

**CENTRAL CONTROL  
PROGRAM INSTRUCTIONS  
NO. 1 ELECTRONIC SWITCHING SYSTEM**

CONTENTS	PAGE	CONTENTS	PAGE
1. GENERAL . . . . .	2	RESULTANT DAR ADDRESS . . . . .	9
2. INFORMATION CONTAINED ON A PROGRAM LISTING PAGE . . . . .	3	ARITHMETIC ZERO . . . . .	9
ADDRESS COLUMN . . . . .	3	LOGICAL ZERO . . . . .	9
ENCODED INSTRUCTION COLUMNS . . . . .	3	4. LOGICAL OPERATIONS . . . . .	9
EDITOR LINE AND MACRO LEVEL NUMBER . . . . .	3	LOGICAL PRODUCT (AND) . . . . .	9
A. Editor Line Number . . . . .	3	LOGICAL UNION (OR) . . . . .	9
B. Macro Level Number . . . . .	7	EXCLUSIVE OR . . . . .	9
LINE NUMBER COLUMN . . . . .	7	5. ABBREVIATIONS IN THE OPERATION CODES . . . . .	9
SYMBOLIC INSTRUCTION FORMAT . . . . .	7	ORIGINS AND DESTINATIONS . . . . .	9
A. Location Field . . . . .	7	OPERATIONS . . . . .	10
B. Operation Field . . . . .	7	CONDITIONS . . . . .	11
C. Variable Field . . . . .	7	SPECIAL INDICATORS . . . . .	11
COMMENT FIELD . . . . .	8	6. VARIABLE AND OPTION FIELDS . . . . .	11
3. PROGRAMMING TERMS . . . . .	8	DA SUBFIELD . . . . .	12
INDEXING . . . . .	8	R SUBFIELD . . . . .	12
INDIRECT TRANSFER . . . . .	8	M SUBFIELD . . . . .	12
EFFECTIVE DA NUMBER . . . . .	8	L SUBFIELD . . . . .	12
RESULTANT DAR NUMBER . . . . .	8	CJ SUBFIELD . . . . .	13
		7. ENCODING OF CC INSTRUCTIONS . . . . .	13

**NOTICE**

Not for use or disclosure outside the  
Bell System except under written agreement

CONTENTS	PAGE	CONTENTS	PAGE
<b>8. FLOW DIAGRAMS . . . . .</b>	<b>14</b>	<b>F. Interrupt Combined Instructions . . . . .</b>	<b>82</b>
<b>BASIC OPERATIONS . . . . .</b>	<b>16</b>	<b>MAINTENANCE AND SPECIAL PURPOSE INSTRUCTIONS . . . . .</b>	<b>83</b>
<b>GENERAL PURPOSE OPERATIONS . . . . .</b>	<b>19</b>	<b>9. GLOSSARY . . . . .</b>	<b>94</b>
<b>A. Move Operations . . . . .</b>	<b>19</b>	<b>FIGURES</b>	
<b>B. Add Operations . . . . .</b>	<b>27</b>	<b>1. Sample of a Program Listing Page . . . . .</b>	<b>6</b>
<b>C. Subtract Operations . . . . .</b>	<b>29</b>	<b>2. Breakdown of Encoding Columns . . . . .</b>	<b>7</b>
<b>D. Compare Operations . . . . .</b>	<b>31</b>	<b>3. Symbols and Abbreviations Used in Flow Diagrams . . . . .</b>	<b>15</b>
<b>E. Logical Operations . . . . .</b>	<b>34</b>	<b>TABLES</b>	
<b>F. Transfer Operations . . . . .</b>	<b>44</b>	<b>A. Alphabetic Listing of Instruction Mnemonic Codes Referenced to Instruction Definitions . . . . .</b>	<b>4</b>
<b>G. Combined Shift and Rotate Operations . . . . .</b>	<b>51</b>	<b>B. Categorical Index of Operation Codes . . . . .</b>	<b>5</b>
<b>H. Double Destination Operations . . . . .</b>	<b>56</b>	<b>C. Sample Instruction Encoding . . . . .</b>	<b>14</b>
<b>I. Miscellaneous Operations . . . . .</b>	<b>57</b>	<b>1. GENERAL</b>	
<b>INPUT-OUTPUT INSTRUCTIONS . . . . .</b>	<b>60</b>	<b>1.01</b>	This section describes the program instructions used by the central control (CC) of the 2 Wire No. 1 Electronic Switching System (ESS) and 4-Wire No. 1 ESS. Also included are the octal codes associated with the instructions and flow diagrams which describe the gating of the instructions processed by the CC.
<b>RESTRICTIONS ON INPUT-OUTPUT INSTRUCTIONS MA, WA, MAS, WAS, MSF, AND WSF . . . . .</b>	<b>69</b>	<b>1.02</b>	This section is reissued for the following reasons:
<b>A. K (Accumulator) Restrictions . . . . .</b>	<b>69</b>	(a)	To include Issue 2 Addendum 3
<b>B. F Register Restrictions . . . . .</b>	<b>70</b>	(b)	To revise paragraph 8.144
<b>C. Y Register Restrictions . . . . .</b>	<b>70</b>	(c)	To make minor corrections
<b>D. L Register Restrictions . . . . .</b>	<b>70</b>	(d)	To add Part 9 Glossary.
<b>COMBINED INSTRUCTIONS . . . . .</b>	<b>70</b>		
<b>A. General Purpose Combined Instructions . . . . .</b>	<b>70</b>		
<b>B. Dial Pulse Scan Combined Instructions . . . . .</b>	<b>72</b>		
<b>C. Line Scan Combined Instructions . . . . .</b>	<b>75</b>		
<b>D. Trunk Scan Combined Instructions . . . . .</b>	<b>77</b>		
<b>E. Network Combined Instructions . . . . .</b>	<b>79</b>		

Since this is a general revision, change arrows ordinarily used to indicate changes have been omitted.

**1.03** The instruction definitions and flow diagrams are shown in Part 8. Flow diagrams show the timing of instruction gating in the CC. The instructions are referred to by mnemonic codes. These codes are listed alphabetically in Table A with references to the instruction definitions shown in Part 8.

**1.04** The program instructions are also listed in another form as shown in Table B. Here the instructions are grouped according to category. The categories are identical to those given in Part 8 (Flow Diagrams). Table B is useful to find a particular CC instruction when a general form of the job to be carried out is known, such as an input-output job.

## 2. INFORMATION CONTAINED ON A PROGRAM LISTING PAGE

**2.01** A program listing (PR) is a computer generated list of instructions and related information for a program unit(s). A program unit is called a PIDENT (program identification). In order to interpret the instructions contained in a PIDENT, there should be an understanding of the information contained in a list of instructions (PR). Figure 1 gives an example of a typical list of program instructions on a page of a PR.

### ADDRESS COLUMN

**2.02** The left-most column of seven print positions contains the octal addresses, either relocatable or absolute, of the program instructions. A relocatable address is represented by only six print positions, whereas an absolute address occupies all seven print positions. A relocatable address is relative to the PIDENT's actual starting address, which can be found in the program map. The absolute address gives the actual program store (PS) address where the instruction is contained on the memory card. To get the absolute address for an instruction with a relocatable address in the PR, locate the actual octal starting address of the pident in the program map (PK-1A002-XX) and add the relocatable octal address to it. The addresses shown in Fig. 1 are relocatable addresses. If these columns are blank, then the line is used

for an auxiliary purpose such as a programmer's comment, control card, macro name, etc.

### ENCODED INSTRUCTION COLUMNS

**2.03** The two columns of five and eight octal digits in Fig. 1 represent the instruction in octal, without Hamming or parity check bits. (The 3-digit column preceding the encoding columns is used as a programming tool during system testing and is not discussed.) The five octal digits of the first column, representing bits 36 through 23 with the most significant octal digit representing bits 35 and 36, contain the encoding of the basic instruction and most of the options. The eight octal digits of the second column, bits 22 through 0, include the data or address (DA) field of the instruction. The DA field may consist of 23 or 21 bits depending on whether bits 22 and 21 are required to represent the octal encoding of the instruction. If the octal encoding for an instruction can be fully represented in bits 36 through 23, the DA field consists of 23 bits (0 through 22). For some options, bits 21 and 22 are included with bits 36 through 23 in representing the octal encoding for an instruction. In this case, the DA field consists of only 21 bits (0 through 20). In either case, the most significant octal digit of columns one and two represents only two bits. (The octal digit will not be greater than three.)

**2.04** The combined shift or rotate instructions make use of additional bits to further define the particular instruction. In addition to the bits already discussed, bit 19 is used to indicate the direction of shift or rotation and bits 18 through 14 specify the amount of shift or rotation.

**2.05** The letter L appended to column two of encoded instruction columns (Fig. 2) indicates that the DA field is a relocatable address. The letter V appended to column two indicates that this location contains a reference into a transfer table. This table contains the address for the transfer.

### EDITOR LINE AND MACRO LEVEL NUMBER

#### A. Editor Line Number

**2.06** The editor line number is inserted by an editor program which may be used in the assembly of the PR. The form of the number is

TABLE A

## ALPHABETIC LISTING OF INSTRUCTION MNEMONIC CODES REFERENCED TO INSTRUCTION DEFINITIONS

INSTRU- CTION	PAR. NO.										
ABR	8.31	HCMK	8.76	LG	8.133	PMY	8.48	QZM	8.69	TUPMK	8.115
AFR	8.31	HCMKU	8.76	LH	8.136	PMZ	8.48	SBR	8.35	TZRFU	8.108
AJR	8.31	HCØKM	8.80	LM	8.15	PWK	8.41	SFR	8.35	TZRFZ	8.109
AKR	8.31	HFM	8.69	LN	8.123	PWX	8.47	SJR	8.35	UMK	8.44
ALR	8.31	HJM	8.69	MA	8.91	PWY	8.47	SKR	8.35	UMK MJ	8.112
AMK	8.30	HKU	8.139	MAS	8.93	PWZ	8.47	SLR	8.35	UMX	8.50
AWK	8.29	HLM	8.68	MB	8.17	Q	8.53	SMK	8.34	UMY	8.50
AWRP	8.32	HMB	8.70	MBCS	8.145	QAMK	8.75	SWK	8.33	UMZ	8.50
AXR	8.31	HMBCS	8.70	MBØP	8.125	QBM	8.67	SXR	8.35	UWK	8.43
AYR	8.31	HMC	8.71	MC	8.18	QC	8.54	SYR	8.35	UWX	8.49
AZR	8.31	HMCII	8.72	MCII	8.19	QCMK	8.76	SZR	8.35	UWY	8.49
BG	8.132	HMF	8.74	MCLF	8.20	QCMKU	8.76	T	8.57	UWZ	8.49
BH	8.135	HMJ	8.74	MD	8.98	QFM	8.69	TAULM	8.118	WA	8.90
BM	8.13	HML	8.73	MF	8.23	QJM	8.69	TAUMK	8.111	WAS	8.92
BMAP	8.120	HMX	8.74	MFMK	8.82	QLM	8.68	TCAU	8.60	WB	8.25
BMØP	8.121	HMY	8.74	MJ	8.23	QMB	8.70	TCAZ	8.60	WD	8.97
BN	8.122	HMZ	8.74	MJMK	8.82	QMBCS	8.70	TCGE	8.60	WF	8.27
CMK	8.37	HØKM	8.79	MK	8.22	QMC	8.71	TCGMX	8.116	WJ	8.27
CMKU	8.39	HPMK	8.75	MKII	8.24	QMCII	8.72	TCLE	8.60	WK	8.28
CWK	8.36	HPMX	8.77	ML	8.21	QMF	8.74	TCLU	8.60	WL	8.26
CWKU	8.38	HPMY	8.77	MLMK	8.82	QMJ	8.74	TCLZ	8.60	WNPS	8.142
CWR	8.40	HPMZ	8.77	MSF	8.96	QML	8.73	TCM	8.60	WSF	8.95
EE	8.83	HSMK	8.75	MX	8.23	QMX	8.74	TCMMF	8.117	WV	8.143
EEF	8.88	HUMK	8.75	MXMK	8.82	QMY	8.74	TCP	8.60	WX	8.27
EGBN	8.87	HUMX	8.77	MY	8.23	QMZ	8.74	TKAU	8.59	WY	8.27
EMMS	8.144	HUMY	8.77	MYMK	8.82	QØKM	8.81	TKAZ	8.59	WZ	8.27
ENAM	8.85	HUMZ	8.77	MZ	8.23	QPMK	8.75	TKGE	8.59	XGKU	8.140
ENTJ	8.58	HXM	8.69	MZMK	8.82	QPMX	8.77	TKLE	8.59	XHKC	8.141
EXC	8.86	HXMK	8.75	NB	8.126	QPMY	8.77	TKLU	8.59	XM	8.16
EZEM	8.84	HYM	8.69	NBTA	8.130	QPMZ	8.77	TKLZ	8.59	XMK	8.46
FG	8.134	HZM	8.69	NBTB	8.131	QS	8.55	TKM	8.59	XN	8.124
FH	8.137	JKMSF	8.110	NF	8.129	QSC	8.56	TKP	8.59	XWK	8.45
FM	8.16	JM	8.16	NJ	8.129	QSMK	8.75	TRAU	8.61	YM	8.16
FN	8.124	JN	8.124	NK	8.128	QUMK	8.75	TRAZ	8.61	YN	8.124
GBNHJ	8.119	KG	8.134	NL	8.127	QUMX	8.77	TRGE	8.61	ZM	8.16
GKC	8.138	KH	8.137	NX	8.129	QUMY	8.77	TRLE	8.61	ZN	8.124
H	8.51	KM	8.14	NY	8.129	QUMZ	8.77	TRLU	8.61		
HAMK	8.75	KMKUS	8.113	NZ	8.129	QXM	8.69	TRLZ	8.61		
HBM	8.67	KMKXS	8.114	PMK	8.42	QXMK	8.75	TRM	8.61		
HC	8.52	KN	8.124	PMX	8.48	QYM	8.69	TRP	8.61		

**TABLE B**  
**CATEGORICAL INDEX OF OPERATION CODES**

CATEGORY	OPERATION		INSTRUCTIONS
GENERAL PURPOSE	MOVE		BM, FM, JM, KM, LM, MB, MC, MCII, MCLF, MF, MJ, MK, MKII, ML, MX, MY, MZ, WB, WF, WJ, WK, WL, WX, WY, WZ, XM, YM, ZM
	ADD		ABR, AFR, AJR, AKR, ALR, AMK, AWK, AWRP, AXR, AYR, AZR
	SUBTRACT		SBR, SFR, SJR, SKR, SLR, SMK, SWK, SXR, SYR, SZR
	COMPARE		CMK, CMKU, CWK, CWKU, CWR
	LOGICAL		H, HC, PMK, PMX, PMY, PMZ, PWK, PWX, PWY, PWZ, Q, QC, QS, QSC, UMK, UMX, UMY, UMZ, UWK, UWX, UWY, UWZ, XMK, XWK
	TRANSFERS		ENTJ, T, TCAU, TCAZ, TCGE, TCLE, TCLU, TCLZ, TCM, TCP, TKAU, TKAZ, TKGE, TKLE, TKLU, TKLZ, TKM, TKP, TRAU, TRAZ, TRGE, TRLE, TRLU, TRLZ, TRM, TRP
	STANDARD SHIFT AND ROTATE COMBINATIONS	MOVE	HBM, HFM, HJM, HLM, HMB, HMBCS, HMC, HMCII, HMF, HMJ, HML, HMX, HMY, HMZ, HXM, HYM, HZM, QBM, QFM, QJM, QLM, QMB, QMBCS, QMC, QMCII, QMF, QMJ, QML, QMX, QMY, QMZ, QXM, QYM, QZM
		ADD	HAMK, QAMK
		SUBTRACT	HSMK, QSMK
		COMPARE	HCMK, HCMKU, QCMK, QCMKU
		LOGICAL	HPMK, HPMX, HPMY, HPMZ, HUMK, HUMX, HUMY, HUMZ, HXMK, QPMK, QPMX, QPMY, QPMZ, QUMK, QUMX, QUMY, QUMZ, QXMK
	SPECIAL COMBINED SHIFT OR ROTATE		HCØKM, HØKM, QØKM
	DOUBLE DESTINATION		MFMK, MJMK, MLMK, MXMK, MYMK, MZMK
	MISCELLANEOUS		EE, EEF, EGBN, ENAM, EXC, EZEM
INPUT-OUTPUT			MA, MAS, MD, MSF, WA, WAS, WD, WSF
COMBINED			GBNHJ, JKMSF, KMKUS, KMKXS, TAULM, TAUMK, TUPMK, TCGMX, TCMMF, TZRFU, TZRFZ, UMKMJ
MAINTENANCE AND SPECIAL PURPOSE INSTRUCTIONS			BG, BH, BMAP, BMØP, BN, EMMS, FG, FH, FN, GKC, HKU, JN, KG, KH, KN, LG, LH, LN, MBCS, MBØP, NB, NBTA, NBTB, NF, NJ, NK, NL, NX, NY, NZ, WNPS, WV, XGKU, XHKC, XN, YN, ZN

EDITOR LINE NUMBERS AND MACRO LEVEL NUMBERS			
RELOCATABLE OCTAL ADDRESSES OF PROGRAM INSTRUCTIONS	ENCODED INSTRUCTION COLUMNS	LINE NUMBERS	SYMBOLIC INSTRUCTION FORMATS
003037	330 C0330 00014170	2026.	13 MZ WRK6
003040	010 00012 00014171	2027.	14 T WRK7,M
003041		2028.	15 INFOR 9GCX
		2029.	16 #LEN TRANSLATION IF PTW IS KNOWN
003041	722 03722 20000000	2030.	17 GLOBAL TRLPTW #K=PTW
003042	120 00120 00014173	2031.	18 TRLPTW WJ E(22),J
003C43	010 00010 00002410L	2032.	19 JM WRK9
		2033.	20 T TRENYY
003044		2034.	21 FEATURE
		2034.0010	22 INFOR 9GOPEN
		2034.0020	23 # ROUTINE TO RETRIEVE THE MTON USED TO PROVIDE BATTERY DURING OPEN
		2034.0030	24 # SWITCHING INTERVALS.
		2034.0040	25 # DSIP-OPEN SWITCHING INTERVAL PROBLEM.
		2034.0050	26 # INPUT: Y=LEN OUTPUT: B=CSIP WORD REGISTERS SAVED: X,Z
003044	750 06755 00376000	2034.0060	27 TRCSIP WK MASK(LENS),Y,PS # K=SELECTCR
003045	120 20160 12514173	2034.0070	28 HJM -DISP(LENS),WRK9 # SELECTCR RIGHT ADJUSTED
003046	342 10342 01760163	2034.0080	29 CMK F2MHTL+1 # COMPARE WITH MAXIMUM LENGTH
003047	352 00352 00014114	2034.0090	30 AMK F4HHTP+1 # K=A(A(SUBTRANSLATOR))
003050	031 00030 20003077L	2034.0100	31 TCGE OSIPER # T/TCC LARGE
003051	350 04350 00000000	2034.0110	32 MK 0,K # K=A(SUBTRANSLATOR)
003052	025 00024 20003077L	2034.0120	33 TKAZ OSIPER # T/UNASSIGNED
003053	534 04535 00001777	2034.0130	34 AYR MASK(INDX),K,PS # K=TAG
003054		2034.0140	35 WC E(20),K,PS # RIGHT CR LEFT HALF?
003054	430 14435 04000000	-001-	36 CNR E(20),K,PS
003055	035 10034 20000466L	2034.0150	37 TCAZ TRRCRT,,J # RC HUNT RIGHT
003056	005 10004 20000535L	2034.0160	38 ENTJ TRRCLF # RC HUNT LEFT

Fig. 1—Sample of a Program Listing Page

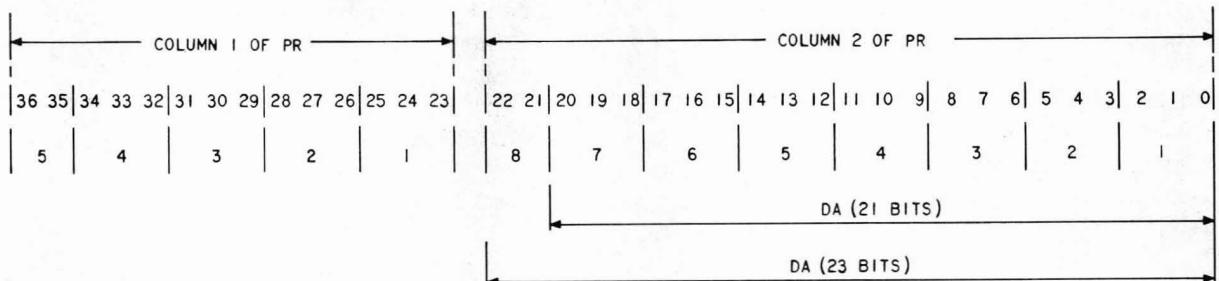


Fig. 2—Breakdown of Encoding Columns

XXXXX.XXXX or XXXXX, where the Xs are decimal digits.

#### B. Macro Level Number

**2.07** A macro instruction, or macro, is an abbreviated form for a sequence of predefined instructions or comments. A macro is an instruction the PR assembly program interprets and converts into zero or more valid machine instructions. Whenever a macro is used, the predefined sequence of valid machine instructions is generated in place of the macro and placed in the PR.

**2.08** The machine instructions generated by the macro are identified by the macro level number. The level number identifies which level of nested macros generated this line of the PR. The form of this number is recognizably different from the form of an editor line number. The number is of the form —XXX— where X is a decimal digit.

**2.09** If a line in the PR does not come from the editor program or is not generated by a macro, this column remains blank.

#### LINE NUMBER COLUMN

**2.10** This column is a sequence of two decimal digits which represent the line numbers on this page of the PR. These numbers start at 01 and may go up to 50.

#### SYMBOLIC INSTRUCTION FORMAT

**2.11** This format is comprised of three columns with each column representing a field. The left-most column is the location field; the middle

column is the operation field; and the right column is the variable field.

#### A. Location Field

**2.12** The location field is used for assigning a symbolic address to an instruction which can then be referred to by other instructions in the program. If the first character in this field is a sharp sign (#), the complete line in the listing is a comment only.

#### B. Operation Field

**2.13** The operation field contains the symbolic instruction codes or macro names. The instruction codes can consist of one to five letters. Macro names can be larger.

#### C. Variable Field

**2.14** The variable and option fields form one undivided field on the programming form. The field consists of seven possible subfields: HQ, A, DA, R, M, L, and CJ. Not all of the subfields can apply to any one instruction. For programming convenience, the subfields are grouped by commas into three or four major subfields. For all instructions except the standard combined shift or rotate class, the grouping is DA, RM, and LCJ. For the standard combined shift or rotate instructions, the grouping is HQ, A, R, and LCJ. When a subfield is not required in an instruction, it is left blank or, in the case of the DA field, made zero. The seven subfields are briefly described as follows.

(a) **HQ Subfield:** In the standard combined shift or rotate instruction, this subfield specifies the extent of the shift or rotation (bits 18 through 14) in decimal. Whether it is H,

HC, or Q is determined by encoding in bits 21 and 19.

- (b) **A Subfield:** In the standard combined shift or rotate instructions, this subfield specifies an address (or a part of an address when indexing is used) of limited size. Only 15 bits are used for this field, including the sign in bit 20 so that the absolute value of the contents of the A subfield cannot exceed octal 37,777 (16,383).
- (c) **Data or Address (DA) Subfield:** This subfield is used to specify either data or an address (or part of an address when indexing is used).
- (d) **R Subfield:** This subfield is used to specify one of the following CC registers:

LETTER	REGISTER
B	Buffer Register
F	First One Register
J	Return Address Register
K	Accumulator
X	Index Register X
Y	Index Register Y
Z	Index Register Z

If an instruction includes an option specifying register modification, the letter indicating the option also appears in the R subfield.

- (e) **M Subfield:** This subfield is used only for an indirect conditional or unconditional transfer. The presence of an M specifies an indirect transfer.
- (f) **L Subfield:** This subfield is used to specify the set logic register option (S) or logical masking option EL, ES, PL, or PS.
- (g) **CJ Subfield:** This subfield, when used, specifies the complement option C or the store return address option J.

#### COMMENT FIELD

**2.15** This field is reserved for explanatory comments, remarks, or references that may

be made by the programmer. This field is marked by a sharp (#) sign at the beginning of each comment.

### 3. PROGRAMMING TERMS

#### INDEXING

**3.01** A number is indexed by a given register when the contents of that register are algebraically added to the number.

#### INDIRECT TRANSFER

**3.02** When an indirect transfer (conditional or unconditional) is specified by the appearance of an M in the M subfield, the initial address is the resultant DAR address. This initial address may be either a call store (CS) or PS address that contains (in bits 19 through 0) the address to which the transfer is made.

#### EFFECTIVE DA NUMBER

**3.03** The effective DA number is the contents of the DA field unless an S, ES, or PS appears in the L subfield, in which case the effective DA number is defined to be zero. An S, ES, or PS in the L subfield specifies that the contents of the DA field are sent to the logic register (LR). For the standard combined shift or rotate instructions, the effective DA number is the contents of the A field, which may be blank or specify a signed number of absolute value not greater than 16,383 (octal 37777).

