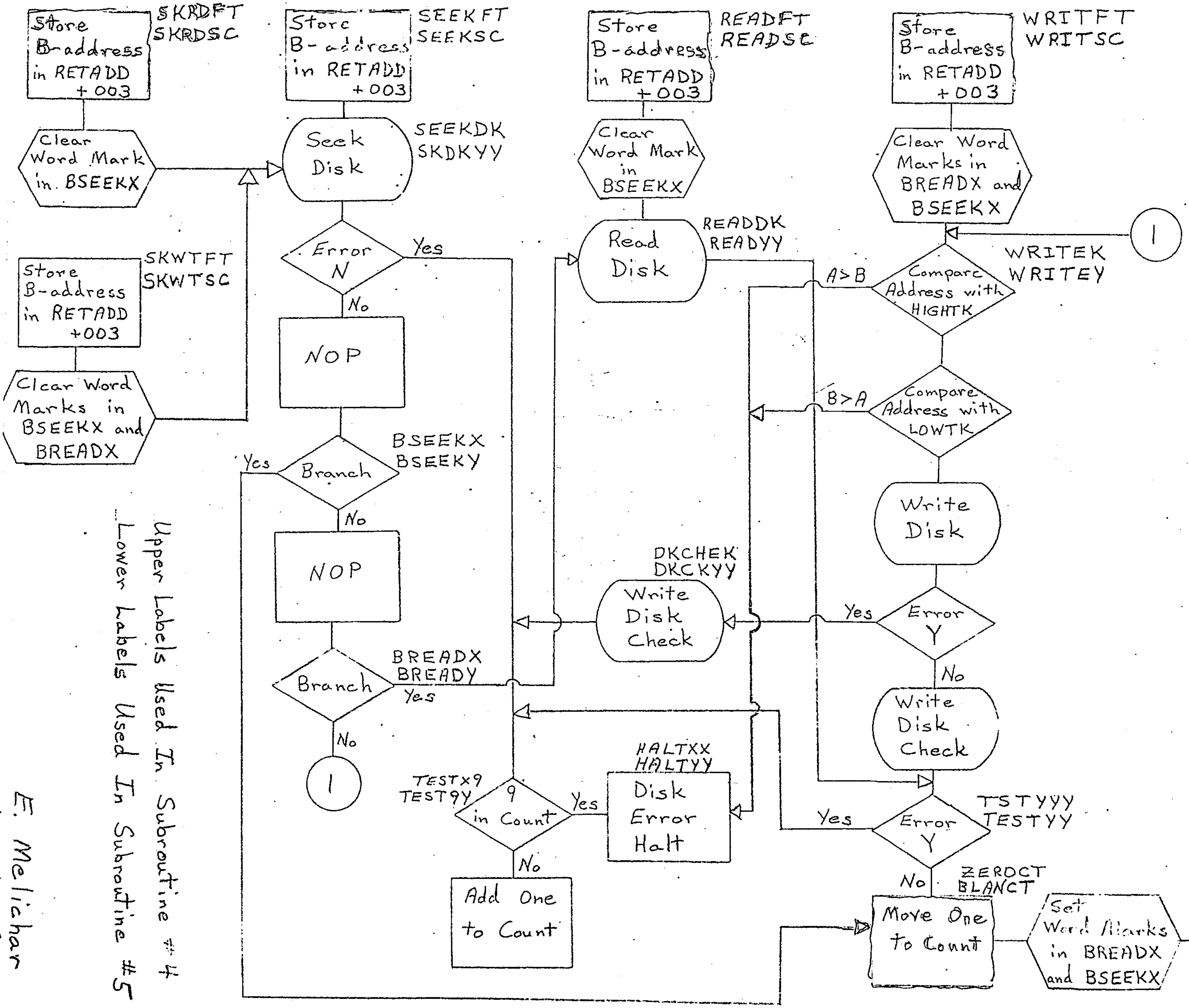
Upper Labels Used In Subroutine #4
Lower Labels Used In Subroutine #5

E. Melichar
May 1, 1962

Disk Operation and Error Subroutines

Subroutine #4, Full Track, Move Mode

Subroutine #5, Sector, Move Mode



Upper Labels Used In Subroutine #4

Lower Labels Used In Subroutine #5

E. Melichar

PRECAUTION

For computers with the indexing feature and with less than 16,000 digits of core storage, the Subroutine should be assembled between locations 1 and 999.