#### RESULTANT DAR NUMBER

**3.04** The resultant DAR number is the effective DA number indexed, if indexing occurs, by the contents of the register identified in the R subfield. The resultant DAR number consists of 22 data bits and a sign bit. When the DA field of an instruction consists of 21 bits (20 information bits and a sign bit), the field is expanded by inserting 0 or 1, whichever is in bit 20, into bit positions 21 and 22. During this expansion, if bit 20 of the DA field is a 1 and either or both bits 21 and 22 of the index register are ones, an overflow (end around carry) results if indexing occurs. The resultant DAR number can take one of the following forms:

- (a) The contents of the DA field plus the contents of an index register (provided that no S

appears in the L subfield and that indexing is used)

- (b) The contents of the DA field (provided no S appears in the L subfield and indexing is not used)
- (c) The contents of an index register (provided an S appears in the L subfield and indexing is available and used)
- (d) Zero (provided an S appears in the L subfield and indexing is not used).

#### RESULTANT DAR ADDRESS

**3.05** The resultant DAR address is that part of the resultant DAR number that is actually used as an address in the execution of the instruction. The resultant DAR number is calculated as a signed number, 22 bits plus a sign bit, and is used as such by instructions treating it as data. When it is a transfer address (PS or CS, direct or indirect), a buffer bus register, or address of data in CS, bits 0 through 19 are treated as a positive address and bits 20 through 22 are ignored. When it is the address of data in PS, bits 0 through 20 are treated as a positive address and bits 21 and 22 are ignored. (A 1 or 0 in bit 20 specifies the left or right part, respectively, of a data word in the PS.)

#### ARITHMETIC ZERO

**3.06** Arithmetic zero can be either plus zero (all zeros) or minus zero (all ones). The C control flip-flop indicate arithmetic zero by a 1 in the homogeneity bit regardless of the state of the sign bit.

#### LOGICAL ZERO

**3.07** Logical zero is the all-zero state of a 23-bit word (plus zero). The C control flip-flops indicate logical zero by a 1 in the homogeneity bit and a 0 in the sign bit.

### 4. LOGICAL OPERATIONS

#### LOGICAL PRODUCT (AND)

**4.01** When two binary numbers are combined by the logical product (AND) operation, each bit of one binary number is matched with the corresponding bit of the other binary number.

When both bits are ones, the result is a 1. When either bit is a 0, the result is a 0. For example:

$$\begin{array}{l}
 A = 1010 \\
 B = 0110 \\
 \text{Logical Product} = 0010
 \end{array}$$

#### LOGICAL UNION (OR)

**4.02** When both binary numbers are combined by the logical union (OR) operation, each bit of one binary number is matched with the corresponding bit of the other binary number. When both bits are 0, the result is a 0. When either bit is a 1, the result is a 1. For example:

$$\begin{array}{l}
 A = 1010 \\
 B = 0110 \\
 \text{Logical Union} = 1110
 \end{array}$$

#### EXCLUSIVE OR

**4.03** When two binary numbers are combined by the EXCLUSIVE OR operation, each bit of one binary number is matched with the corresponding bit of the other binary number. When these bits agree (both bits are ones or both are zeros), the result is a 0. When these bits do not agree (one bit is a 1 and the corresponding bit is a 0), the result is a 1. For example:

$$\begin{array}{l}
 A = 1010 \\
 B = 0110 \\
 \text{EXCLUSIVE OR} = 1100
 \end{array}$$

### 5. ABBREVIATIONS IN THE OPERATION CODES

**5.01** Letters appearing in the operation field have significant meanings. This part explains those letters.

#### ORIGINS AND DESTINATIONS

**5.02** The following letters mean the source from which information is moved or used or the

location to which information is sent in the execution of an instruction:

(a) **W:** The letter W appearing in the operation code designates the formation of the resultant DAR number W. The resultant DAR number is the effective DA number if indexing does not occur.

(b) **M:** The letter M appearing in the operation code means **one** of the following:

- (1) A memory location in either the CS or PS whose address is the resultant DAR address
- (2) A buffer bus register whose address is the resultant DAR address.

In each of the above cases, the specified address is called location M.

(c) **R:** The presence or absence of the letter R in the operation code specifies the use for the register identified in the R subfield. If there is an R in the operation code, the register specified in the R subfield represents the destination of the instruction. If the R is missing from the operation code, the register designated in the R subfield is used for indexing. Any one of the seven letters listed below can appear in the operation code or the R subfield. The indicated code is used in the binary encoding of an instruction involving the corresponding register.

LETTER	REGISTER	OCTAL CODE
—	No Register	0
B	Buffer Register	1
F	First One Register	2
J	Return Address (Jump) Register	3
K	Accumulator	4
X	Index Register X	5
Y	Index Register Y	6
Z	Index Register Z	7

All of the above registers can be used for indexing and are therefore referred to as index registers. In addition to the above listed registers, L for logic register may appear in the operation code, but never in the R subfield and, hence, is not an index register.

(d) **A, S, D, or V:** One of these letters appears as the second letter of the operation code of all input-output instructions to indicate the destination and function of the instruction as follows:

**A** — Peripheral Address Bus

**S** — Scanner

**D** — Central Pulse Distributor

**V** — A register on the buffer bus which functions to control flip-flops or to provide a pulse on one or more leads to perform miscellaneous functions.

(e) **N:** The letter N appears in the operation code of some maintenance instructions and refers to nonmemory locations. The nonmemory locations are control flip-flops or inspection points in certain units of the equipment such as the CS or PS.

(f) **G or H:** The letter G or H appears in the operation code of some maintenance instructions to designate a particular physical CS or PS. (These are the only instructions which explicitly refer to the duplication of units in the system.)

## OPERATIONS

**5.03** The following letters, when appearing as the first letter of the operation code, specify the action to be performed.

A—Add  
 S—Subtract  
 C—Compare  
 Q—Rotate  
 H—Shift (except in the maintenance instruction HKU)  
 P—Logical Product (AND)  
 U—Logical Union (OR)  
 X—EXCLUSIVE OR  
 T—Transfer

### CONDITIONS

**5.04** On conditional transfer instructions, the first letter after T specifies the source of information on which the decision for a transfer is made as follows:

- K:** The contents of the accumulator register (KR)
- C:** The current state of the C control flip-flops
- R:** The state of the C control flip-flops after being set according to the contents of the register identified in the R subfield.

The following letters appear last in the operation code of conditional transfer instructions. They specify the conditions which must exist in K, C, or R (whichever is named in the operation code) for a transfer to occur. S and H in the following list refer to the sign and homogeneity control flip-flops.

P — Plus	S   H	0   0   } 0   1   } }*
M — Minus	S   H	1   0   } 1   1   } }*

AZ — Arithmetic Zero	S   H	0   1   } 1   1   } }*
AU — Arithmetic Unzero	S   H	0   0   } 1   0   } }*
LZ — Logical Zero	S   H	0   1
LU — Logical Unzero	S   H	0   0   } 1   0   } 1   1   } }*
LE — Less than or equal to arithmetic zero	S   H	0   1   } 1   1   } 1   0   } }*
GE — Greater than or equal to arithmetic zero	S   H	0   1   } 1   1   } 0   0   } }*

\*Any one of the two or three combinations.

### SPECIAL INDICATORS

**5.05** The letter U appearing as the last letter of compare instructions CMKU and CWKU means that the homogeneity bit of the C control flip-flops will be set to the logical union (OR) of its former state and of the state corresponding to the difference resulting from the comparison.

**5.06** The letter E appearing as the first letter of the operation code denotes an exceptional symbolic operation code and means that the remaining letters are not necessarily used as the standard abbreviations described in this section. For example, NAM in the instruction ENAM is a mnemonic code for next address to memory.

### 6. VARIABLE AND OPTION FIELDS

**6.01** Throughout the following definitions of options and descriptions of their use, no implication is intended that a given option is available on all instructions nor that, if available, it must be used.

The particular options available with a given operation code are listed in Part 8.

#### DA SUBFIELD

**6.02** The DA or A subfield is used to specify data, an address, or a masking constant. When the ES or PS option is used in the L subfield, the contents of the DA subfield is sent to the L register for masking. Address information in the DA subfield is identified as the resultant DAR address. This is the address referred to when W is specified in an operation code. In an instruction in which the contents of the DA subfield is relevant, a blank DA subfield is interpreted as zero.

#### R SUBFIELD

**6.03** The R subfield, when used, specifies one of the following three functions.

(a) **Identification:** When the operation code contains an R, the R subfield specifies a register by providing the abbreviation of one of the seven index registers listed in 5.02(c). On any instruction whose operation code contains an R and the R subfield is blank, the instruction is executed as though the contents of the unidentified register are zero.

(b) **Set Register (S):** When the R subfield consists of the letter S followed by a register abbreviation, the specified register is set to the contents of the DA field. On conditional transfer instructions, the S option occurs only if the transfer occurs.

(c) **Indexing (I):** If neither (a) nor (b) applies, a register abbreviation in the R subfield will index the effective DA number. If an S, ES, or PS appears in the L subfield of the instruction, the effective DA number to be indexed is zero; in which case, the contents of the register identified in the R subfield become the resultant DAR number.

**6.04** In certain instructions, using the R subfield for identification or indexing, either of two additional letters (A or W) can appear in the R subfield accompanying the register abbreviation. The A or W changes the contents of the register upon execution of the instruction as follows.

(a) **A:** This letter, following the register abbreviation in the R subfield, causes the quantity one to be added algebraically to the contents of the register as follows.

(1) On transfer instructions, the A option (add one) occurs only if the transfer occurs and will be the last step in the execution of the instructions. For example, consider an executed transfer instruction where the R subfield specifies the J register and the A option and the CJ field specify the store return address option J. The quantity one will be added to the new return address after it is placed in the J register.

(2) On instructions other than transfers, the A option occurs immediately after the contents of the register have been used for either identification or indexing as determined by the instruction.

(b) **W:** This letter following the register abbreviation in the R subfield causes the result of the indexing operation to be placed in the register. For example, the W option will add the contents of the DA field to the contents of the register, and will place the result in the register, as well as use this result as the resultant DAR number. On conditional transfer instructions, the W option occurs only if the transfer occurs.

#### M SUBFIELD

**6.05** The M subfield is used only on transfer instructions. The letter M in the subfield specifies an indirect transfer. The resultant DAR address specifies either a CS or PS location that contains the actual transfer address. The DAR address is not the actual transfer address. The actual address is found in bits 19 through 0 of PS or bits 20 through 0 of CS.

#### L SUBFIELD

**6.06** The L subfield is used to specify masking (EL, ES, PL, PS) or the set-the-logic register option (S). The letter P indicates logical product masking (the AND function). The letter E indicates insertion masking that is used only on instructions sending information to either the CS, the data buffer bus register from any other CC register, or from the instruction word itself. The following operations are specified in the L subfield.

- (a) **PL or EL:** These options use the contents of the L register set by a previous instruction for product or insertion masking.
- (b) **PS or ES:** These options cause the contents of the DA subfield to be placed in the L register before the masking is done.
- (c) **S:** The letter S appearing in the L subfield causes the contents of the DA subfield to be placed in the L register, but no masking is done.

**6.07** Assume that an instruction specifies that its resultant DAR number be sent to the BR. On an instruction with the BR as the destination, an E in the L subfield specifies insertion masking control by the contents of the LR. The new information from the DA field or CC register, therefore, replaces the contents of the BR bits corresponding to bits of the LR that contain ones. The BR bits corresponding to LR bits that contain zeros are not changed. The words in the following examples contain 8 bits instead of 23 bits for simplicity:

Contents of CC Register	0011	0011
Contents of LR	1111	0000
Original Contents of BR	0101	0101
Final contents of BR after insertion masking has occurred	011	0101

When E is used in an instruction sending information to the BR as the ultimate destination, the information being sent sets the C control flip-flops. These flip-flops are set before insertion masking and after complementing if called for in the CJ subfield.

**6.08** An instruction that sends information to a destination in the CS and that specifies insertion masking involves the same operations as when the BR is the destination. In addition, the new contents of the BR replace the contents of the CS location specified in the instruction. Therefore, the operation is not a direct insertion into memory. The insertion changes the contents of the BR and the new contents are sent to the CS.

## CJ SUBFIELD

**6.09** This subfield, when used, specifies either of the following options.

- (a) **Complement Option C:** The letter C in the CJ subfield will cause the information going to the masked bus (MB) to be complemented in route to its destination. If a P appears in the L subfield, the logical product (AND) operation occurs before the complementing. If an E appears in the L subfield, the complementing occurs first followed by the insertion of the selected bits into the BR.
- (b) **Store Return Address Option J:** On all transfer instructions except ENTJ, the J option is specified by the letter J in the CJ subfield. This causes the address following the transfer instruction to be placed in the return address register J in the event a transfer does occur. The return address in this case is the address of the ENTJ instruction plus two.

## 7. ENCODING OF CC INSTRUCTIONS

**7.01** As discussed in 2.03, these are two columns in the PR dedicated to the octal encoding of instructions. The five-octal digit column (Fig. 2 and Table C) contains the octal encoding for the basic instruction and some of the options associated with that instruction. Some options are represented by bits 21 and 22 in column two of Fig. 2.

**7.02** The octal encoding of each instruction is given in Part 8. The basic encoding is in parentheses right after the symbolic instruction code. The options for the instruction are also listed with the octal code (in parentheses) representing the option.

**7.03** In order to better understand the encoding and deviation of complete instructions, a sample encoding is given. The sample instruction (Fig. 1) is CWR with options K (index register) and PS. The sum of the octal numbers, 14435, represents the octal encoding for the instruction and options. This is the manner in which the 5-digit number of Fig. 1 is derived. The same procedure is used for any instruction with any options. To arrive at the number in the PR, add the octal equivalent of the options to the basic

**TABLE C**  
**SAMPLE INSTRUCTION ENCODING**

5	4	3	2	1	DIGITS IN COL 1 IN PR (FIG. 2)
36 35	34 33 32	31 30 29	28 27 26	25 24 23	BITS OF PS WORD
0 1	0 0 0	1 0 0	0 1 1	0 0 0	BINARY REPRESENTATION OF BASIC INSTRUCTIONS
1	0	4	3	0	BASIC ENCODING FOR CWR
0	1 0 0	0	0	0	ENCODING FOR K REGISTER OPTION
	4				
0	0	0	0	1 0 1	ENCODING FOR PS OPTION
				5	
1	4	4	3	5	SUM

octal representation of the instruction. There are instances where options require bits 22 and 21 of column 2 (Fig. 2). In these cases, a 6-digit number is added to the basic code with the sixth being digit 8 of column 2 (Fig. 2). All numbers given in Part 8 are octal.

**7.04** For the combined shift or rotate instructions, an additional encoding bit is needed (2.04). Bit 19 is used in addition to bits 22 and 21. Bit 19 is not shown in Part 8. Only bits 22 and 21 are given for the shift and rotate instructions.

## 8. FLOW DIAGRAMS

**8.01** Definitions of symbols and abbreviations used in the flow diagrams are shown in Fig. 3.

**8.02** As described in Part 7, the flow diagrams include the octal encoding for the instructions and options. The codes are given in parentheses.

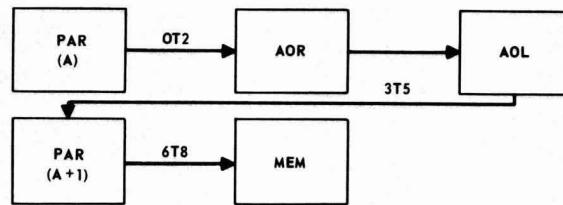
AOL	ADD ONE LOGIC	MB	MASKED BUS
AOR	ADD ONE REGISTER	MEM	MEMORY
ASR	AUXILIARY STORAGE REGISTER	OWD	ORDER WORD DECODER
ADD, AND, OR, EX-OR	ADD, AND, OR, EXCLUSIVE OR CIRCUIT	OWR	ORDER WORD REGISTER
ADDEND KR	ADDEND ACCUMULATOR REGISTER	PAR	PROGRAM ADDRESS REGISTER
AUGEND KR	AUGEND ACCUMULATOR REGISTER	PS	PROGRAM STORE
BOWD	BUFFER ORDER WORD DECODER	QR	Q REGISTER
BOWR	BUFFER ORDER WORD REGISTER	R	REGISTER (ANY REGISTER SPECIFIED BY THE INSTRUCTION)
BR	BUFFER REGISTER	(5) R	WHICHEVER ONE OF 5 REGISTERS IS SPECIFIED IN THE OP CODE
CS	CALL STORE	UB	UNMASKED BUS
COMP	DATA MUST BE COMPLEMENTED	XR	X INDEX REGISTER
FR	FIRST ONE REGISTER	YR	Y INDEX REGISTER
IADR	INDEX ADDEND REGISTER	ZR	Z INDEX REGISTER
IAUR	INDEX AUGEND REGISTER		MASKING OR COMPLEMENTING OPTIONAL
IAOR	INDEX ADDER OUTPUT REGISTER		DATA PASSES THROUGH MASK OR COMPLEMENT CIRCUIT UNCHANGED
JR	RETURN ADDRESS (JUMP) REGISTER	INSERT MASK	INSERTION MASKING
KR	ACCUMULATOR REGISTER	→	DIRECTION OF DATA FLOW BETWEEN CIRCUIT
LR	LOGIC REGISTER	OT6 →	DIRECTION OF DATA FLOW AND TIMING OF GATING SHOWN
MAD	MEMORY ADDRESS DECODER	—→	DIRECTION OF DATA FLOW DURING INDEXING CYCLE
		—OT14→	DIRECTION OF DATA FLOW-INDEX CYCLE TIMING AND GATING
		—PB→	DIRECTION OF DATA FLOW OVER PRIVATE BUS
		NOTE: CIRCUITS NOT SHOWN IN THIS FIGURE ARE IDENTIFIED WITHIN THE CIRCUIT BLOCK.	

Fig. 3—Symbols and Abbreviations Used in Flow Diagrams

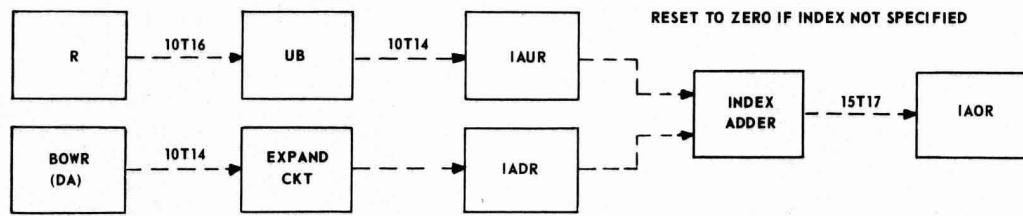
## SECTION 231-001-102

### BASIC OPERATIONS

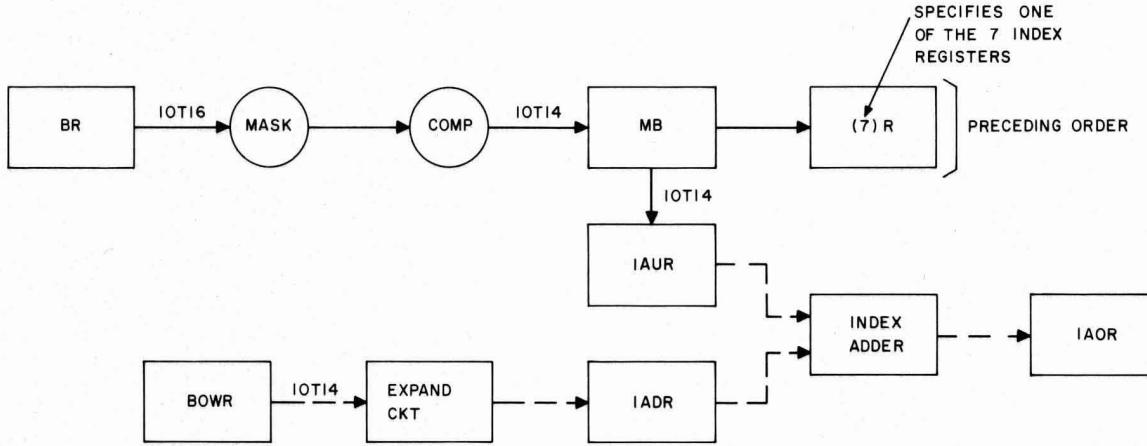
8.03 Timing for the normal incrementing of the PAR (program address register).



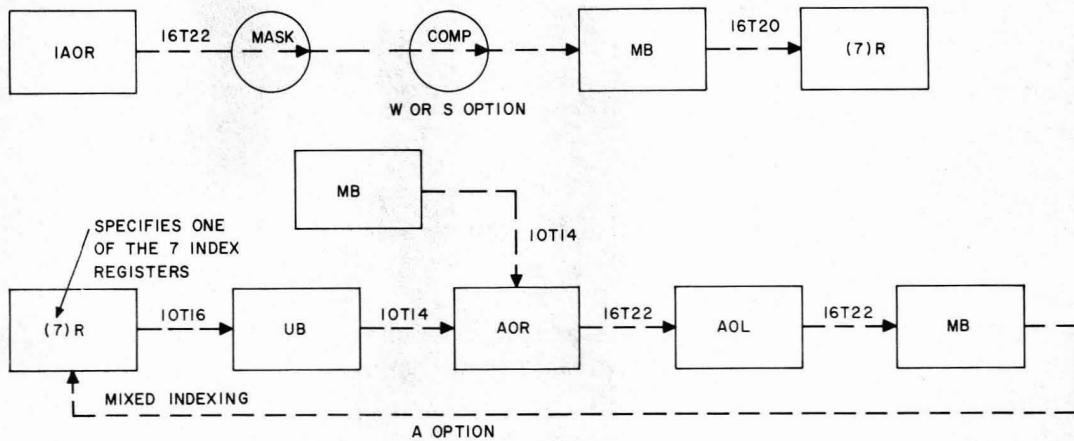
8.04 Timing for the indexing cycle.



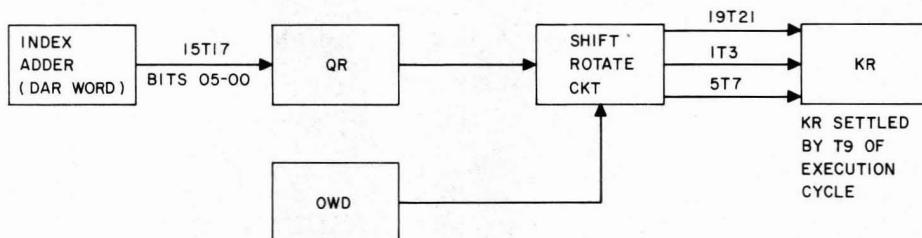
8.05 Timing for the mixed indexing cycle.



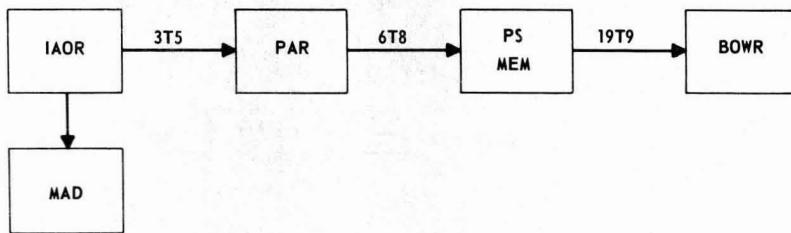
8.06 Timing for the R subfield options.



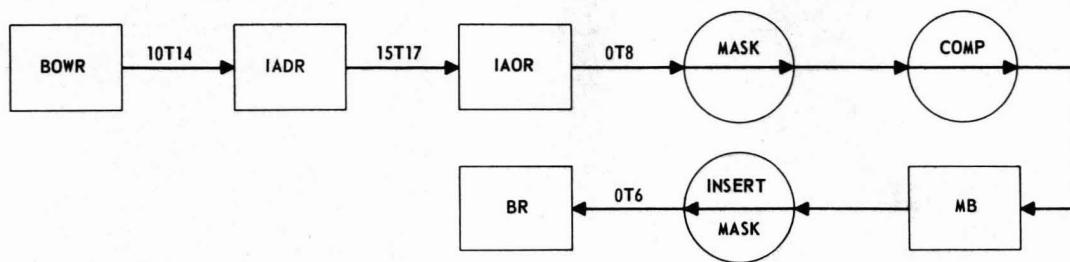
**8.07** Timing for the H and Q operations.



**8.08** A read data from memory operation using a PS address is shown below. The read-write sequencer is activated when the memory address decoder (MAD) recognizes that the address in the index adder output register (IAOR) is a PS address.

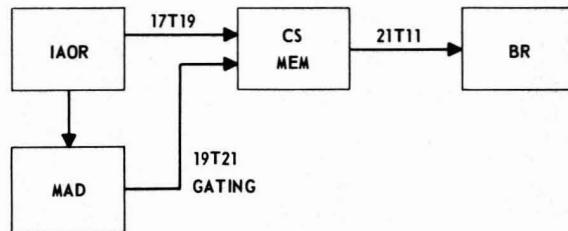


**8.09** The data obtained from the PS is gated to the BR. Once the data is in the BR, the processing is completed under control of the order word register (OWR). The instruction was retained in the OWR while data was being gated into the BR.

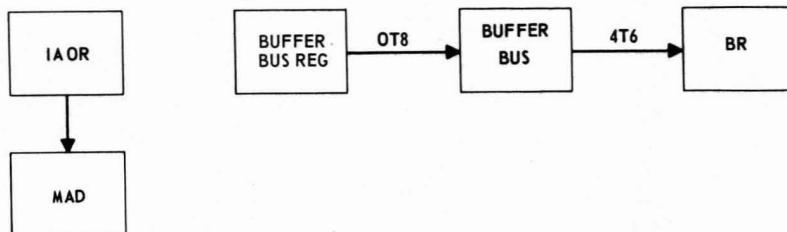


**8.10** Read-memory instructions that specify CS addresses require use of a sequencer when the effective execution time is more than one 5.5-microsecond machine cycle. The CS reading operations are as follows:

- Reading from a CC CS (one cycle)
- Reading from a single processor CS immediately after writing into a signal processor CS (four cycles)
- Reading from a signal processor CS immediately after any operation other than writing into a signal processor CS (two cycles).



**8.11** A read-memory instruction using a buffer bus register address reads data from any buffer bus register.



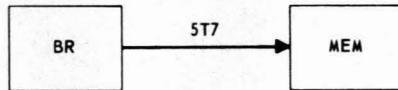
## GENERAL PURPOSE OPERATIONS

## A. Move Operations

**8.12** The effective execution time (in 5.5-microsecond machine cycles) of move operations where information in a CC register is written into a CS is as follows:

- (a) Writing into a CC CS (one cycle)
- (b) Writing into a signal processor CS immediately after writing into the same location or into another signal processor CS location (two cycles)
- (c) Writing into a signal processor CS immediately after any operation other than writing into a signal processor CS (one cycle).

**8.13 BM(00102)—Buffer Register to Memory:** The contents of the data buffer register (B) replace the contents of location M.



## Options:

## RM

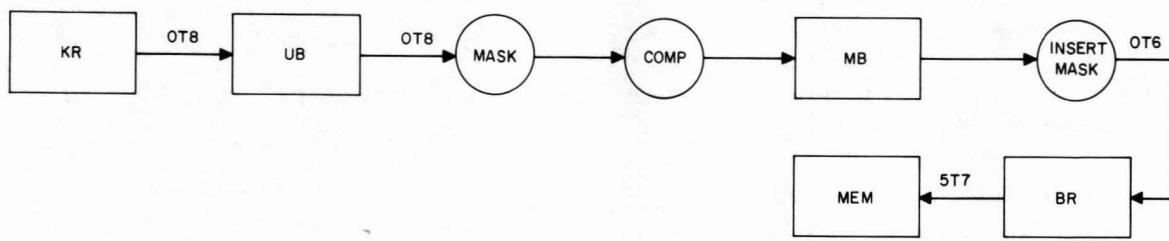
I (0V000 ;V = Index Reg No.)  
 A (20000)  
 W (00000 1)  
 S (20000 1)

## LCJ

ES (00003), The ES option is used  
 as S option

**Restrictions:** The data buffer register B must not appear in the R subfield except as generated by the MCBAM macro. If B is specified in the R subfield, the contents of B replace the contents of the memory location whose address was generated on the preceding instruction.

**8.14 KS(00112)—The contents of the accumulator K, after possible product masking and/or complementing, replace or are insertion masked into the BR. The contents of the BR replace the contents of location M.**



**Options:** RM

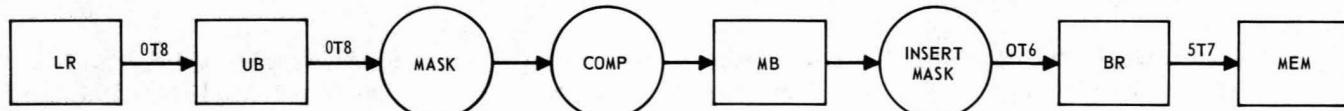
I(0V000 ;V = Index Reg No.)  
 A(20000)  
 W(00000 1)  
 S(20000 1)

LCJ

EL(00000 2)  
 PL(00004 2)  
 ES(00001)  
 PS(00005)  
 S(20041)  
 C(10000)

**Restrictions:** K must not appear in the R subfield except as generated by the MKKAM macro. If K is specified in the R subfield, the contents of K, after possible product masking, insertion masking, or complementing, replace the contents of the memory location whose address was generated on the preceding instruction.

**8.15 LM(00100)—Logic Register to Memory:** If neither ES nor PS masking is specified in the instruction, the contents of the LR, after possible complementing, replace or are insertion masked (EL) into the contents of the BR. The new contents of the BR replace the contents of location M. If either ES or PS masking is called for in the instruction, the contents of the DA field replace the contents of the LR. The new contents of the LR, after possible complementing, replace or are insertion masked into the contents of the BR. The new contents of BR replace the contents of location M.



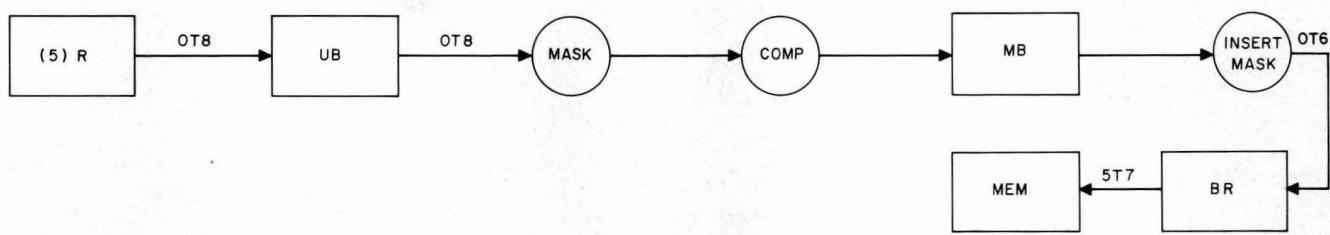
**Options:** RM

I(0V000 ;V = Index Reg No.)  
 A(20000)  
 W(00000 1)  
 S(20000 1)

LCJ

EL(00000 2)  
 ES(00001)  
 PS(00005)  
 C(10000)

**8.16 FM(00110), JM(00120), XM(00122), YM(00130), and ZM(00132):** The contents of register F, J, X, Y, or Z (whichever is specified in the operation code), after possible product masking and/or complementing, replace or are insertion masked into the BR. The contents of the BR replace the contents of location M.

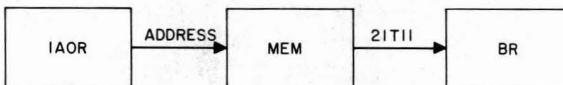
**Options:****RM**

I(0V000;V = Index Reg No.)  
 A(20000)  
 W(00000 1)  
 S(20000 1)

**LCJ**

EL(00000 2)  
 PL(00004 2)  
 ES(00001)  
 PS(00005)  
 S(20041)  
 C(10000)

**8.17 MB(00202)—Memory to Data Buffer Register:** The contents of location M replace the contents of the BR. The C control flip-flops are not affected by this operation.

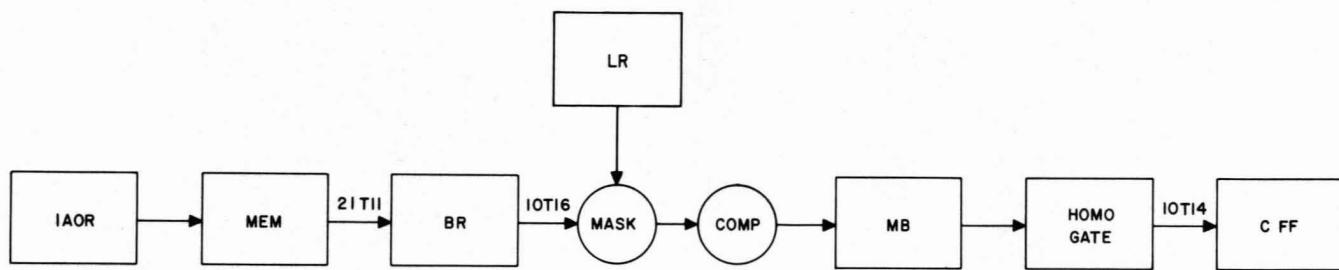
**Options:****RM**

I(0V000;V = Index Reg No.)  
 A(20000)  
 W(00000 1)  
 S(20000 1)

**LCJ**

S(20401)

**8.18 MC(00210)—Memory to C Control Flip-Flops:** The contents of location M replace the contents of the BR and, after possible product masking and/or complementing, set the C control flip-flops.



Options:

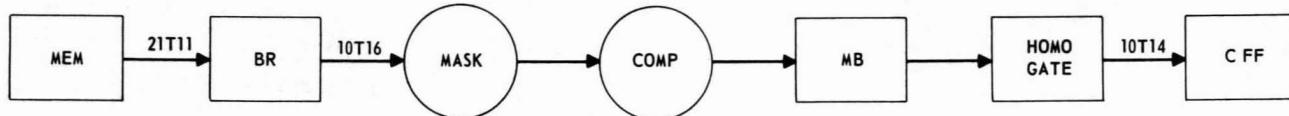
RM

LCJ

I(0V000 ;V = Index Reg No.)  
 A(20000)  
 W(00000 1)  
 S(20000 1)

PL(00000 2)  
 PS(00001)  
 S(20401)  
 C(10000)

**8.19 MCII(00212)—Memory to C Control Flip-Flops and Inhibit Interrupts:** The contents of location M replace the contents of the BR and, after possible product masking and/or complementing, set the C control flip-flops. During the following cycle, interrupts of levels H through K are inhibited and, if location M is in the SP call store, the SP is stopped during the following cycle.



Options:

RM

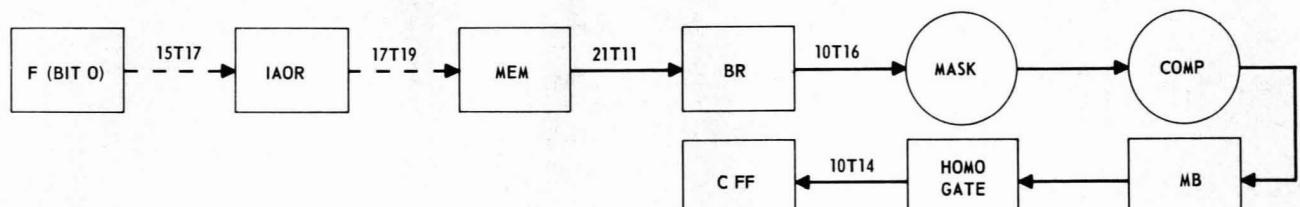
LCJ

I(0V000 ;V = Index Reg No.)  
 A(20000)  
 W(00000 1)  
 S(20000 1)

PL(00000 2)  
 PS(00001)  
 S(20401)  
 C(10000)

**Restrictions:** If MCII was the last instruction executed prior to an A through G interrupt, it must be repeated following the interrupt. The data buffer bus register B must not be used as an index register because it is destroyed on the first execution of MCII.

**8.20 MCLF(00214)—Memory to C Control Flip-Flops: Address Modified by Low Bit of First One Register (FR):** The least significant bit of the FR replaces the next to least significant bit of the resultant DAR address in computing the address of location M. The contents of this location replace the contents of the BR and, after possible product masking and/or complementing, set the C control flip-flops. If an index register other than the FR is specified with register modification S or W, the index register will be set to the resultant DAR address with bit position 1 modified by the least significant bit of the FR.

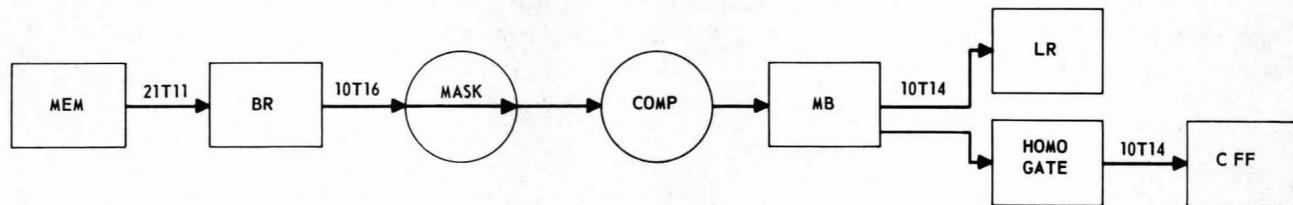
**Options:****RM****LCJ**

I(0V000 ;V = Index Reg No.)  
 A(20000)  
 W(00000 1)  
 S(20000 1)

PL(00000 2)  
 PS(00001)  
 S(20401)  
 C(10000)

**Restrictions:** If register F appears in the R subfield, register modification S, A, or W is not available.

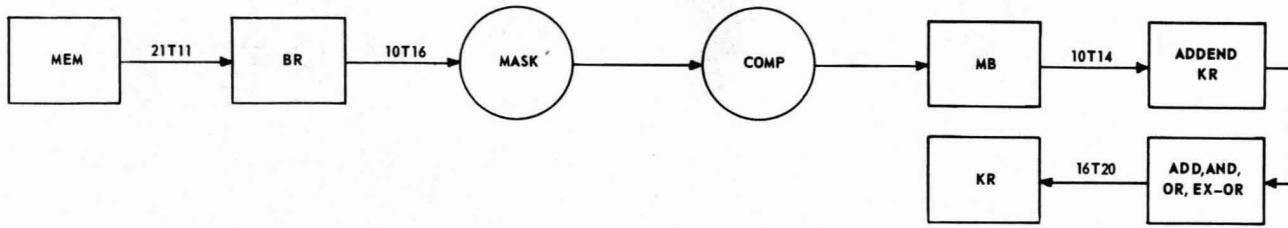
**8.21 ML(00320)—Memory to Logic Register:** The contents of location M replace the contents of the BR and, after possible complementing, set the C control flip-flops and replace the contents of the LR.

**Options:****RM****LCJ**

I(0V000 ;V = Index Reg No.)  
 A(20000)  
 W(00000 1)  
 S(20000 1)

C(10000)

**8.22 MK(00350)—Memory to Accumulator:** The contents of location M replace the contents of the BR and, after possible product masking and/or complementing, replace the contents of the KR. The C control flip-flops are not set by this operation.



Options:

RM

LCJ

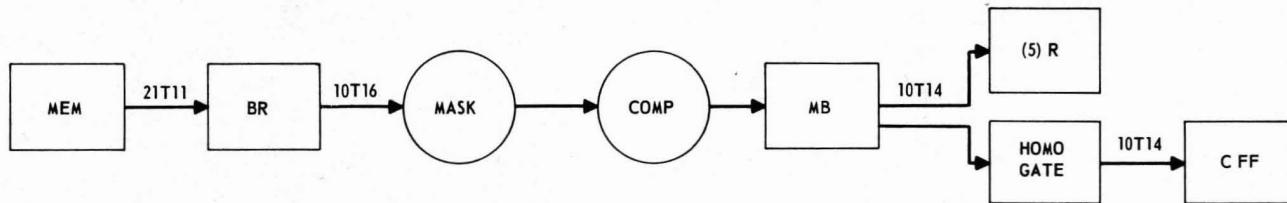
I(0V000; V = Index Reg No.)  
 A(20000)  
 W(00000 1)  
 S(20000 1)

PL(00000 2)  
 PS(00001)  
 S(20401)  
 C(10000)

Restrictions: The following instruction must not be

- (1) AKR or SKR
- (2) One of the following instructions with K in the R subfield: AWRP, CWR, and all TR instructions.

**8.23 MF(00222), MJ(00322), MX(00310), MY(00300), and MZ(00330)—Memory to Register:** The contents of location M replace the contents of the BR and, after possible product masking and/or complementing, set the C control flip-flops and replace the contents of the FR, JR, XR, YR, or ZR, whichever is specified in the operation code.



Options:

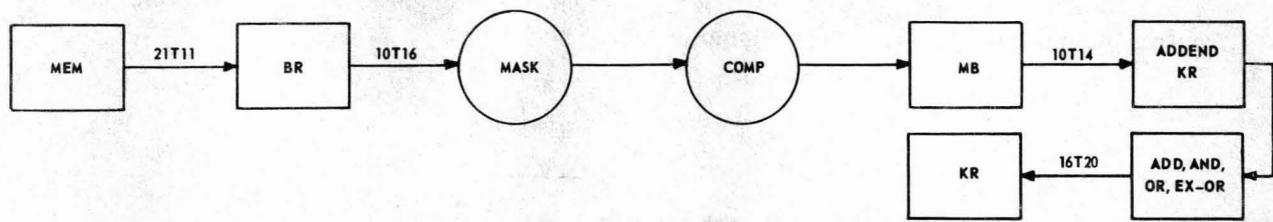
RM

LCJ

I(0V000; V = Index Reg No.)  
 A(20000)  
 W(00000 1)  
 S(20000 1)

PL(00000 2)  
 PS(00001)  
 S(20401)  
 C(10000)

**8.24 MKII(00376)—Memory to Accumulator; Inhibit Interrupts:** The contents of location M replace the contents of the BR and, after possible product masking and/or complementing, replace the contents of the KR. The sign bit of the C control flip-flops is set to 1 if the sign bit of the new contents of the KR is a 1 or either of the signal processor flag flip-flops contains a 1. If the sign bit of the new contents of the KR and both of the signal processor flag flip-flops are 0, the sign bit of the C control flip-flops will be 0. During the following cycle, interrupts of levels H through K are inhibited. If location M is in the SP call store, the SP is stopped during the following cycle. Neither the B register nor K register can be used as an index register because their contents are destroyed on the first execution of MKII.

**Options:****RM**

I(0V000 ;V = Index Reg No.)  
 A(20000)  
 W(00000 1)  
 S(20000 1)

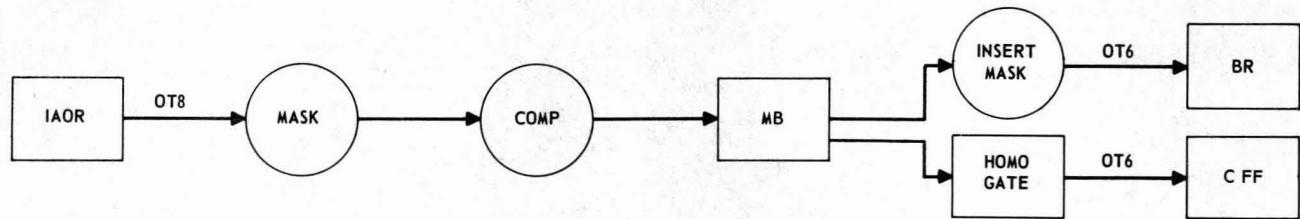
**LCJ**

PL(00000 2)  
 PS(00001)  
 S(20401)  
 C(10000)

**Restrictions:** The following instruction must not be

- (1) AKR or SKR.
- (2) One of the following instructions with K in the R subfield: AWRP, CWR, TR instructions.
- (3) If MKII was the last instruction executed prior to an A through G interrupt, it must be repeated following the interrupt. The data buffer bus register B must not be used as an index register because it is destroyed on the first execution of MKII.

**8.25 WB(00600)—Word to BR:** The resultant DAR number W, after possible product masking and/or complementing, sets the C control flip-flops, replaces the contents of the BR, or is insertion masked into the contents of the BR.

**Options:****RM**

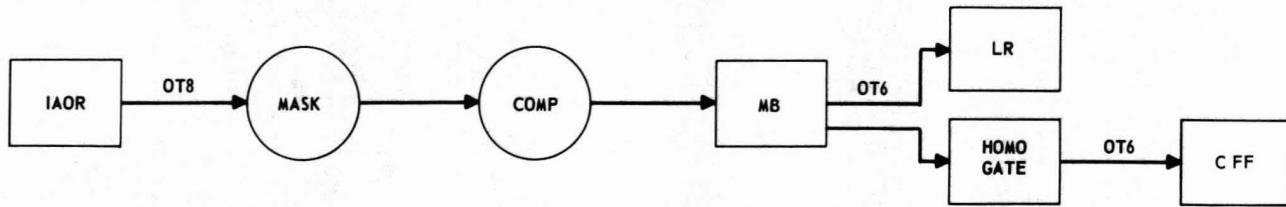
I(0V000 ;V = Index Reg No.)

**LCJ**

EL(00002)  
 ES(00003)  
 PL(00004)  
 PS(00005)  
 S(00001)  
 C(10000)

## SECTION 231-001-102

**8.26 WL(00720)—Word to LR:** The resultant DAR number W, after possible complementing, sets the C control flip-flops and replaces the contents of the LR.



Options:

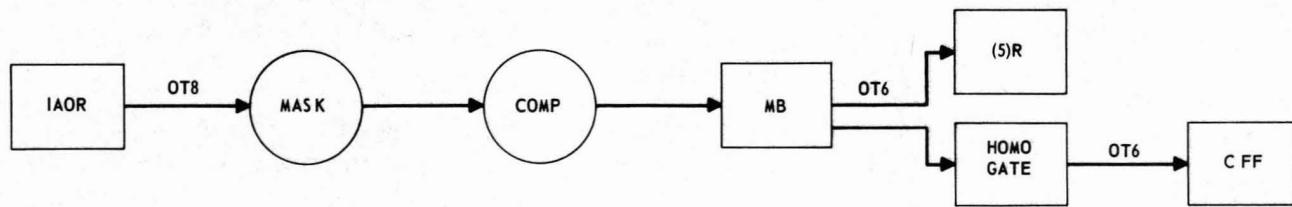
RM

LCJ

I(0V000;V = Index Reg No.)

C(10000)

**8.27 WF(00622), WJ(00722), WX(00710), WY(00700), and WZ(00730)—Word to Register:** The resultant DAR number W, after possible product masking and/or complementing, sets the C control flip-flops and replaces the contents of FR, JR, XR, YR, or ZR, whichever is specified in the operation code.



Options:

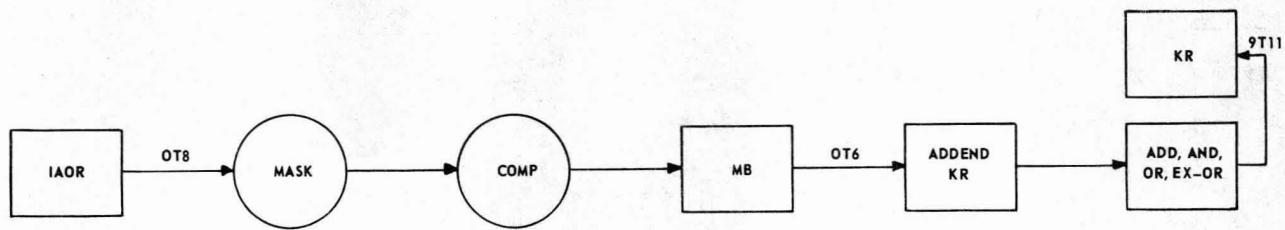
RM

LCJ

I(0V000;V = Index Reg No.)

PL(00004)  
PS(00005)  
S(00001)  
C(10000)

**8.28 WK(00750)—Word to Accumulator:** The resultant DAR number W, after possible product masking and/or complementing, replaces the contents of the KR. The C control flip-flops are not set.

**Options:****RM**

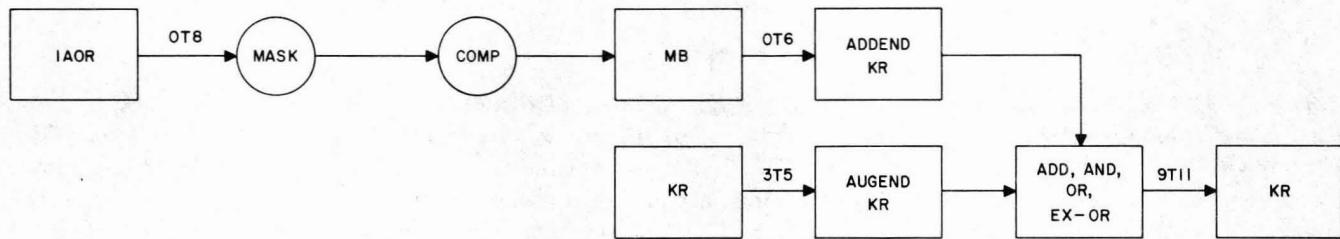
I(0V000;V = Index Reg No.)

**LCJ**

PL(00004)  
 PS(00005)  
 S(00001)  
 C(10000)

**B. Add Operations**

**8.29 AWK(00752)—Add Word to Accumulator:** The resultant DAR number W, after possible product masking and/or complementing, is added algebraically to the contents of the KR and the sum remains in the KR.

**Options:****RM**

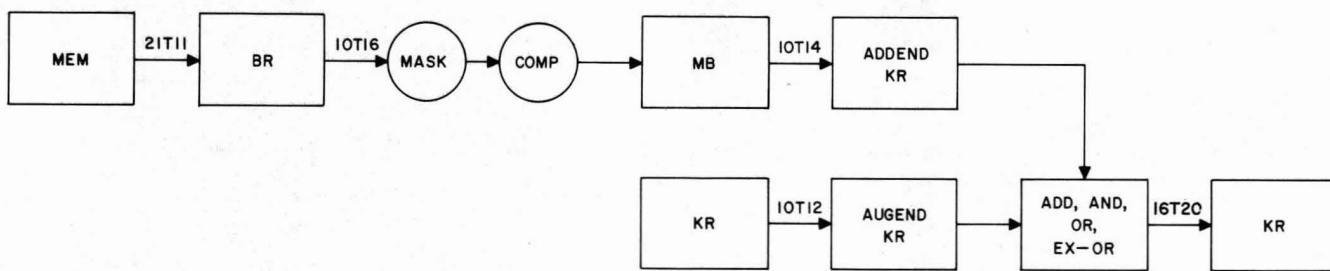
( I(0V000;V=Index Reg No.) )

**LCJ**

PL(00004)  
 PS(00005)  
 S(00001)  
 C(10000)

**8.30 AMK(00352)—Add Memory to Accumulator:** The contents of location M replace the contents of the BR and, after possible product masking and/or complementing, are added algebraically to the contents of the KR; the sum remains in the KR. The C control flip-flops are not set.

## SECTION 231-001-102



### Options:

#### RM

I(0V000;V=Index Reg No.)  
 A(20000)  
 W(00000 1)  
 S(20000 1)

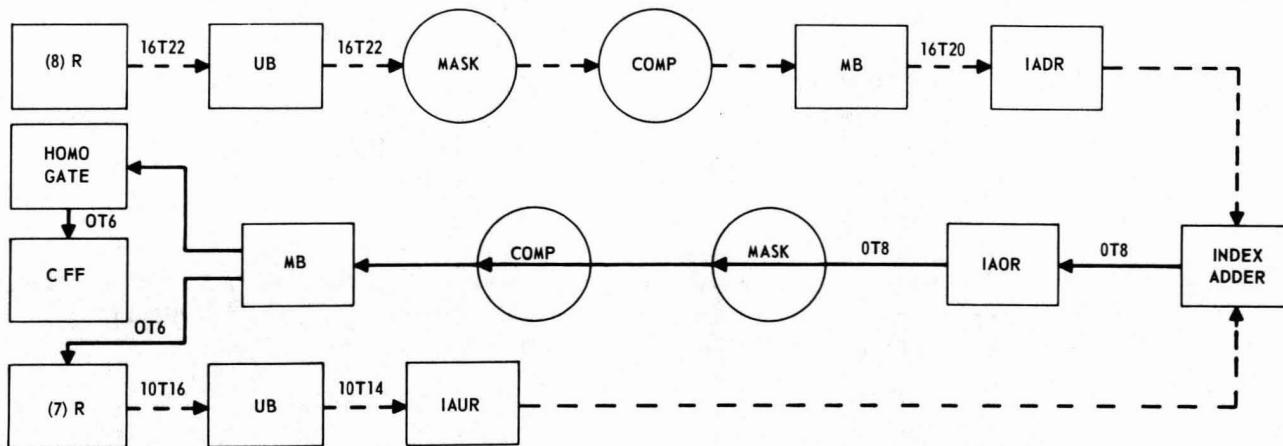
#### LCJ

PL(00000 2)  
 PS(00001)  
 S(20401)  
 C(10000)

**Restrictions:** The following instruction must not

- (1) Specify K in the R subfield
- (2) Be AKR or SKR.

**8.31 ABR(00522), AFR(00524), AJR(00526), AKR(00530), ALR(00520), AXR(00532), AYR(00534), and AZR(00536)—Add Contents of Registers Specified by Operation Field and R Subfield:** The contents of the BR, FR, JR, KR, LR, XR, YR, or ZR, whichever is specified in the operation code, after possible product masking and/or complementing, are added algebraically to the contents of the register identified in the R subfield (BR, FR, JR, KR, XR, YR, or ZR). The sum sets the C control flip-flops and replaces the original contents of the register identified in the R subfield.



### Options:

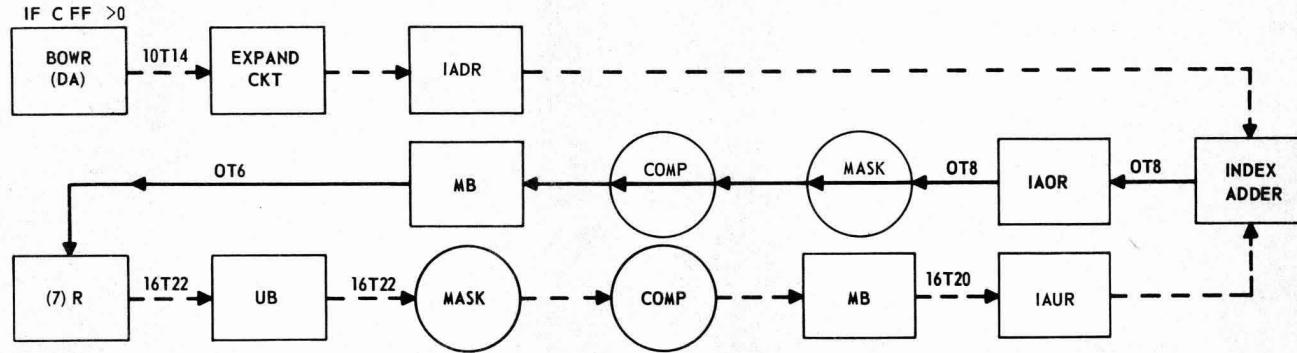
#### RM

I(0V000;V=Index Reg No.)

#### LCJ

PL(00000 2)  
 PS(00001)  
 C(10000)

**8.32 AWRP(00432)—Add Word to Register if C Control Flip-Flops Are Positive:** If the sign bit of the C control flip-flops is positive, the contents of the register identified in the R subfield (BR, FR, JR, KR, XR, YR, or ZR), after possible product masking and/or complementing, are added algebraically to the effective DA number. This sum replaces the contents of the identified register. The C control flip-flops are not affected. If the sign bit of the C control flip-flop is negative, no action takes place.



Options:

RM

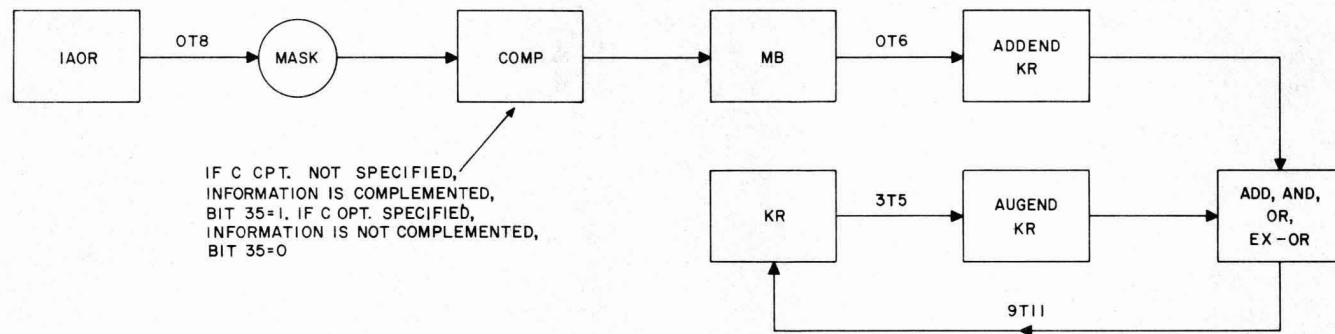
I(0V000;V=Index Reg No.)

LCJ

PL(00004)  
C(10000)

### C. Subtract Operations

**8.33 SWK(10752)—Subtract Word from Accumulator:** The resultant DAR number W, after possible product masking and/or complementing, is subtracted from the contents of the KR and the difference remains in the KR. The C control flip-flops are not set.



Options:

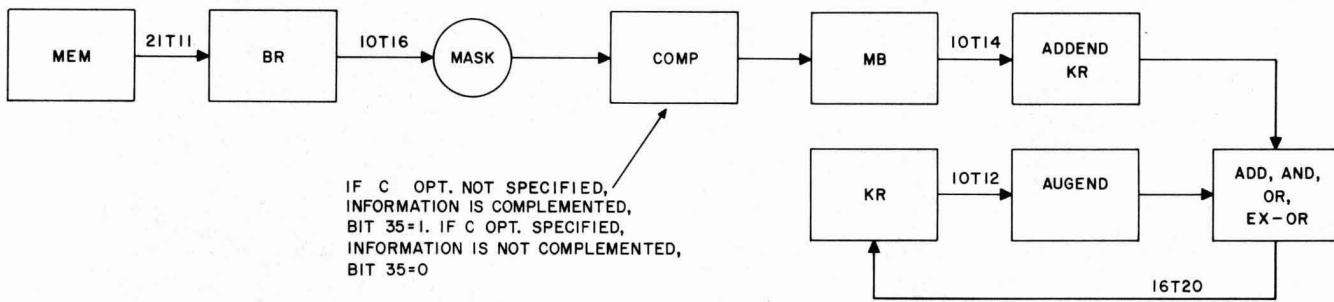
RM

I(0V000;V=Index Reg No.)

LCJ

PL(00004)  
PS(00005)  
S(00001)  
C(-10000)

**8.34 SMK(10352)—Subtract Memory from Accumulator:** The contents of location M replace the contents of the BR and, after possible product masking and/or complementing, are subtracted from the contents of the KR and the difference remains in the KR. The C control flip-flops are not set.

**Options:****RM**

- I(0V000;V=Index Reg No.)
- A(20000)
- W(00000 1)
- S(20000 1)

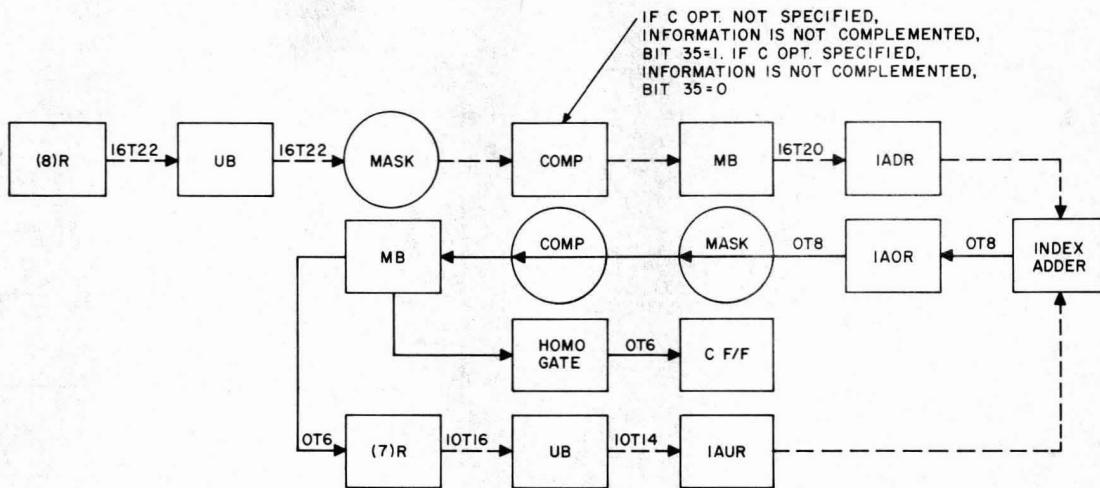
**LCJ**

- PL(00000 2)
- PS(00001)
- S(20401)
- C(-10000)

**Restrictions:** The following instruction must not

- (1) Specify K in the R subfield
- (2) Be AKR or SKR.

**8.35 SBR(10522), SFR(10524), SJR(10526), SKR(10530), SLR(10520), SXR(10532), SYR(10534), and SZR(10536)—Subtract Contents of Register Specified by Operation Field from Register Specified by R Subfield:** the contents of the BR, FR, JR, KR, LR, XR, YR, or ZR, whichever is specified in the operation code, after possible product masking and/or complementing, are subtracted from the contents of the register identified in the R subfield (BR, FR, JR, KR, XR, YR, or ZR). The difference sets the C control flip-flops and replaces the contents of the register specified in the R subfield.



Options:

RM

I(0V000;V=Index Reg No.)

LCJ

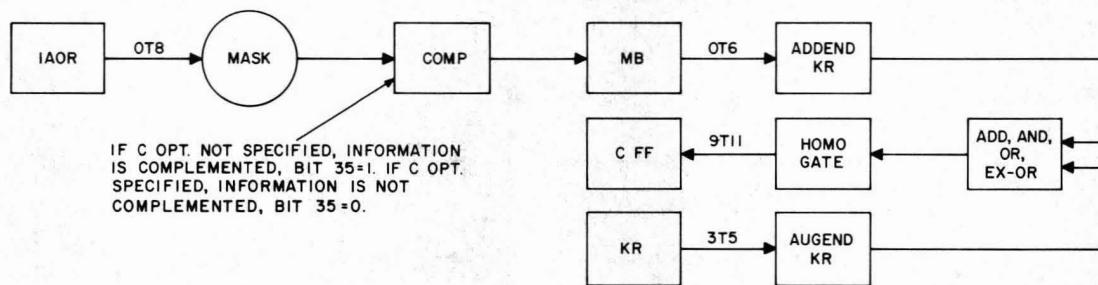
PL(00000 2)

PS(00001)

C(-10000)

#### D. Compare Operations

**8.36 CWK(10742)—Compare Word with Accumulator:** The resultant DAR number W, after possible product masking and/or complementing, is subtracted from the contents of the KR. The contents of KR remain unchanged, but the C control flip-flops are set according to the difference.



Options:

RM

I(0V000;V=Index Reg No.)

LCJ

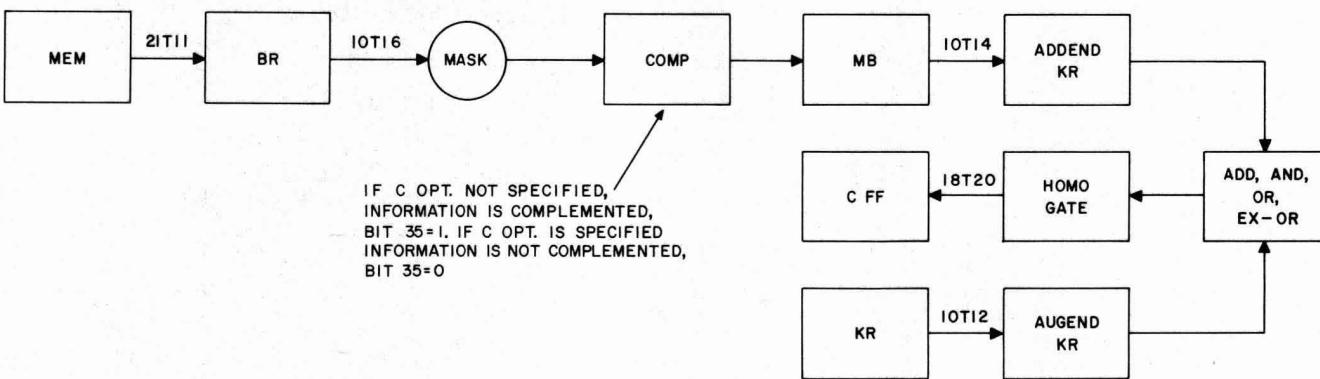
PL(00004)

PS(00005)

S(00001)

C(-10000)

**8.37 CMK(10342)—Compare Memory with Accumulator:** The contents of location M replace the contents of the BR and, after possible product masking and/or complementing, are subtracted from the contents of the KR. The contents of the KR remain unchanged, but the C control flip-flops are set according to the difference.



Options:

RM

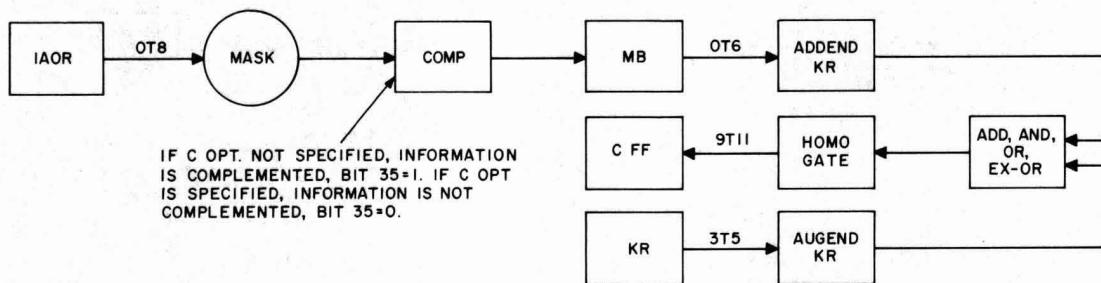
I(0V000;V=Index Reg No.)  
A(20000 0)  
W(00000 1)  
S(20000 1)

LCJ

PL(00000 2)  
PS(00001)  
S(20401)  
C(-10000)

**Restrictions:** The following instruction must not be one of the early transfer instructions: TAULM, TAUMK, TUPMK, TCGMX, or TCMMF.

**8.38 CWKU(10740)—Compare Word with Accumulator; OR Homogeneity:** This instruction is the same as the CMK instruction except that the C control homogeneity flip-flop is set to the logical union (OR) of its former state and the state corresponding to the difference. The homogeneity bit will be a 1 after the instruction if it were a 1 before the instruction or if the difference resulting from the interval subtraction is homogeneous (all ones or all zeros).



Options:

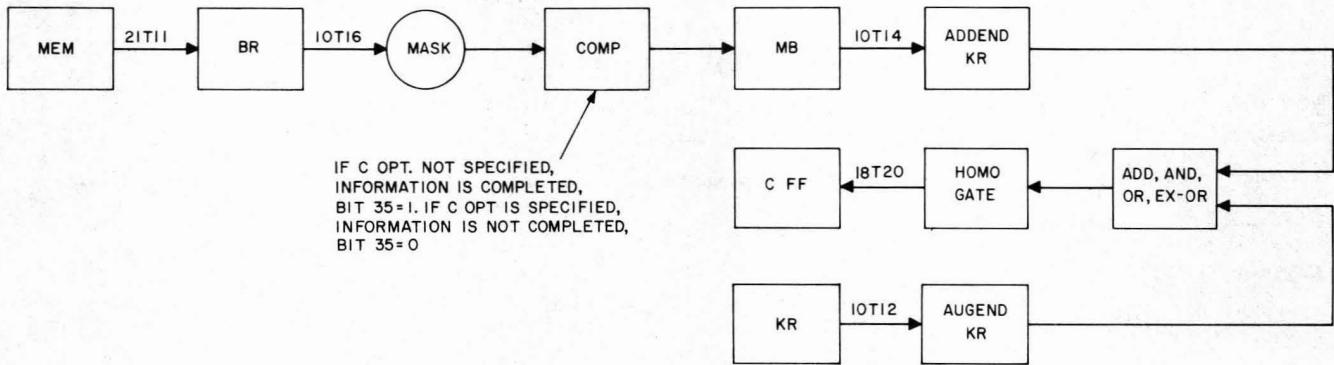
RM

I(0V000;V=Index Reg No.)

LCJ

PL(00004)  
PS(00005)  
S(00001)  
C(-10000)

**8.39 CMKU(10340)—Compare Memory with Accumulator; OR Homogeneity:** This instruction is the same as the CMK instruction except that the C control homogeneity flip-flop is set to the logical union (OR) of its former state and the state corresponding to the difference. The homogeneity bit will be a 1 after the instruction if it were a 1 before the instruction or if the difference resulting from the internal subtraction is homogenous (all ones or all zeroes).

**Options:****RM**

I(0V000;V=Index Reg No.)  
A(20000)  
W(00000 1)  
S(20000 1)

**LCJ**

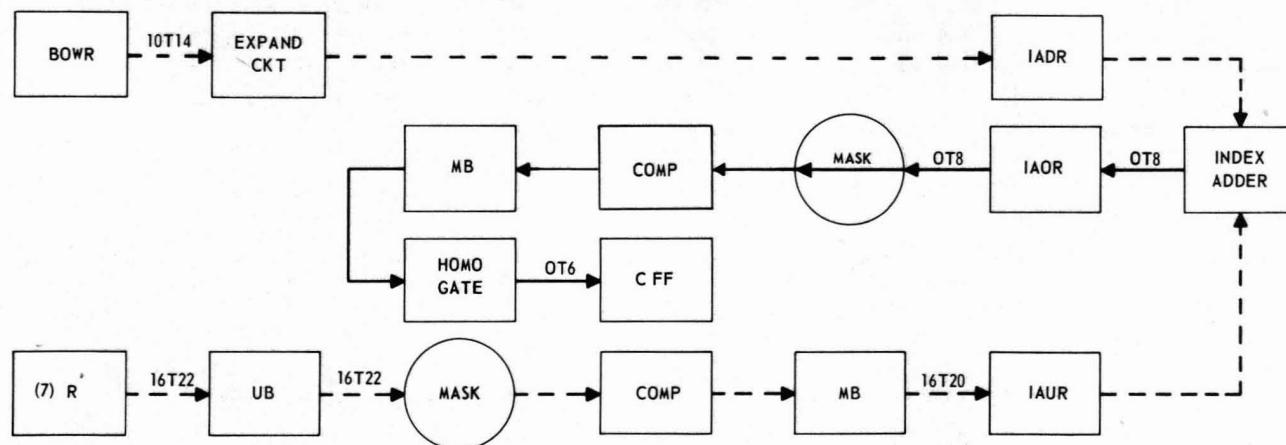
PL(00000 2)  
PS(00001)  
S(20401)  
C(-10000)

**Restrictions:** The following instruction must not be one of the early transfer instructions; TAULM, TAUMK, TUPMK, TCGMX, or TCMMF.

**8.40 CWR(10430)—Compare Word with Register:** The contents of the register identified in the R subfield (B, F, J, K, X, Y, or Z) may be product-masked and, from this, the effective DA number is subtracted. The original contents of the identified register remain unchanged, but the C control flip-flops are set according to the difference.

**Note:** CWR differs from the other compare instructions (CWK, CMK, CWKU, CMKU) in the following ways.

- (1) It is R rather than W that is subject to masking.
- (2) When W equals R, the C control flip-flops are set to logical (plus) zero.



Options:

RM

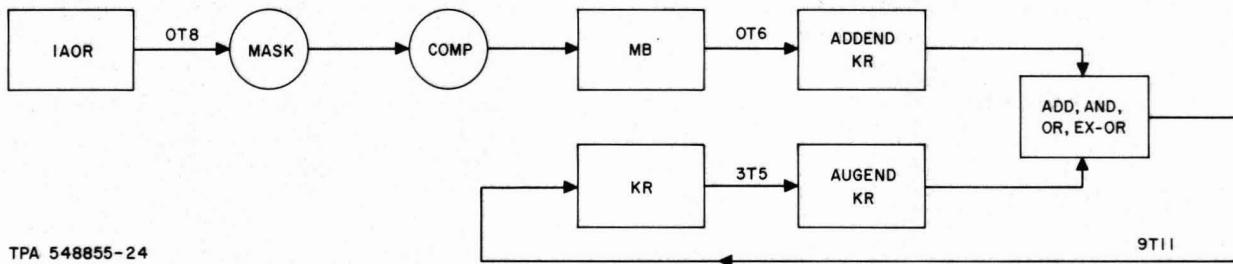
I(0V000;V=Index Reg No.)

LCJ

PL(00004)  
PS(00005)  
S(00001)

## E. Logical Operations

**8.41 PWK(00642)—Logical Product (AND) of Word with Accumulator:** The resultant DAR number W, after possible product masking and/or complementing, is combined by the logical product (AND) function with the contents of the KR; the result replaces the original contents of the KR.



Options:

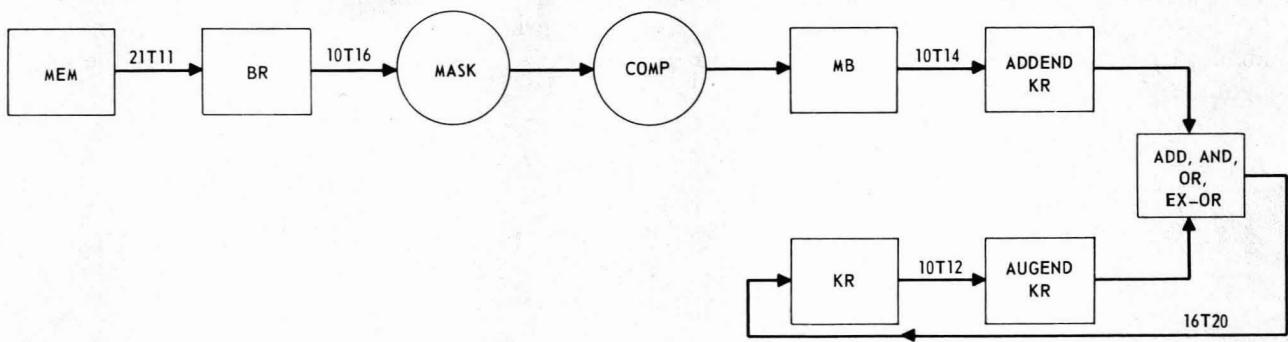
RM

I(0V000;V=Index Reg No.)

LCJ

PL(00004)  
PS(00005)  
S(00001)  
C(10000)

**8.42 PMK(00242)—Logical Product (AND) of Memory with Accumulator:** The contents of location M replace the contents of the BR and, after possible product masking and/or complementing, are combined by the logical product (AND) function with the contents of the KR; the result replaces the original contents of the KR.

**Options:****RM**

I(0V000 ;V = Index Reg No.)  
 A(20000)  
 W(00000 1)  
 S(20000 1)

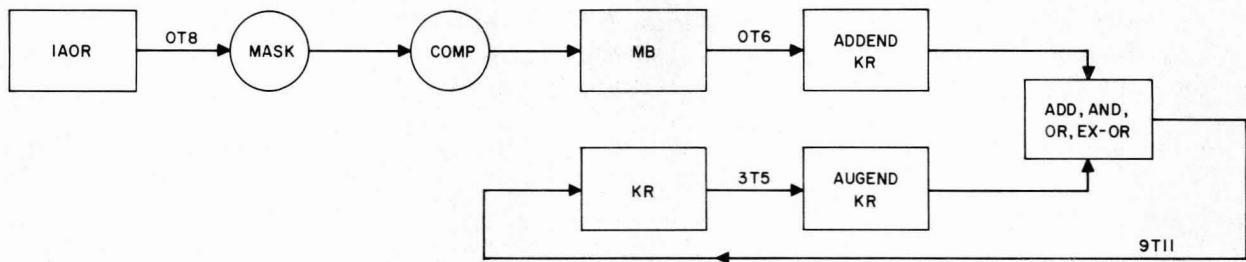
**LCJ**

PL(00000 2)  
 PS(00001)  
 S(20401)  
 C(10000)

**Restrictions:** The following instruction must not

- (1) Specify K in the R subfield
- (2) Be AKR or SKR.

**8.43 UWK(00670)—Logical Union (OR) of Word with Accumulator:** The resultant DAR number W, after possible product masking and/or complementing, is combined by the logical union (OR) function with the contents of the KR, and the result replaces the original contents of the KR.

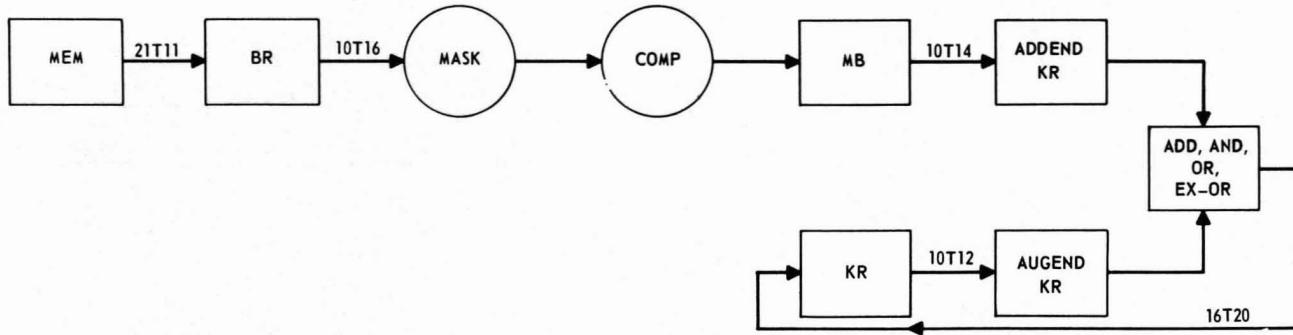
**Options:****RM**

I(0V000 ;V = Index Reg No.)

**LCJ**

PL(00004)  
 PS(00005)  
 S(00001)  
 C(10000)

**8.44 UMK(00270)—Logical Union (OR) of Memory with Accumulator:** The contents of location M replace the contents of the BR and, after possible product masking and/or complementing, are combined by the logical union (OR) function with the contents of the KR; the result replaces the original contents of the KR.



**Options:**

**RM**

I(0V000; V = Index Reg No.)  
 A(20000)  
 W(00000 1)  
 S(20000 1)

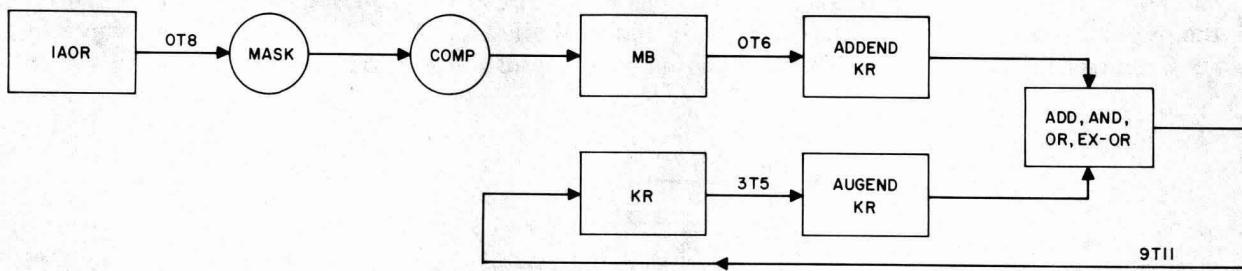
**LCJ**

PL(00000 2)  
 PS(00001)  
 S(20401)  
 C(10000)

**Restrictions:** The following instruction must not

- (1) Specify K in the R subfield
- (2) Be AKR or SKR.

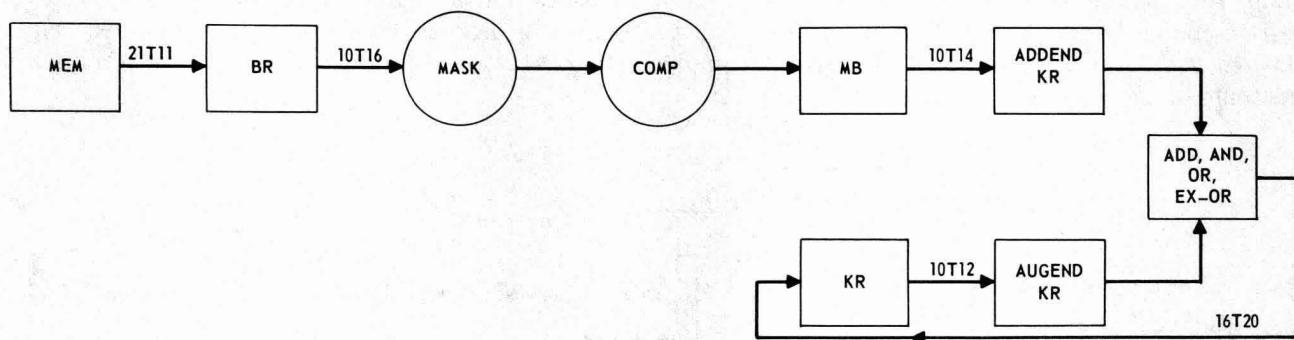
**8.45 XWK(00650)—EXCLUSIVE OR of Word with Accumulator:** The resultant DAR number W, after possible product masking and/or complementing, is combined by the EXCLUSIVE OR function with the contents of the KR; the result replaces the original contents of the KR.

**Options:****RM****LCJ**

I (0V000 ; V = Index Reg No.)

PL(00004)  
 PS(00005)  
 S(00001)  
 C(10000)

**8.46 XMK(00250)—EXCLUSIVE OR of Memory with Accumulator:** The contents of location M replace the contents of the BR and, after possible product masking and/or complementing, are combined by the EXCLUSIVE OR function with the contents of the KR; the result replaces the original contents of the KR.

**Options:****RM****LCJ**

I (0V000 ; V = Index Reg No.)  
 A(20000)  
 W(00000 1)  
 S(20000 1)

PL(00000 2)  
 PS(00001)  
 S(20401)  
 C(10000)

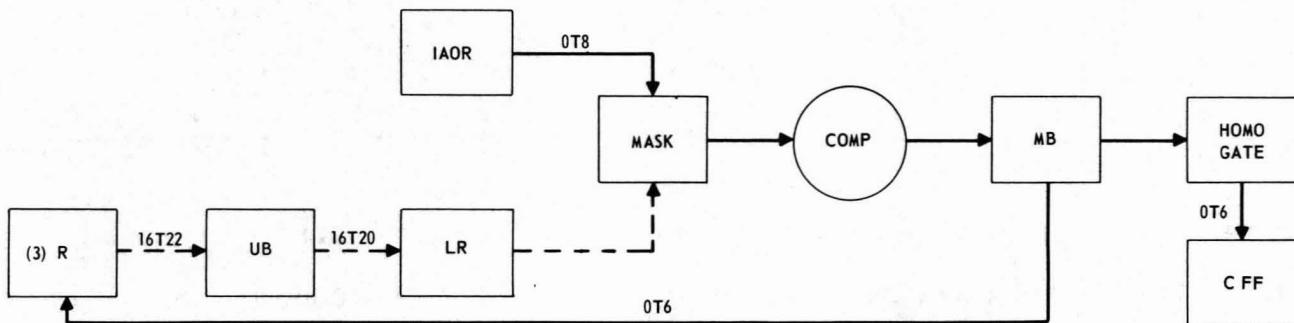
**Restrictions:** The following instruction must not

- (1) Specify K in the R subfield
- (2) Be AKR or SKR.

**8.47 PWX(00716), PWY(00706), and PWZ(00736)—Logical Product (AND) of Word with Register:** The original contents of the XR, YR, or ZR, whichever is specified in the operation code, replace

## SECTION 231-001-102

the contents of the LR. The resultant DAR number W is combined by the logical product (AND) function with the original contents of the specified register now in the LR. The result, after possible complementing, sets the C control flip-flops and replaces the original contents of the specified register.



Options:

RM

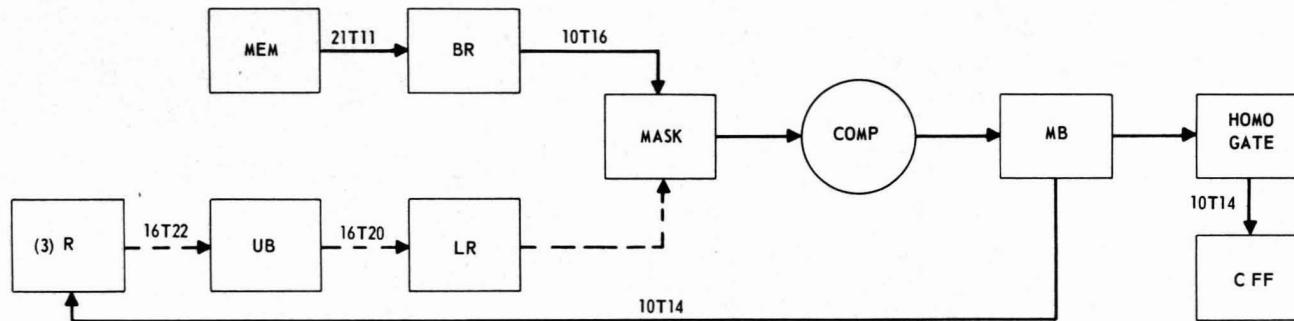
I(0V000; Index Reg No.)

LCJ

C(10000)

### 8.48 PMX(00312 2), PMY(00302 2), and PMZ(00332 2)—Logical Product (AND) of Memory with Register:

**Register:** The original contents of the XR, YR, or ZR, whichever is specified in the operation code, replace the contents of the LR. The contents of location M replace the contents of the BR and are combined by the logical product (AND) function with the original contents of the specified register now in the LR. The result, after possible complementing, sets the C control flip-flops and replaces the contents of the specified register.



Options:

RM

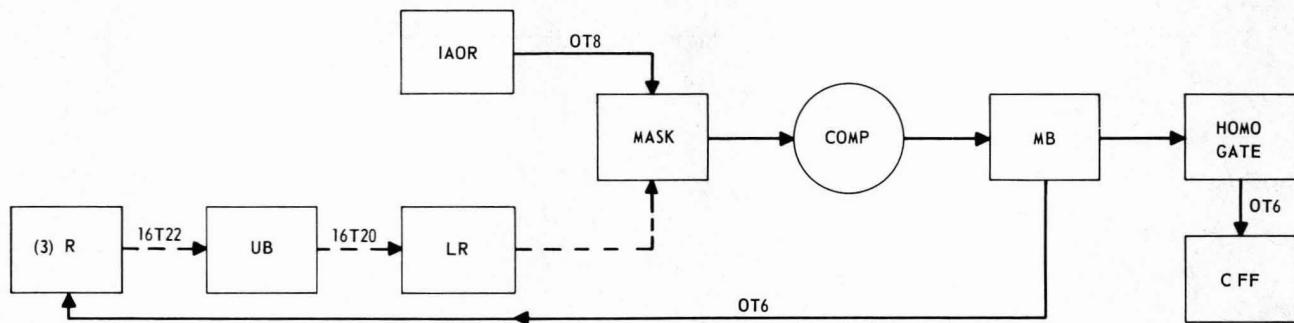
I(0V000; V = Index Reg No.)  
A(20000)  
W(00000 1)  
S(20000 1)

LCJ

C(10000)

**Restrictions:** If the register appearing in the R subfield is the same as the register specified in the operation code, option A, W, or S is not available.

**8.49 *UWX(00712), *UWY(00702), and *UWZ(00732)—Logical Union (OR) of Word with Register:***** The original contents of the XR, YR, or ZR, whichever is specified in the operation code, replace the contents of the LR. The resultant DAR number W is combined by the logical union (OR) function with the original contents of the specified register now in the LR. The result, after possible complementing, sets the C control flip-flops and replaces the contents of the specified register.



**Options:**

**RM**

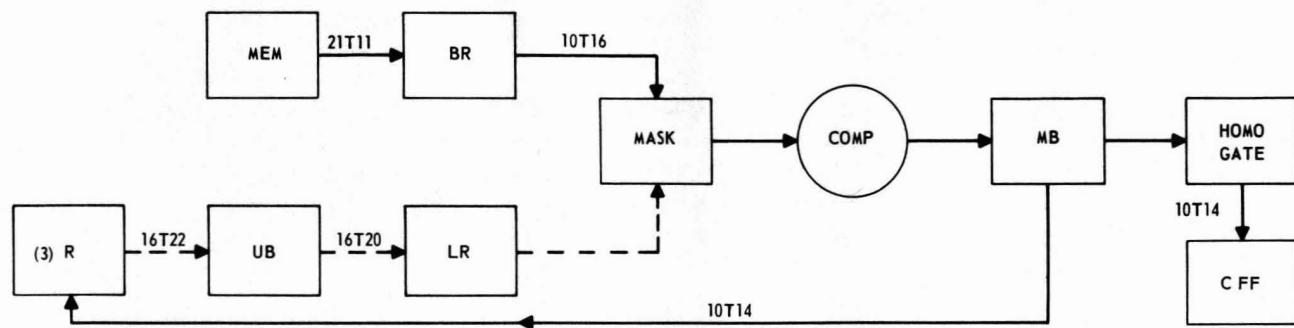
**LCJ**

**I(0V000;V = Index Reg No.)**

**C(10000)**

**8.50 *UMX(00312), *UMY(00302), and *UMZ(00332)—Logical Union (OR) of Memory with Register:*****

The original contents of the XR, YR, or ZR, whichever is specified in the operation code, replace the contents of the LR. The contents of location M replace the contents of the BR and are combined by the logical union (OR) function with the contents of the specified register now in the LR. The result, after possible complementing, sets the C control flip-flops and replaces the contents of the specified register.

**Options:****RM**

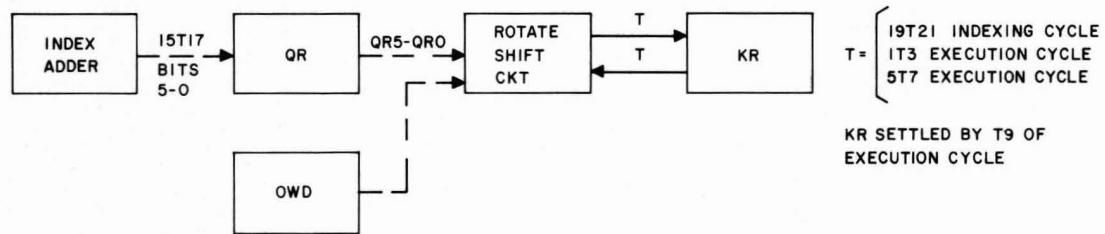
I(0V000;V = Index Reg No.)  
 A(20000)  
 W(00000 1)  
 S(20000 1)

**LCJ**

C(10000)

**Restrictions:** If the register appearing in the R subfield is the same as the register specified in the operation code, option A, W, or S is not available.

**8.51 H(00005 2)—Shift:** The contents of the KR are shifted the number of places specified by the six least significant bits of the resultant DAR number. Bit position 5 (the most significant of the six bits) is treated as the sign bit of the number specified in the remaining five bits. For example, 101111 is as -16; note that the full 23-bit form of -16 is 17 ones followed by 101111. A positive number specifies a shift to the left; a negative number specifies a shift to the right. Bits shifted past position 0 or 22 of the KR are lost. Positions made vacant are filled with zeros. Any number that has an absolute value of 23 through 31, if used, will cause no change in the contents of the KR.

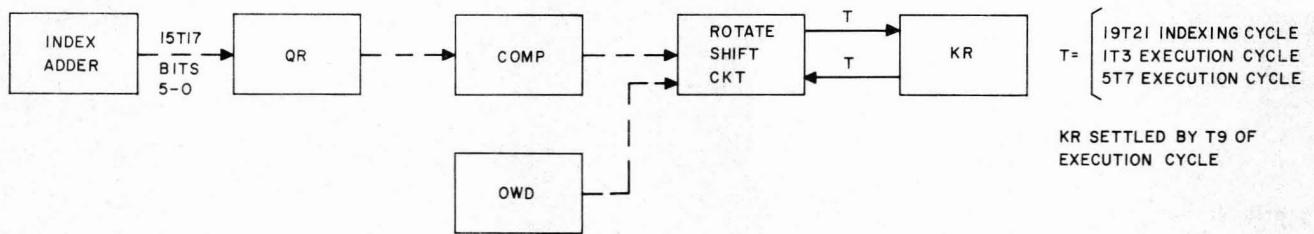
**Options:****RM**

I(0V000;V = Index Reg No.)  
 A(20000)

**Restrictions:** If register K is specified in the R subfield, option A is not available.

**8.52 HC(10005 2)—Shift Complemented:** The contents of the KR are shifted the number of places specified by the complement of the six least significant bits of the resultant DAR number. Bit position 5 (the most significant of the six bits) is treated as the sign bit of the number specified in the

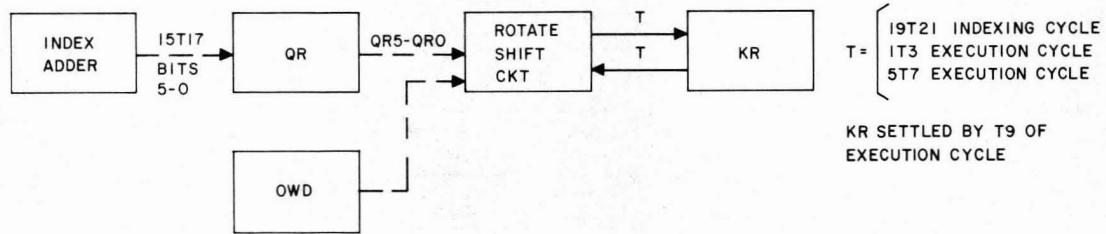
remaining five bits. For example, 101111 is -16 and is complemented to 010000 or +16. The complement of a positive number is negative and specifies a shift to the right; the complement of a negative number is positive and specifies a shift to the left. Bits shifted past position 0 or 22 of the KR are lost. Positions made vacant are filled with zeros. Any number that has an absolute value of 23 through 31, is used, will cause no change in the contents of the KR.

**Options:****RM**

I (0V000;V = Index Reg No.)  
A (20000)

**Restrictions:** If register K is specified in the R subfield, option A is not available.

**8.53 QC(00001 2)—Rotate:** The contents of the KR are rotated the number of places specified by the six least significant bits of the resultant DAR number. Bit position 5 (the most significant of the six bits) is treated as the sign bit of the number specified in the remaining five bits. A positive number specifies a rotation to the left; a negative number specifies a rotation to the right. Bits rotated past position 0 or 22 of the KR enter the opposite end. Any number that has an absolute value of 23 through 31, if used, will cause no change in the contents of the KR.

**Options:****RM**

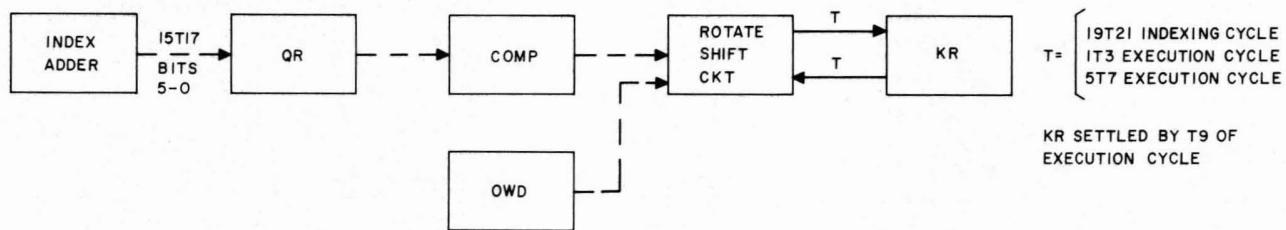
I (0V000;V = Index Reg No.)  
A (20000)

**Restrictions:** If register K is specified in the R subfield, option A is not available.

**8.54 QC(10001 2)—Rotate Complemented:** The contents of the KR are rotated the number of places specified by the complement of the six least significant bits of the resultant DAR number. Bit position 5 (the most significant of the six bits) is treated as the sign bit of the number specified in the remaining five bits. The complement of a positive number specifies a rotation to the right; the complement of a negative number specifies a rotation to the left. Bits rotated past position 0 or 22 of the KR enter

## SECTION 231-001-102

the opposite end. Any number that has an absolute value of 23 through 31, if used, will cause no change in the contents of the KR.

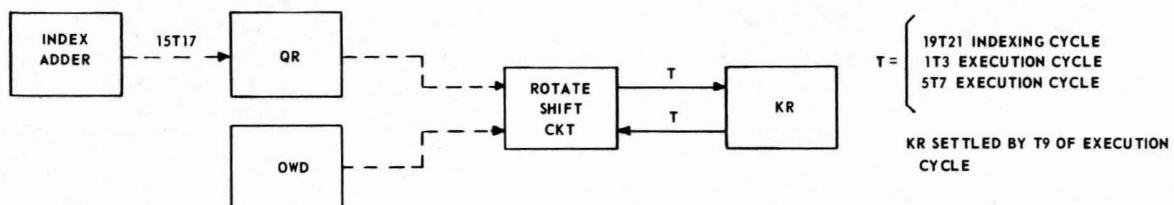


**Options:** RM

I(0V000;V = Index Reg No.)  
A(20000)

**Restrictions:** If register K is specified in the R subfield, option A is not available.

**8.55 QS(00001)—Rotate Sixteen Bits:** The 16 bits of the KR numbered 6 through 21 are rotated the number of places specified by the six least significant bits of the resultant DAR number. Bit position 5 (the most significant of the six bits) is treated as the sign bit of the number specified in the remaining five bits. A positive number specifies a rotation to the left; a negative number specifies a rotation to the right. Bits rotated past position 6 enter position 21; bits rotated past position 21 enter position 6. The right-most six bits, 0 through 5, and bit 22 remain unchanged. Any number that has an absolute value of 16 through 22 will rotate the number of places determined by the absolute value (16 through 22) less 16. For example, with absolute value 22, the bits will rotate six places. Any number that has an absolute value of 23 through 31, if used, will cause no change in the contents of the KR.

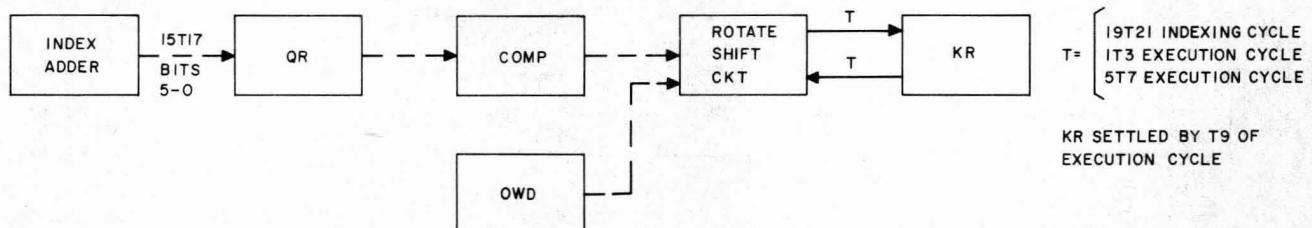


**Options:** RM

I(0V000;V = Index Reg No.)  
A(20000)

**Restrictions:** If register K is specified in the R subfield, option A is not available.

**8.56 QSC(10001)—Rotate Sixteen Bits Complemented:** The 16 KR bits numbered 6 through 21 are rotated the number of places specified by the complement of the six least significant bits of the resultant DAR number. Bit position 5 (the most significant of the six bits) is treated as the sign bit of the number specified in the remaining five bits. The complement of a positive number is negative and specifies a rotation to the right; the complement of a negative number is positive and specifies a rotation to the left. Bits rotated past position 6 enter position 21; bits rotated past position 21 enter position 6. The right-most six bits, 0 through 5, and bit 22 remain unchanged. Any number that has an absolute value of 16 through 22 will rotate the number of places determined by the absolute value (16 through 22) less 16. Any number that has an absolute value of 23 through 31, if used, will cause no change in the contents of the KR.



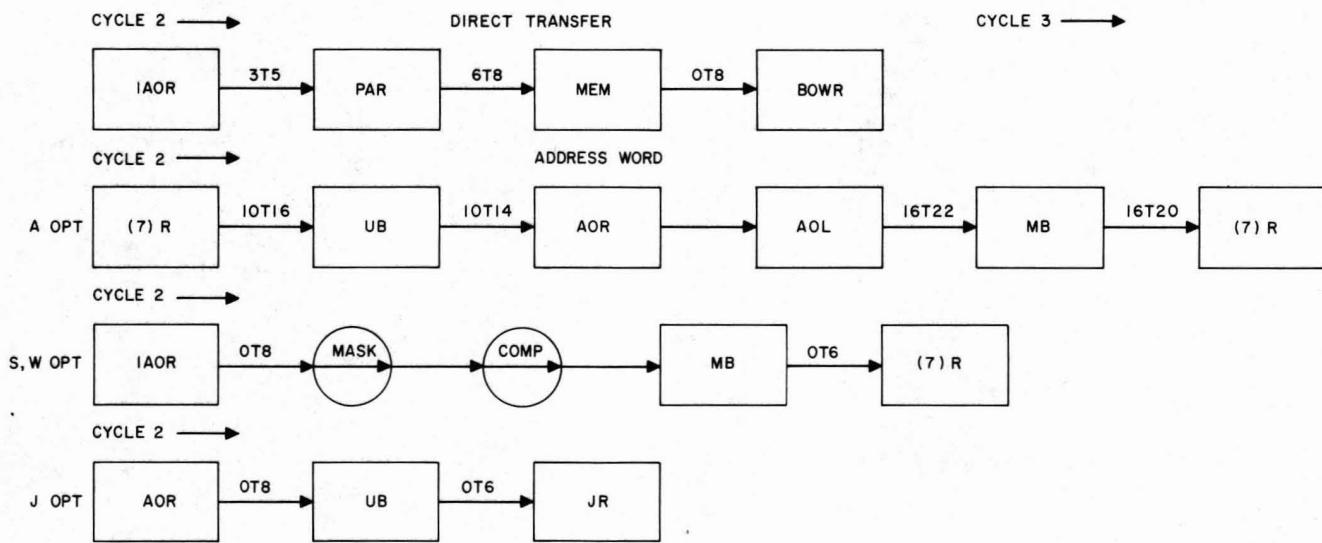
**Options:** RM

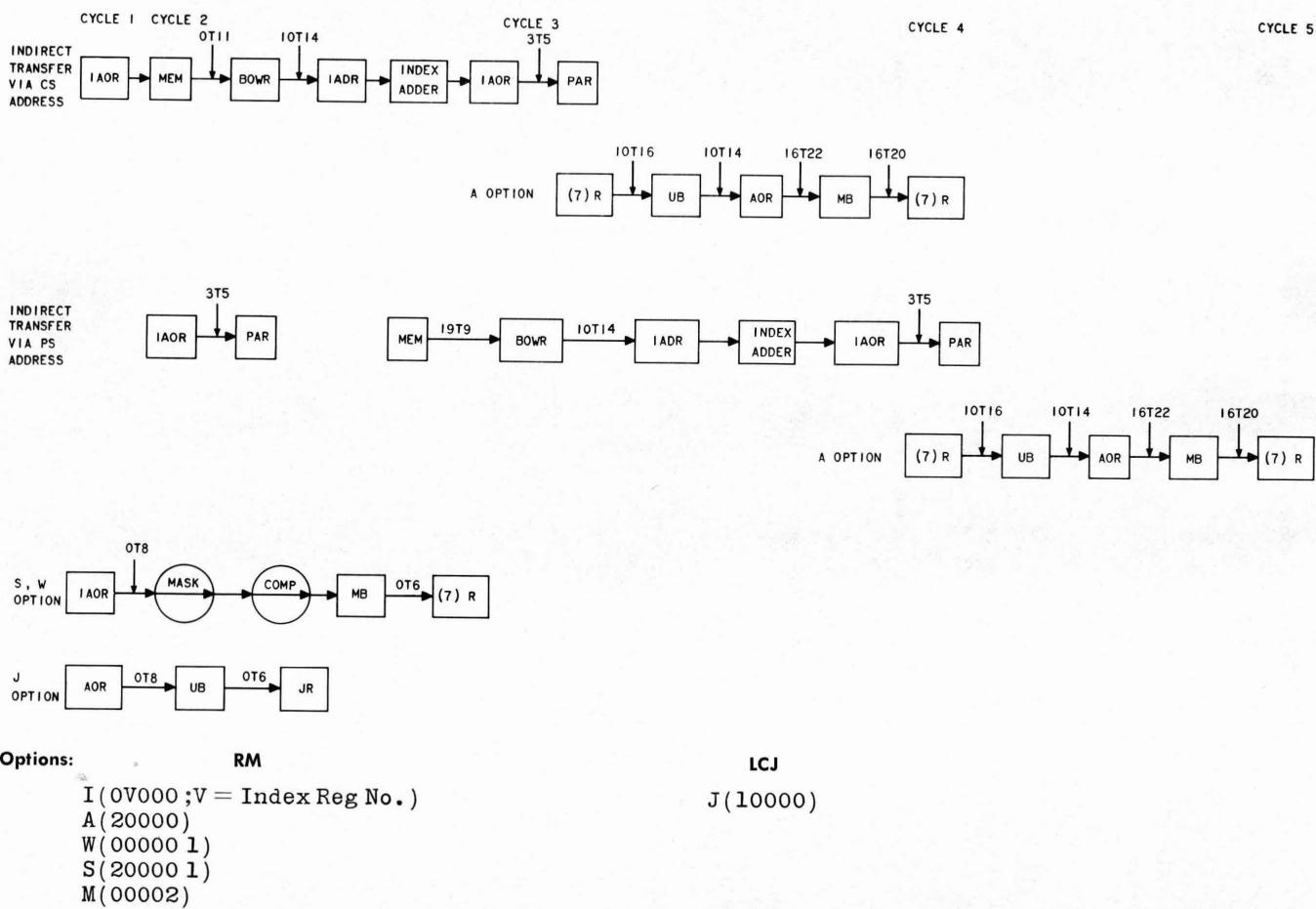
I(0V000;V = Index Reg No.)  
A(20000)

**Restrictions:** If register K is specified in the R subfield, option A is not available.

## F. Transfer Operations

**8.57 T(00010)—Transfer:** Program control is transferred to another instruction. In the case of a direct transfer, specified by an empty M subfield, the address of this instruction is the resultant DAR address. In the case of an indirect transfer, specified by an M in the M subfield, the address of the instruction is contained in a location specified by the resultant DAR address. The contents of bits 0 through 19 of this location contain the address of the instruction to which control is transferred.

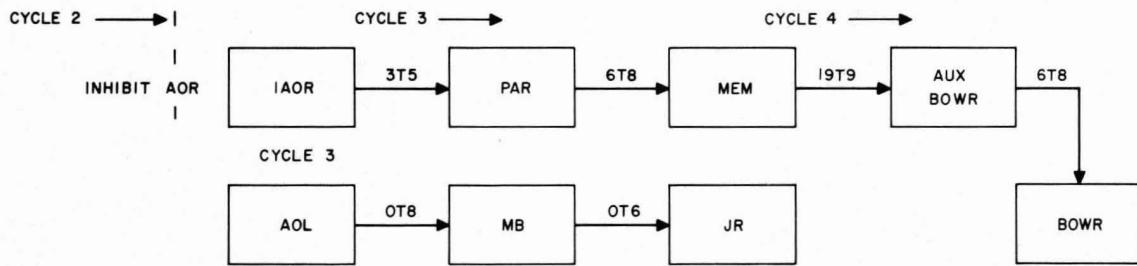




**Note:** If both the J option and A option are specified, the return address placed in J is incremented by one if J is equal I.

**8.58 ENTJ(10004 2)—Execute Next Instruction and Transfer:** First, any indexing and/or register modifications specified by the instruction are performed. Then, the return address (in this case, two plus the address of the ENTJ instruction) is placed in the JR. A transfer of control to the resultant DAR address is begun and, during the transfer time, the next sequential instruction is executed. During the execution of ENTJ and instruction that follows it, interrupts of levels H through K are inhibited. (If ENTJ is followed by a one-cycle instruction, both of these instructions are executed in a total of two cycles.) The instruction that follows ENTJ has the following effects.

- (a) If the sequential instruction immediately following ENTJ is an executed transfer, transfer specified by ENTJ does not occur. The JR is set to 2, plus the address of the ENTJ instruction, unless the executed transfer instruction specifies the store return address option J. When option J is specified, the JR is set to the resultant DAR address given in the ENTJ instruction.
- (b) If a register specified in the ENTJ instruction is changed in the following sequential instruction, ENTJ will use the original contents of the register before the change.
- (c) If the sequential instruction immediately following ENTJ is ENAM, the address stored in memory by ENAM is the resultant DAR address of the ENTJ instruction.



Options:

RM

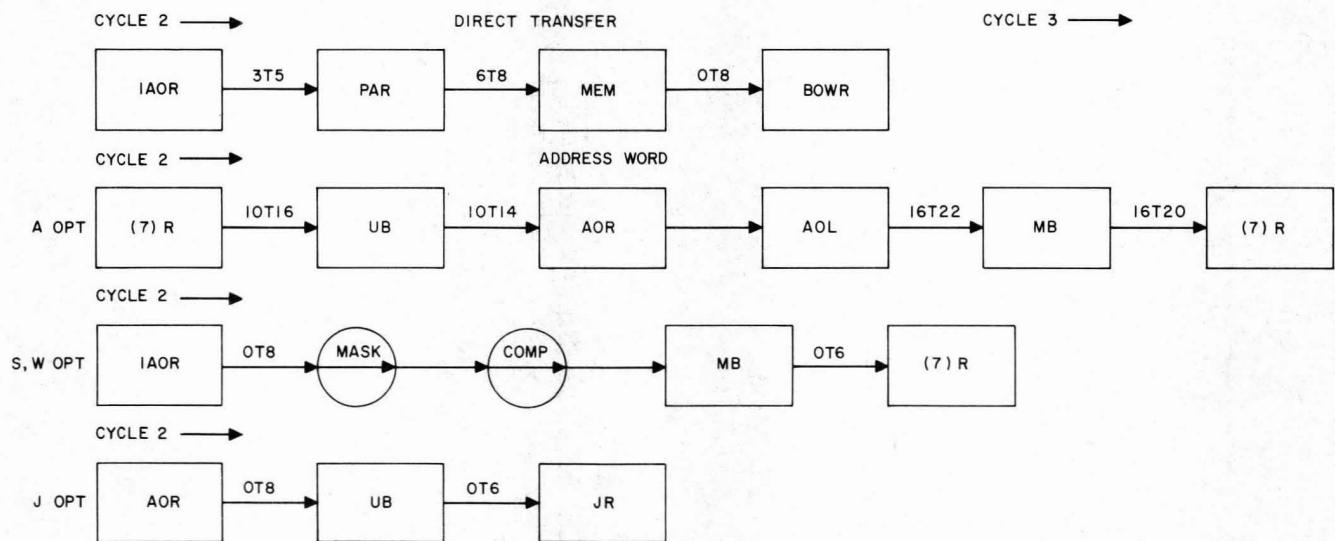
I(0V000;V = Index Reg No.)  
 A(20000)  
 W(00000 1)  
 S(20000 1)

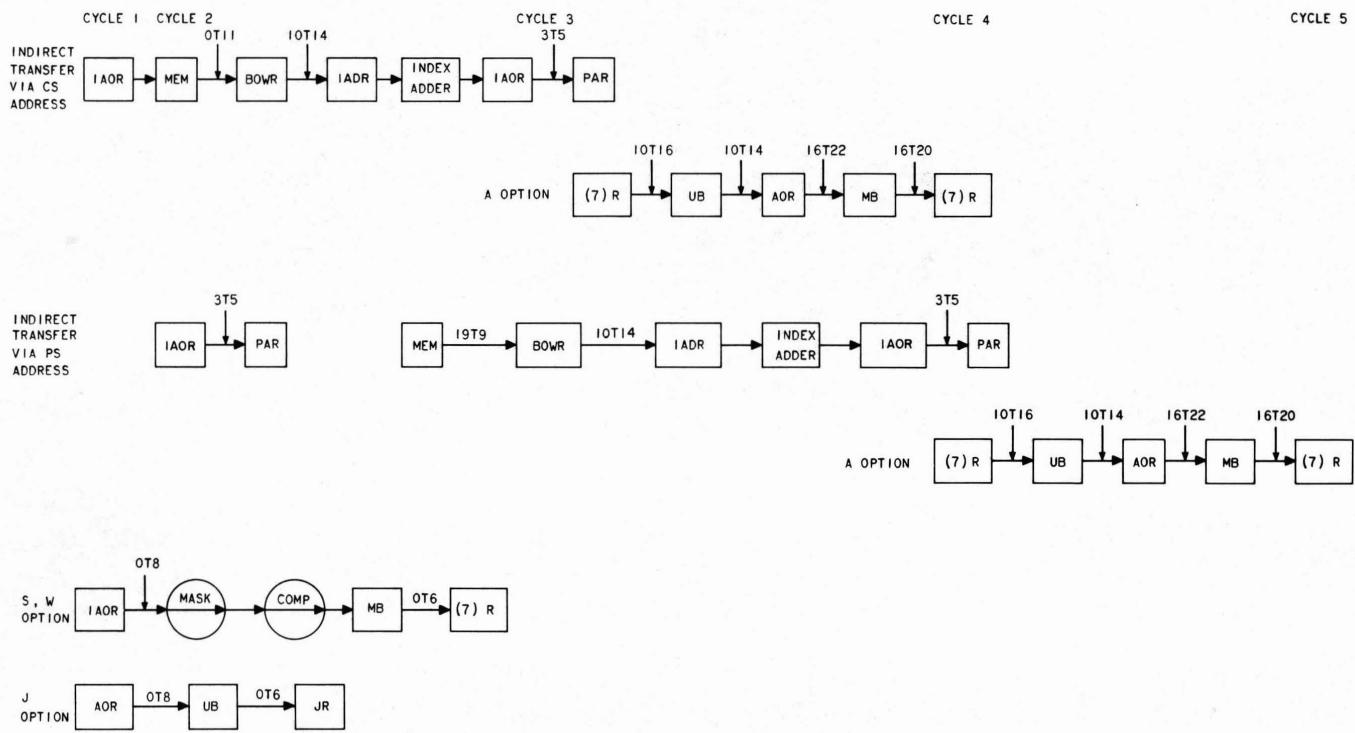
**Restrictions:**

- (1) If the sequential instruction immediately following ENTJ is a transfer (T, TK..., TC..., TR...), the ENTJ instruction must not specify the S (set register) option or indexing option A or W.
- (2) The following sequential instruction must not be ENTJ, MCII, MJ, UMKMJ, AJR, SJR, EGBN, MKII, EXC, or one specifying register J in the R subfield.
- (3) The instruction executed at the address specified by ENTJ is subject to those restrictions resulting from the execution of the sequential instruction immediately following ENTJ.
- (4) Any programs executed from the call store cannot use the instruction ENTJ.
- (5) The instruction executed at the address specified by EXC must not be ENTJ.
- (6) The resultant DAR address of the ENTJ instruction cannot be a call store address.
- (7) The following instruction must not be a memory read or write instruction that changes the register used for indexing ENTJ.

**8.59 TKP(00020), TKM(00025), TKAZ(00024 2), TKAU(00021 2), TKLZ(00024), TKLU(00021), TKLE (00025 2), TKGF(00020 2)—Transfer if KR Condition is P, M, AZ, AU, LZ, LU, LF, or GE:**

If the contents of the KR satisfy the conditions specified by the operation code, program control is transferred to another instruction. In the case of a direct transfer, specified by an empty M subfield, the address of this instruction is the resultant DAR address. In the case of an indirect transfer, specified by an M in the M subfield, the address of the instruction is contained in a location specified by the resultant DAR address. Bits 0 through 19 of this location contain the address of the instruction to which control is transferred.





Options:

RM

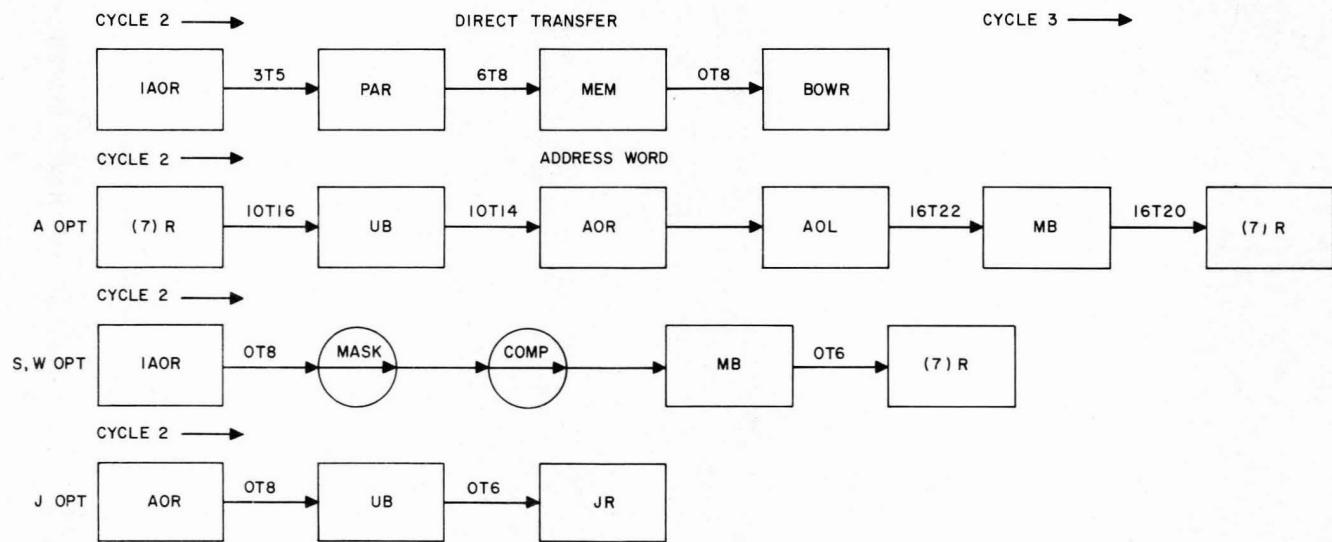
LCJ

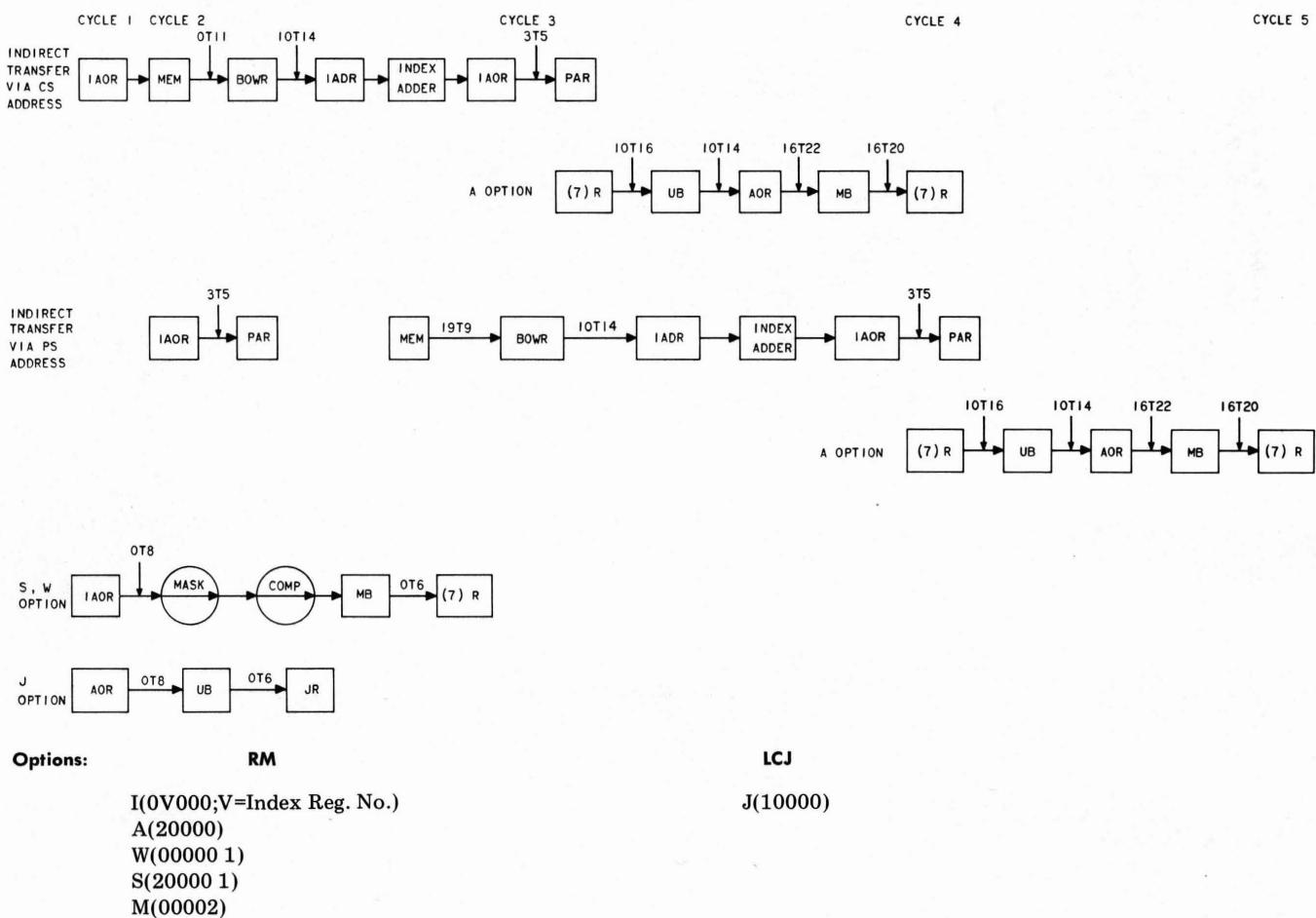
I(0V000;V = Index Reg No.)  
 A(20000)  
 W(00000 1)  
 S(20000 1)  
 M(00002)

J(10000)

**Note:** If both the J option and the A option are specified, the return address placed in J is incremented by 1 if J is equal to I.

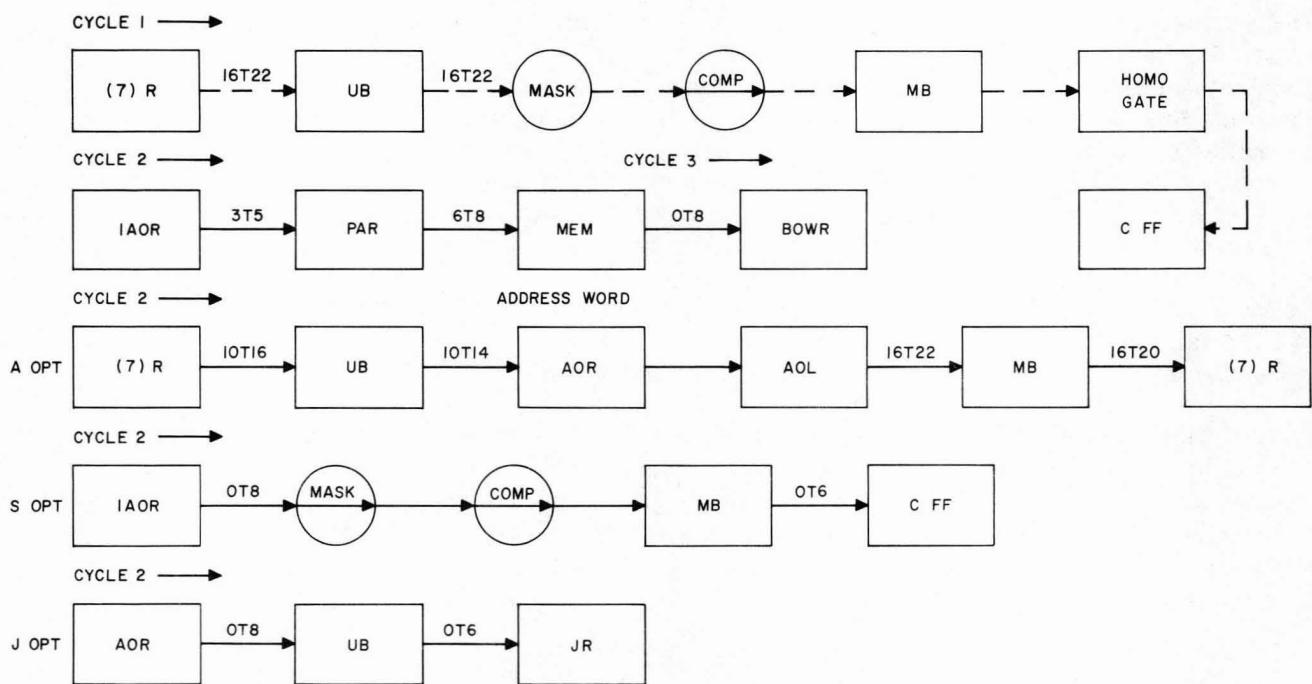
**8.60 *TCP(00030), TCM(00035), TCAZ(00034 2), TCAU(00031 2), TCLZ(00034), TCLU(00031), TCLE (00035 2), TCGE(00030 2)—Transfer if Conditions of C Flip-Flops Are P, M, AZ, AU, LZ, LU, LE, or GE:*** If the C control flip-flops satisfy the conditions indicated in the operation code, program control is transferred to another instruction. In the case of a direct transfer, specified by an empty M subfield, the address of this instruction is the resultant DAR address. In the case of an indirect transfer, specified by an M in the M subfield, the address of the instruction is contained in a location specified by the resultant DAR address. Bits 0 through 19 of this location contain the address of the instruction to which control is transferred.





**Note:** If both the J option and the A option are specified, the return address placed in J is incremented by one if J is equal to I.

**8.61 TRP(00060), TRM(00065), TRAZ(00064 2), TRAU(00061 2), TRLZ(00064), TRLU(00061), TRLE (00065 2), TRGE(00060 2)**—Transfer if Condition of One of Seven Registers is P, M, AZ, AU, LZ, LU, LE, or GE: The C control flip-flops are set to correspond to the contents, possibly PL masked, of the register specified in the R subfield. If the conditions indicated in the operation code are satisfied, program control is transferred to another instruction. In the case of a direct transfer, specified by an empty M subfield, the address of this instruction is the resultant DAR address. In the case of an indirect transfer, specified by an M in the M subfield, the address of the instruction is contained in a location specified by the resultant DAR address. Bits 0 through 19 of this location contain the address of the instruction to which control is transferred.



**Options:**

**RM**

I(0V000;V=Index Reg No.)  
A(20000)  
S(20000 1)  
M(00002)

**LCJ**

PL(00010)  
J(10000)

**Note:** If both the J option and the A option are specified, the return address placed in J is incremented by 1 if J is equal to I.

#### G. Combined Shift and Rotate Operations

**8.62** The operation code of instructions in this category begins with either the letter H or Q. These letters mean that immediately after the indexing of the instruction, the contents of the KR will either be shifted (H) or rotated (Q) before performing the move, add, subtract, compare, or logical

## SECTION 231-001-102

operation specified by the latter part of the operation code. There are two types of shift and rotate combinations: standard and special.

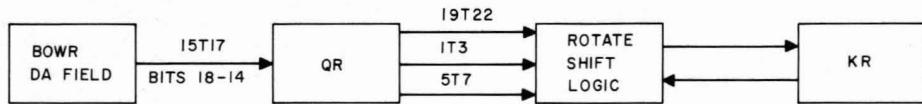
### Standard Shift and Rotate Combinations

**8.63** The variable field of these instructions consists of four subfields: HQ, A, R, and LCJ. The address of location M is determined according to the contents of the A field. After possible indexing, the contents of the KR are shifted or rotated according to the first letter (H or Q) of the instruction and the contents of the HQ subfield.

**8.64** If the first letter of the operation code is H, the contents of the KR are shifted the number of places specified by the contents of the HQ subfield. A positive number specifies a shift to the left; a negative number specifies a shift to the right. Bits shifted past position 0 or 22 of the KR are lost. Positions made vacant are filled with zeros. If the first letter of the operation code is Q, the contents of the KR are rotated the number of places determined by the contents of HQ subfield. A positive number specifies a rotation to the left by the number of places given. A negative number specifies a rotation to the left by the number of places determined by the sum of 23 and the negative number. The result of the latter rotation is the same as if a negative number caused a rotation to the right by its absolute value. Bits rotated past position 22 of the KR enter the opposite end. Numbers whose absolute value is 23 through 31 must not be used.

**8.65** The A subfield of the DA field can be blank or can specify an assigned number of the absolute value not greater than 16,383 (that is, a number of 15 bits including the sign in bit 20). This is the effective DA number which becomes the resultant DAR address after possible indexing.

**8.66** The remainder of the instruction is executed according to the general purpose instruction specified by the letters following the H or Q of the operation code. The flow for the shift and rotate portion of the instructions is shown below. This diagram applies to all standard shift and rotate instructions. For the complete flow diagram, add the one below to the diagram for the instruction without shifting or rotating.



### 8.67 HBM(20142 1), QBM(20142)

**Options:** RM

I(0V000;V=Index Reg No.)

### 8.68 HLM(20140 1), QLM(20140)

**Options:** RM

I(0V000;V=Index Reg No.)

LCJ

EL(00000 2)  
C(10000)

8.69 *HFM(20150 1), QFM(20150), HJM(20160 1), QJM(20160), HXM(20162 1), QXM(20162), HYM(20170 1), QYM(20170), HZM(20172 1), QZM(20172)*

<b>Options:</b>	<b>RM</b>	<b>LCJ</b>
I (0V000 ;V = Index Reg No.)		EL(00000 2) PL(00004 2) C(10000)

8.70 *HMB(20602 1), QMB(20602), HMBCS(20630 1), QMBCS(20630)*

<b>Options:</b>	<b>RM</b>
I (0V000 ;V = Index Reg No.)	

8.71 *HMC(20610 1), QMC(20610)*

<b>Options:</b>	<b>RM</b>	<b>LCJ</b>
I (0V000 ;V = Index Reg No.)		PL(00000 2) C(10000)

8.72 *HMCII(20612 1), QMCII(20612)*

<b>Options:</b>	<b>RM</b>	<b>LCJ</b>
I (0V000 ;V = Index Reg No.)		PL(00000 2) C(10000)

**Restrictions:** Register B must not be used as an index register because it is destroyed on the first execution of either HMCII or QMCII.

8.73 *HML(20720 1), QML(20720)*

<b>Options:</b>	<b>RM</b>	<b>LCJ</b>
I (0V000 ;V = Index Reg No.)		C(10000)

**SECTION 231-001-102**

**8.74 HMF(20622 1), QMF(20622), HHJ(20722 1), QMJ(20722), HMX(20710 1), QMX(20710), HMY(20700 1), QMY(20700), HMZ(20730 1), QMZ(20730)**

<b>Options:</b>	<b>RM</b>	<b>LCJ</b>
	I(0V000 ;V = Index Reg No.)	PL(00000 2) C(10000)

**8.75 HAMK(20752 1), QAMK(20752), HSMK(30752 1), QSMK(30752), HPMK(20642 1), QPMK(20642), HUMK(20670 1), QUMK(20670), HXMK(20650 1), QXMK(20650)**

<b>Options:</b>	<b>RM</b>	<b>LCJ</b>
	I(0V000 ;V = Index Reg No.)	PL(00000 2) *C(10000)

\*For HSMK and QSMK, the C option has the octal representation of 00000.

**Restrictions:** The following instruction must not

- (1) Specify K in the R subfield
- (2) Be AKR or SKR.

**8.76 HCMK(20742 1), QCMK(20742), HCMKU(20740 1), QCMKU(20740)**

<b>Options:</b>	<b>RM</b>	<b>LCJ</b>
	I(0V000 ;V = Index Reg No.)	PL(00000 2) C(10000)

**Restrictions:** The following instruction must not be one of the early transfer instructions: TAULM, TAUMK, TUPMK, TCGMX, TCMMF.

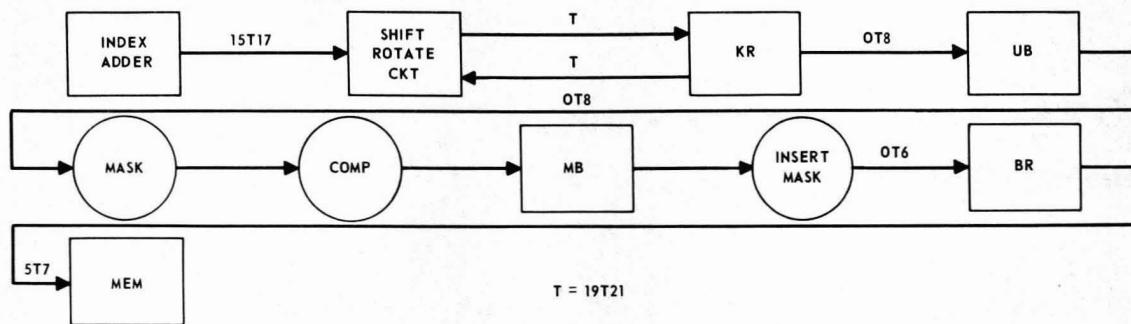
**8.77 HPMX(20712 3), QPMX(20712 2), HMPY(20702 3), QPMY(20702 2), HPMZ(20732 3), QPMZ(20732 2), HUMX(20712 1), QUMX(20712), HUMY(20702 1), QUMY(20702), HUMZ(20732 1), QUMZ(20732)**

<b>Options:</b>	<b>RM</b>	<b>LCJ</b>
	I(0V000 ;V = Index Reg No.)	C(10000)

**Special Shift and Rotate Combinations**

**8.78** In these instructions only a rotation of one place to the left or a shift of one place to the left or right is allowed. No HQ subfield is needed for rotation and for shift instructions; the left or right choice is specified in the operation code. The A subfield is the same as in 8.65.

**8.79 HOKM(20152 1):** First the resultant DAR number determines the address of location M after which the contents of the KR are shifted to the left one place and bit position 0 is made 0. Then the new contents of the KR, after possible product masking and/or complementing, replace or are insertion masked into the contents of the BR, and the new contents of the BR replace the contents of location M.

**Options:****RM****LCJ**

I (0V000 ;V = Index Reg No.)

EL(00000 2)

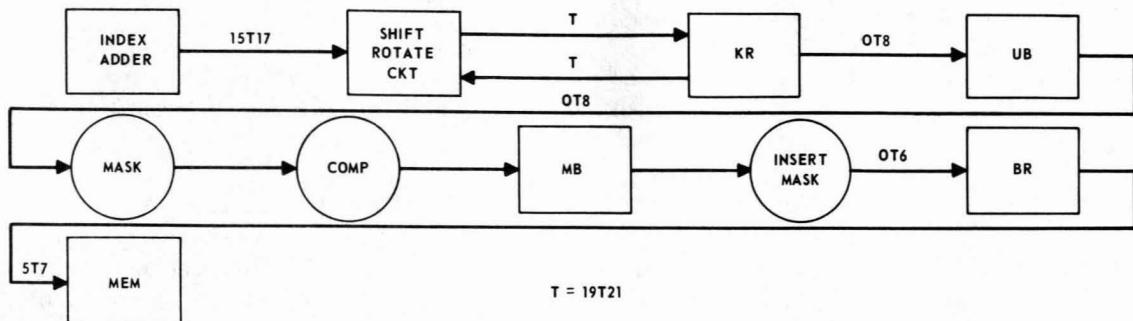
PL(00004 2)

C(10000)

**Note:** Bit 14 is 1.

**8.80 HCOKM(20152 1):** First the resultant DAR number determines the address of location M after which the contents of the KR are shifted to the right one place and bit position 22 is made 0. Then the new contents of the KR, after possible product masking and/or complementing, replace or are insertion masked into the contents of the BR, and the new contents of the BR replace the contents of location M.

## SECTION 231-001-102



Options:

RM

LCJ

I(0V000 ;V = Index Reg No.)

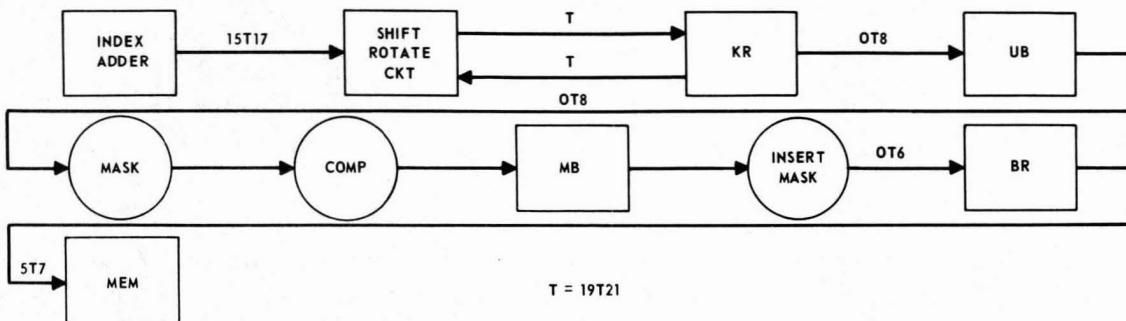
EL(00000 2)

PL(00004 2)

C(10000)

Note: Bits 19 and 14 are 1.

**8.81 QOKM(20152):** First the resultant DAR number determines the address of location M after which the contents of the KR are rotated to the left one place. Then the new contents of the KR, after possible product masking and/or complementing, replace or are insertion masked into the contents of the BR, and the new contents of the BR replace the contents of location M.



Options:

RM

LCJ

I(0V000 ;V = Index Reg No.)

EL(00000 2)

PL(00004 2)

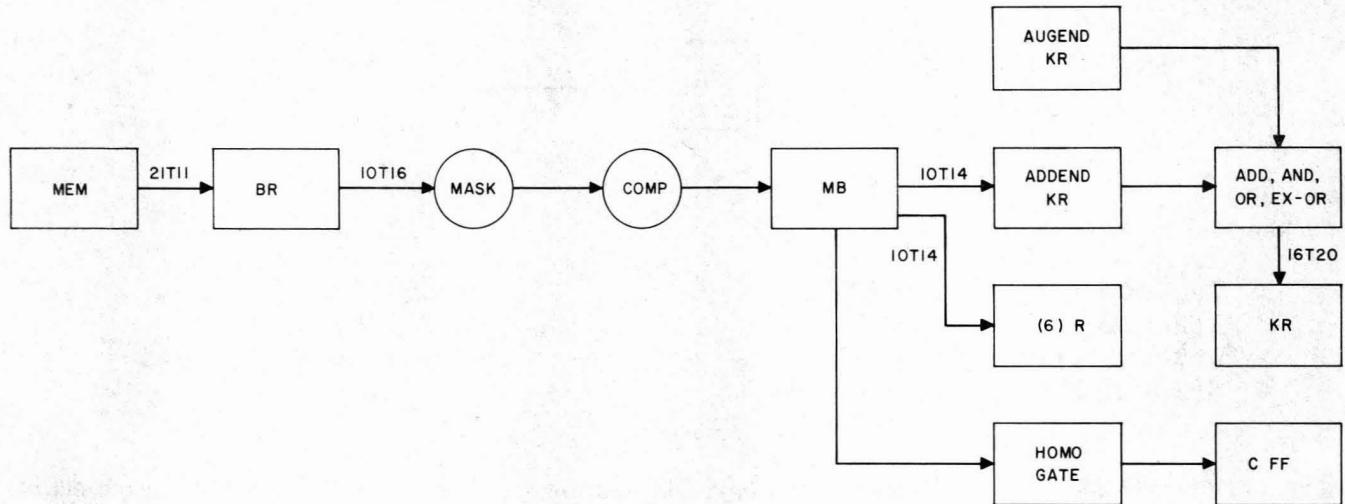
C(10000)

Note: Bits 19 and 14 are 1.

## H. Double Destination Operations

**8.82 MFMK(20622), MJMK(20722), MLMK(20720), MXMK(20710), MYMK(20700), MZMK(20730):** The contents of location M replace the contents of the B register and, after possible product masking and/or complementing, set the C control flip-flops and replace the contents of both the K register and register F, J, L, X, Y or Z, whichever is specified in the operation code. The effective DA number can

be blank or can specify a signed number of absolute value not greater than octal 37777 (that is, a number of 15 bits including the sign bit).



**Note:** Bits 18 and 17 are 1.

**Options:** RM

LCJ

I(0V000;V = Index Reg No.)

\*PL(00000 2)  
C(10000)

\*If the instruction is MLMK, the PL masking option is not available.

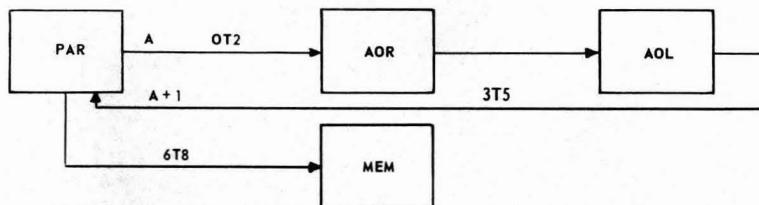
**Note:** Bits 18 and 17 are 1.

**Restrictions:** The following instruction must not

- (1) Be AKR or SKR
- (2) Specify K in the R subfield.

#### I. Miscellaneous Operations

**8.83 EE(00000)—NO-OP:** This instruction is a blank instruction (all zeros) which consumes one cycle of time without doing anything.



## SECTION 231-001-102

**8.84 EZEM(00042)**—Plus zero or if complementing is called for, minus zero replaces or is insertion masked into the contents of the B register. The new contents of B replace the contents of location M.



Options:

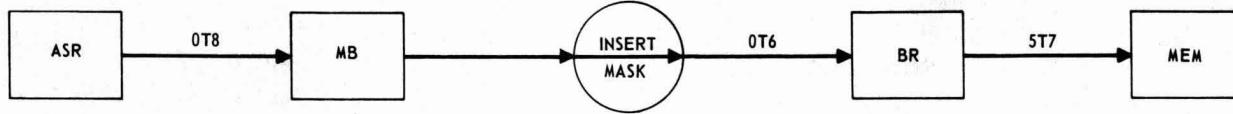
RM

LCJ

I(0V000 ;V = Index Reg No.)  
A(20000)  
W(00000 1)  
S(20000 1)

EL(00100 2)  
ES(00101)  
C(10000)

**8.85 ENAM(00046)—Next Address to Memory:** The address of the next instruction in sequence replaces the contents of the BR and location M.

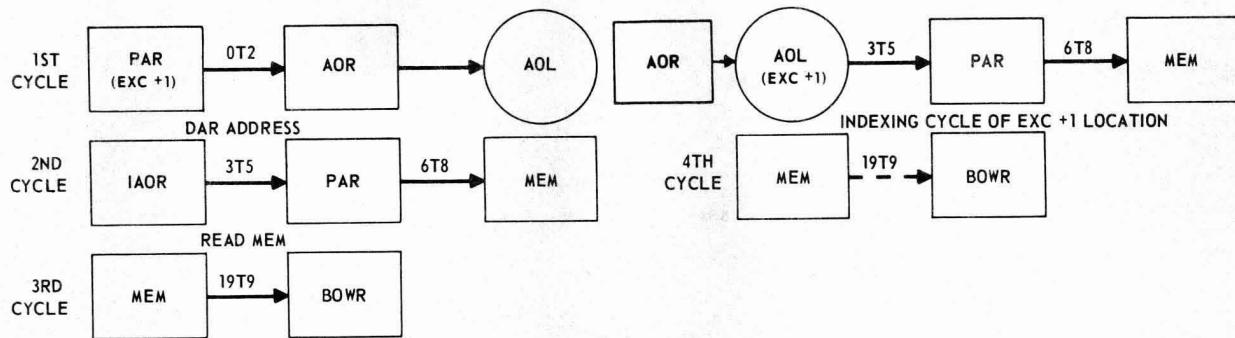


Options:

RM

I(0V000 ;V = Index Reg No.)  
A(20000)  
W(00000 1)  
S(20000 1)

**8.86 EXC(00002)—Execute:** This instruction calls for execution of the PS instruction specified by the resultant DAR address. During execution all interrupts are inhibited. If this execution does not result in a transfer, program control is returned to the instruction following EXC. If the instruction executed does result in a transfer, program control is transferred to the instruction to which the transfer is made. If the instruction executed is a transfer specifying the store return address option J and the transfer occurs, the JR will be set to the address of the instruction following EXC.



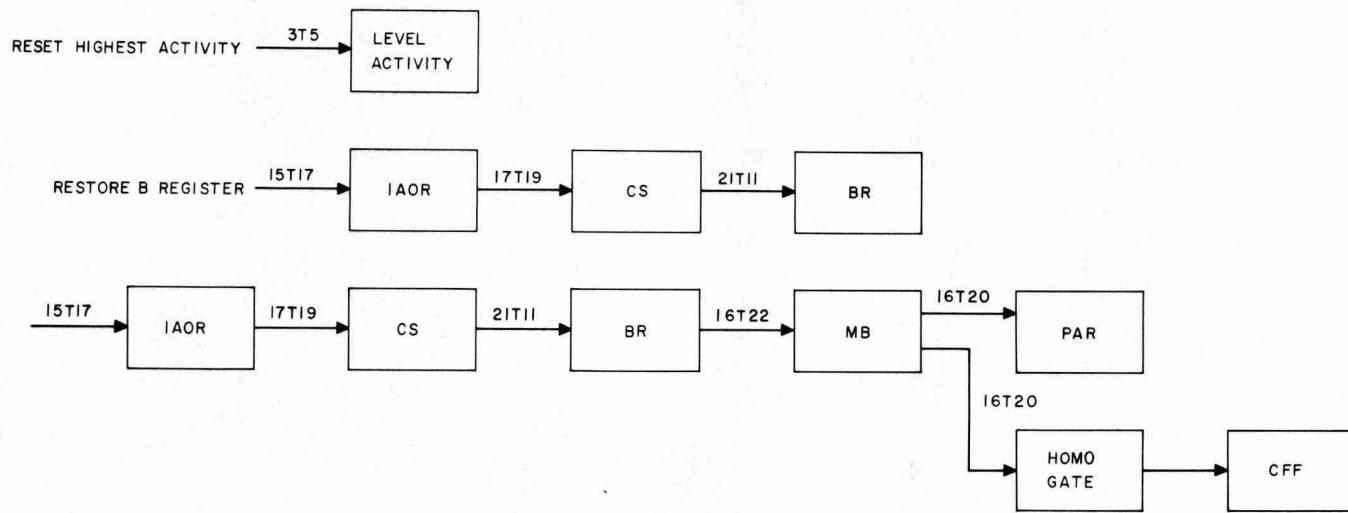
**Options:** RM

I(0V000 ;V = Index Reg No.)  
 A(20000)  
 W(00000 1)  
 S(20000 1)

**Restrictions:** The executed instruction must not be

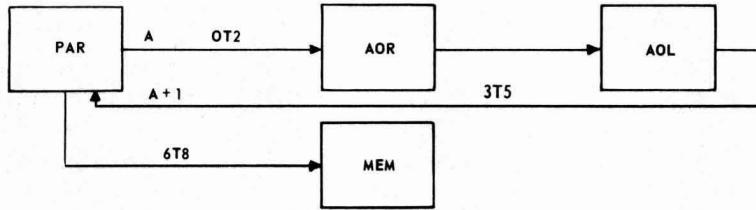
- (1) ENTJ
- (2) One that changes the register used for indexing the EXC instruction
- (3) MCII, MKII, HMCII, or QMCII
- (4) In the call store.

**8.87 EGBN(00560)—Go Back to Normal:** A flip-flop in the level activity register is set to 1 upon the occurrence of an interrupt level corresponding to that flip-flop. In addition, before transferring to the appropriate program, the interrupt circuitry stores the contents of the BR, the state of the C control flip-flops, and the interrupted program's return address. This information is stored in two adjacent CS locations. The addresses of these instructions are permanently associated with the particular interrupt level. When the instruction EGBN is executed, the highest level flip-flop that is set to 1 in the level activity register is reset to 0. The sequence circuitry restores the contents of the BR and the C control flip-flops and returns control to the interrupted program's return address.



**Restrictions:** The preceding instruction cannot be one that extracts information from the PS.

**8.88 EEF(00004)—NO-OP:** The EEF instruction causes a level A interrupt resulting in a transfer to the interrupt program address associated with level A. The interrupt can be inhibited by a manually operated console switch; in which case, each EEF instruction is executed as a NO-OP (EE). The contents of the DA field has no effect on the execution of the instruction but may be used for debugging purposes by programs which read it as 20-bit data.



#### INPUT-OUTPUT INSTRUCTIONS

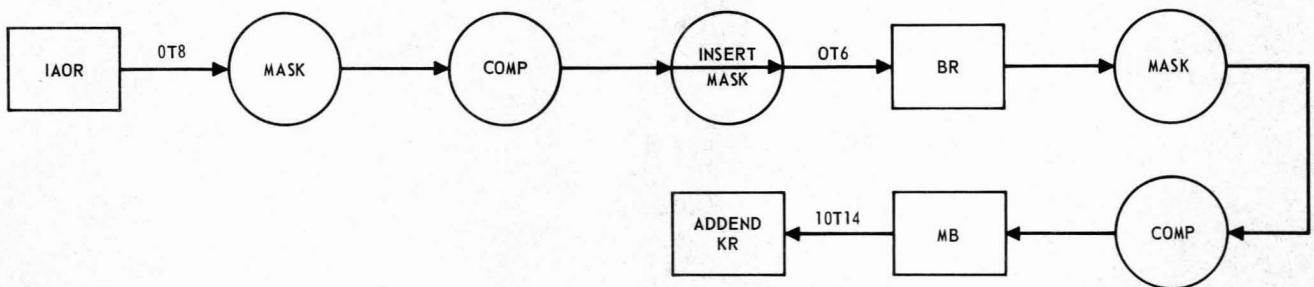
**8.89** This class of instructions consists of eight instructions. The first letter of the operation code is M or W. The second letter is A, S, or D and has the following significance:

- A:** Specifies peripheral unit address bus or central pulse distributor (CPD) address bus in certain operations that send information to a peripheral unit.
- S:** Specifies that an address be sent to the scanner via the peripheral unit address bus.
- D:** Specifies that the CPD be used in selecting and enabling peripheral equipment and for high-speed control of certain flip-flops.

The letter S appearing as the third letter of the operation code means that a reading is expected on the scanner answer bus to the LR from a peripheral unit capable of sending data to the CC. The letter F appearing as the last letter of the operation code means that the enable address is set up in the F

register and sent out to the CPD on the same instruction. This eliminates the necessity for presetting register F by a previous instruction as must be done on instructions MA, WA, MAS, and WAS.

**8.90 WA(00760)—Word to Peripheral Unit Address Bus:** The resultant DAR number W replaces the contents of the BR and then, after possible product masking and/or complementing, replaces the contents of the addend KR. Also according to the contents of bits 9, 8, and 7 of the FR, the resultant DAR number is sent either directly or through specified translation circuitry onto the peripheral unit address bus. If bits 9, 8, and 7 are 101, the resultant DAR number is sent directly onto the CPD address bus. Bit configurations are specified in 8.94. This instruction does not affect the C control flip-flops.

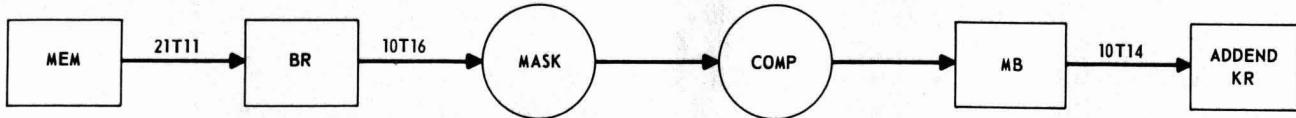


**Restrictions:** On instruction following WA:

- (1) The following instruction must not be an input-output instruction.
- (2) Refer to K (accumulator) restrictions (8.99 and 8.100).
- (3) Refer to F register restrictions (8.101 and 8.102).
- (4) If bit 6 of F is a 1, refer to Y register restrictions (8.103 and 8.104).

**8.91 MA(00360)—Memory to Peripheral Unit Address Bus:** The contents of location M, after possible masking and/or complementing, replace the contents of the addend KR. Also according to the contents of bits 9, 8, and 7 of the FR, the same data is sent either directly or through specified translation circuitry onto the peripheral unit address bus. If these bits are 101, the same data is sent directly onto the CPD address bus. Bit configurations are specified in 8.94. This instruction does not affect the C control flip-flops.

## SECTION 231-001-102



**Options:**

**RM**

I(0V000 ;V = Index Reg No.)  
 A(20000)  
 W(00000 1)  
 S(20000 1)

**LCJ**

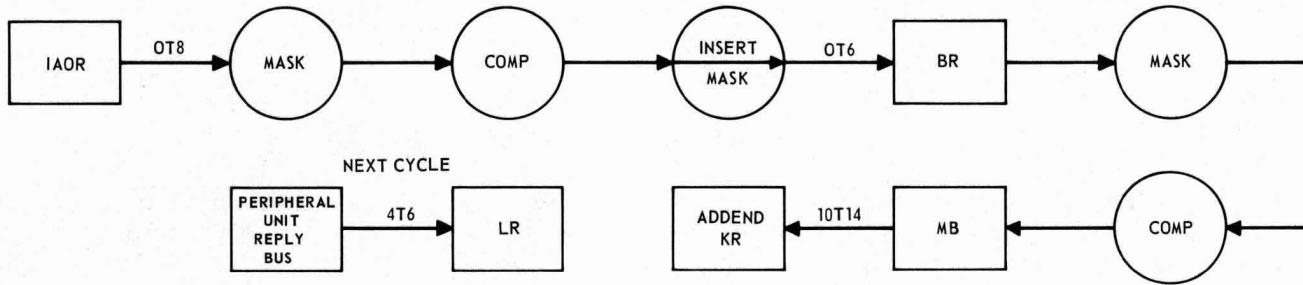
PL(00000 2)  
 PS(00001)  
 S(20401)  
 C(10000)

**Restrictions:** On instructions following MA:

- (1) The following instruction must not be an input-output instruction.
- (2) Refer to K (accumulator) restrictions (8.99 and 8.100).
- (3) Refer to F register restrictions (8.101 and 8.102).
- (4) If bit 6 of F is a 1, refer to Y register restrictions (8.103 and 8.104).

### 8.92 WAS(00762)—Word to Peripheral Unit Address Bus and Response from Scanner Answer Bus:

The same CC actions occur in the execution of this instruction as in the execution of the instruction WA. In addition, the contents of the LR are cleared (made all zeros) after which (during a period of time extending beyond the cycle time of this instruction) bits 0 through 15 accept the information of the scanner answer bus from the peripheral unit. If a scanner all-seems-well failure occurs, bit 22 of the LR is set to a 1. The WAS instruction does not affect the C control flip-flops. The bit configurations for WAS are the same as those for WA covered in 8.94.



**Options:**

**RM**

I(0V000 ;V = Index Reg No.)

**LCJ**

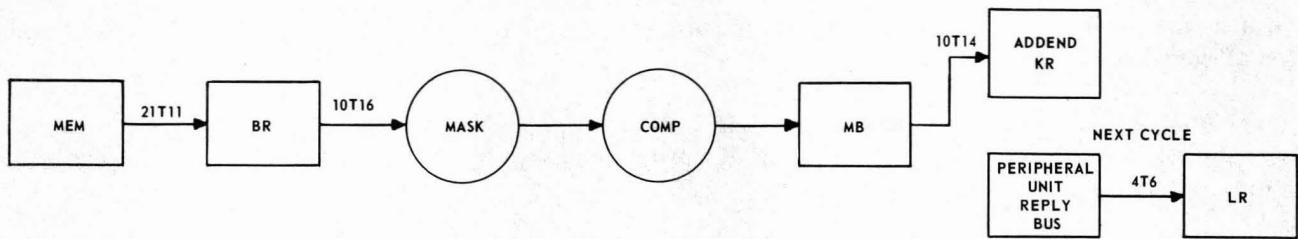
PL(00004)  
 PS(00005)  
 C(10000)

**Restrictions:** On instructions following WAS:

- (1) The following instruction must not be an input-output instruction.
- (2) K (accumulator) restrictions (8.99 and 8.100).
- (3) F register restrictions (8.101 and 8.102).
- (4) If bit 6 of FR is a 1, Y restrictions (8.103 and 8.104).
- (5) LR restrictions (8.105 and 8.106).

**8.93 MAS(00362)—Memory to Peripheral Unit Address Bus and Response from Scanner Answer Bus:**

The same CC actions occur in the execution of this instruction as in the execution of the instruction MA. In addition, the contents of the LR are cleared (made all zeros) after which (during a period of time extending beyond the cycle time of this instruction) bits 0 through 15 accept the information of the scanner answer bus from the peripheral unit. If a scanner all-seems-well failure occurs, bit 22 of the LR is set. The MAS instruction does not affect the C control flip-flops. The bit configurations for MAS are the same as those for MA covered in 8.94.



**Options:**

**RM**

I(0V000 ;V = Index Reg No.)  
A(20000)  
W(00000 1)  
S(20000 1)

**LCJ**

PL(00000 2)  
PS(00001)  
C(10000)

**Restrictions:** On instructions following MAS:

- (1) The following instruction must not be an input-output instruction.
- (2) K (accumulator) restrictions (8.99 and 8.100).
- (3) F register restrictions (8.101 and 8.102).
- (4) If bit 6 of FR is a 1, Y restrictions (8.103 and 8.104).
- (5) LR restrictions (8.105 and 8.106).

**8.94 WA, MA, WAS, and MAS Bit Configurations:** If bit positions 9, 8, and 7 of the FR specify any of the seven binary codes other than 101, then bits 10, 11, 12, and 13 of the FR control the execute translators in selecting the correct CPD. A particular point within a CPD is chosen by bits 14, 15, and 16 of the FR that made a vertical range selection and bits 17, 18, 19, and 20, 21, and 22 that make a horizontal range selection, thereby enabling the peripheral unit. A controller within the peripheral unit and a peripheral unit address bus are determined by using bits 10 and 14 of the FR as follows:

FR BIT 10	FR BIT 14	CONTROLLER	PERIPHERAL UNIT ADDRESS BUS
0	0	0	0
0	1	1	1
1	0	1	0
1	1	0	1

If bit position 6 of the FR contains a 1, the contents of the Y index register (YR) are cleared and the YR accepts an enable verify word. This word is a response via the CPD from the peripheral unit actually addressed. If the responding unit is not the unit addressed, an all-seems-well scanner check (wired) will cause a level F interrupt. If bit position 5 of the FR contains a 1, a wired check is made. If an all-seems-well signal is not received as the result of a wired check, an improper address or scanner malfunction is indicated and a level F interrupt occurs. Bits 0 through 4 of the FR are not used. The bit configurations are as follows:

- (a) **Short Binary (MA/WA and MAS/WAS):** If bits 9, 8, and 7 of the FR are 000, the contents of the addend KR (23 bits) plus an overall parity bit computed on the 23 bits of the BR are sent directly to the peripheral unit address bus.
- (b) **Long Binary (MA/WA and MAS/WAS):** If bits 9, 8, and 7 of the FR are 001, the contents of the addend KR (23 bits) plus the contents of bit positions 0 through 12 of the KR are sent directly to the peripheral unit address bus. Bits 13 through 22 of the KR are ignored. If the KR appears in the R subfield of the instruction and any register modification is specified, the KR will be modified before bits 0 through 12 are sent out on the peripheral unit address bus. The contents of the addend KR are unaffected by this register modification.
- (c) **Line Switching Frames/Circuits 4:1 (MA/WA and MAS/WAS):** If bits 9, 8, and 7 of the FR are 010, the contents of bits 0 through 9, 12 through 15, and 20 through 22 of the addend KR are sent through translation circuitry and activate 10 of the 36 leads of the peripheral unit address bus. Bits 10, 11, 16, 17, 18, and 19 of the addend KR are ignored. Bit position 5 of the FR must contain a 0 because the line switching frames/circuits do not contain the equipment necessary to make an all-seems-well scanner check.
- (d) **Line Switching Frames/Circuits 2:1 (MA/WA and MAS/WAS):** If bits 9, 8, and 7 of the FR are 011, the contents of bits 0 through 3, 5 through 9, 12 through 15, and 20 through 22 of the addend KR are sent through translation circuitry and activate 9 of the 36 leads of the peripheral unit address bus. Bits 4, 10, 11, 16, 17, 18, and 19 of the addend KR are ignored. Bit position 5 of the FR must contain a 0 because the line switching frames/circuit do not contain the equipment necessary to make an all-seems-well scanner check.
- (e) **Trunk or Junctor Switching Frames/Circuits (MA/WA and MAS/WAS):** If bits 9, 8, and 7 of the FR are 100, the contents of bits 0 through 13 and bits 20 through 22 of the addend KR are sent through translation circuitry and activate 13 of the 36 leads of the peripheral unit address bus.

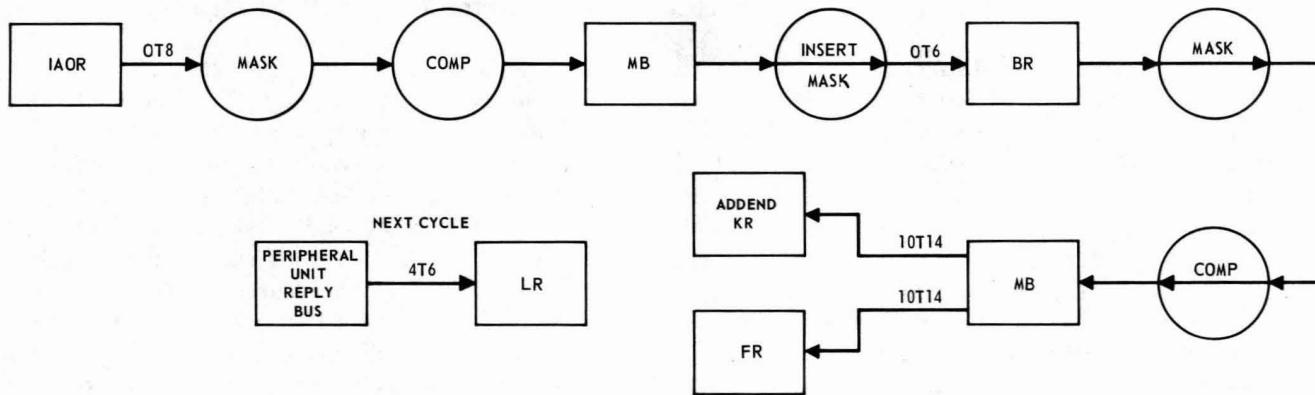
bus. Bits 14 through 19 of the addend KR are ignored. Bit position 5 of the FR must contain a 0 because the trunk and junctor switching frames/circuit do not contain the equipment necessary to make an all-seems-well scanner check.

- (f) **Signal Distributor (MA/WA):** If bits 9, 8, and 7 of the FR are 110, the contents of bits 0 through 10 of the addend KR are sent through translation circuitry and activate 7 of the 36 leads of the peripheral unit address bus. Bits 11 through 22 of the addend KR are ignored.
- (g) **Normal Scanner (MA/WA and MAS/WAS):** If bits 9, 8, and 7 of the FR are 111, the contents of bits 4 through 9 of the addend KR are sent through translation circuitry and activate 2 of the 36 leads of the peripheral unit address bus. Bits 0 through 3 and bits 10 through 22 of the addend KR are ignored.
- (h) **Skew Scanner (MAS/WAS):** If bits 9, 8, and 7 of the FR are 110, the contents of bits 16 through 21 of the addend KR are sent through translation circuitry and activate 2 of the 36 leads of the peripheral unit address bus. Bits 0 through 15 and bit 22 of the addend KR are ignored.
- (i) **Long Binary CPD Address Bus:** If bit positions 9, 8, and 7 of the FR are 101, then bits 10, 11, 12, and 13 of the FR control the execute translators in selecting the correct CPD. The contents of bits 0 through 15 of the addend KR and the contents of bits 0 through 15 of the KR are sent directly onto the CPD address bus. The CPD address bus leads, thus activated by this binary code, enable the CPD output. If bit position 16 of the KR contains a 1, a test lead pulse is also sent out on the CPD address bus. If bit position 17 of the KR contains a 1, reset lead pulse is also sent out on the CPD address bus. Bits 16 through 22 of the addend KR and bits 18 through 22 of the KR are ignored. If bit position 6 of the FR contains a 1, the contents of the YR are cleared (made all zeros) and YR accepts an enable verify word. This word is a response via the CPD from the peripheral unit actually addressed. If it is not the unit intended, a wired-check will cause a level F interrupt. Bit position 5 of the FR must contain a 0 since an all-seems-well scanner check, which depends on information from the peripheral unit address bus, cannot be made. The remaining bits (0 through 4 and 14 through 22) of the FR are ignored. If the KR appears in the R subfield of the instruction and any register modification is specified, the KR will be modified before any of its contents are sent out on the CPD bus. The contents of the addend KR are unaffected by this register modification.

**8.95 WSF(00660)—Word to Scanner and Set FR:** The resultant DAR number W, after possible product masking and/or complementing, replaces the contents of the BR, FR, and addend KR. The contents of bits 10, 11, 12, and 13 of the FR control the execute translators in selecting the CPD. A particular point within the CPD is chosen by bits 14, 15, and 16 of the FR that make a vertical range selection and bits 17, 18, 19, 20, 21, and 22 that make a horizontal range selection, thereby enabling the scanner. A controller within a peripheral unit and a peripheral unit address bus are determined by using bits 10 and 14 of the FR as follows:

FR BIT 10	FR BIT 14	CONTROLLER	PERIPHERAL UNIT ADDRESS BUS
0	0	0	0
0	1	1	1
1	0	1	0
1	1	0	1

- (a) The contents of bits 4 through 9 of the addend KR are sent through translation circuitry and activate 2 of the 36 leads on the peripheral unit address bus to select the scanner row address.
- (b) The contents of the LR are cleared. Then, during a period of time extending beyond the cycle time of this instruction, bit positions 0 through 15 receive the answer from the scanner.
- (c) The contents of the YR are cleared and the YR accepts an enable verify word. The verify word is a response via the CPD from the peripheral unit actually addressed. If the responding unit is not the unit addressed, an all-seems-well scanner check (wired) will cause a level F interrupt. If an all-seems-well signal is not received as the result of a wired check, an improper address or scanner malfunction is indicated and a level F interrupt occurs.
- (d) The WSF instruction does not affect the C control flip-flops.

**Options:****RM**

I (0V000 ; V = Index Reg No.)

**LCJ**PL(00004)  
PS(00005)  
C(10000)**Restrictions:** On instructions following WSF:

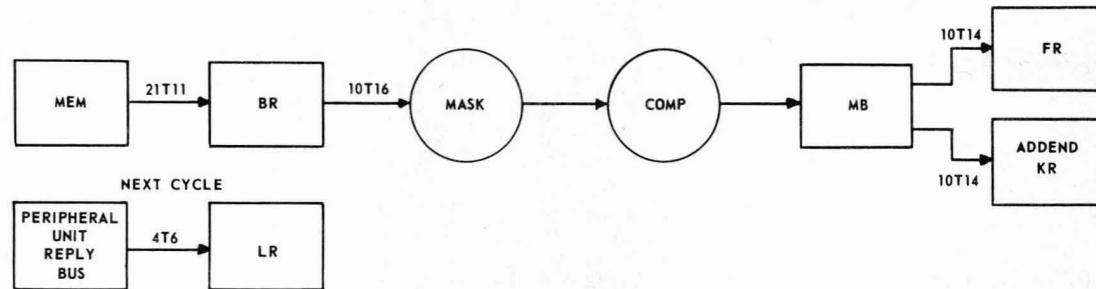
- (1) The following instruction must not be an input-output instruction.
- (2) K (accumulator) restrictions (8.99 and 8.100).
- (3) F register restrictions (8.101 and 8.102).
- (4) Y register restrictions (8.103 and 8.104).
- (5) L register restrictions (8.105 and 8.106).

**8.96 MSF(00260)—Memory to Scanner and Set FR:** The contents of location M replace the contents of the BR and, after possible product masking and/or complementing, replace the contents of the FR and the addend KR. The contents of bits 10, 11, 12, and 13 of the FR control the execute translators in selecting the CPD. A particular point within the CPD is chosen by bits 14, 15, and 16 of the FR that make a vertical range selection and bits 17, 18, 19, 20, 21, and 22 that make a horizontal range

selection, thereby enabling the scanner. A controller and a peripheral unit address bus are determined by using bits 10 and 14 of the FR as follows:

FR BIT 10	FR BIT 14	CONTROLLER	PERIPHERAL UNIT ADDRESS BUS
0	0	0	0
0	1	1	1
1	0	1	0
1	1	0	1

- (a) The contents of bits 4 through 9 of the addend KR are sent through translation circuitry and activate 2 of the 36 leads on the peripheral unit address bus to select the scanner row address.
- (b) The contents of the LR are cleared. Then, during a period of time extending beyond the cycle time of this instruction, bit positions 0 through 15 receive the answer from the scanner.
- (c) The contents of the YR are cleared and the YR accepts an enable verify word. This verify word is a response via the CPD from the peripheral unit actually addressed. If the responding unit is not the unit addressed, an all-seems-well scanner check (wired) will cause a level F interrupt. If an all-seems-well signal is not received as the result of a wired-check, an improper address or scanner malfunction is indicated and a level F interrupt occurs.
- (d) The MSF instruction does not affect the C control flip-flops.

**Options:****RM****LCJ**

I(0V000 ; V = Index Reg No.)  
 A(20000)  
 W(00000 1)  
 S(20000 1)

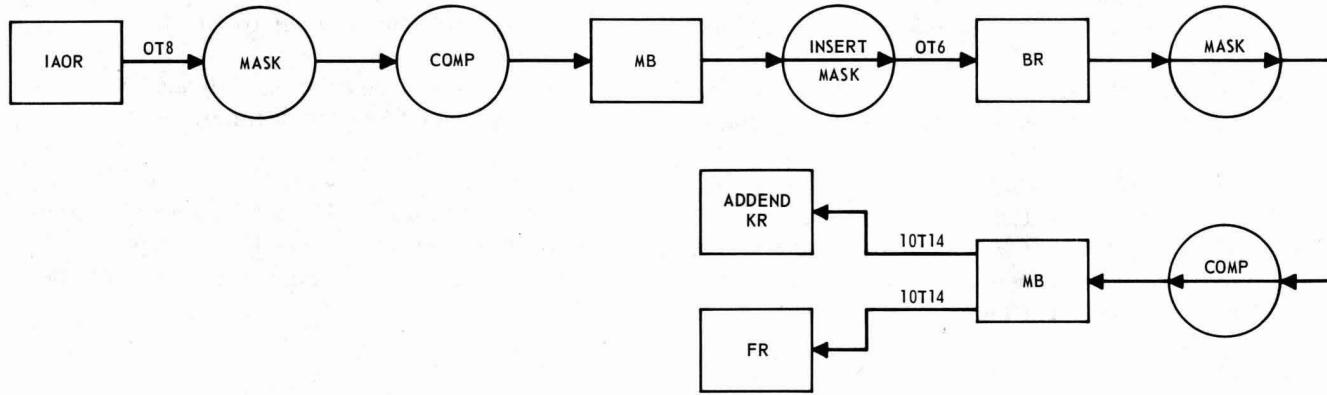
PL(00000 2)  
 PS(00001)  
 C(10000)

**Restrictions:** On instructions following MSF:

- (1) The following instruction must not be an input-output instruction.
- (2) K (accumulator) restrictions (8.99 and 8.100).

- (3) F register restrictions (8.101 and 8.102).
- (4) Y register restrictions (8.103 and 8.104).
- (5) L register restrictions (8.105 and 8.106).

**8.97 WD(00662)—Word to CPD:** The resultant DAR number W, after possible product masking and/or complementing, replaces the contents of the BR, the contents of the FR, and the contents of the addend KR. Bits 10, 11, 12, and 13 of the FR control the execute translators in selecting the CPD. A particular point within a CPD is chosen by bits 9, 14, 15, and 16 of the FR that make a vertical range selection and bits 17, 18, 19, 20, 21, and 22 that make a horizontal range selection, thereby setting or resetting a particular flip-flop, relay, etc. If bit position 8 of the FR contains a 1, a confirming pulse will be sent. Bits 0 through 7 of the FR are ignored. The WD instruction does not affect the C control flip-flops.

**Options:****RM****LCJ**

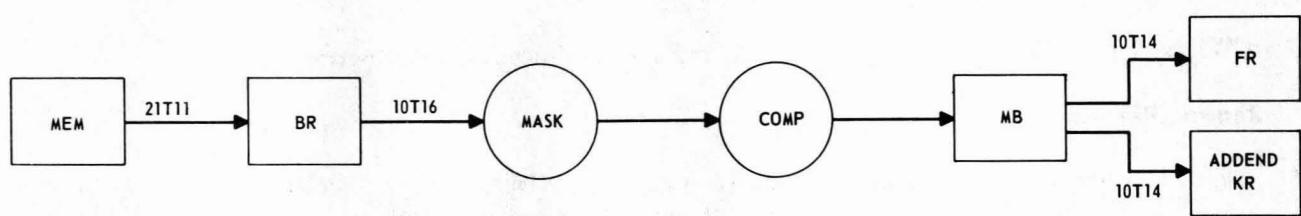
I(0V000 ;V = Index Reg No.)

PL(00004)  
 PS(00005)  
 S(00001)  
 C(10000)

**Restrictions:**

- (1) The following instruction must not change register F.
- (2) If the following instruction consumes only one cycle of time in its execution, the next instruction, unless it is another input-output instruction, must not change register F.

**8.98 MD(00262)—Memory to CPD:** The contents of location M replace the contents of the BR and, after possible product masking and/or complementing, replace the contents of the FR and the addend KR. Bits 10, 11, 12, and 13 of the FR control the execute translators in selecting the correct CPD. A particular point within a CPD is chosen by bits 9, 14, 15, and 16 of FR that make a vertical range selection and bits 17, 18, 19, 20, 21, and 22 of the FR that make a horizontal range selection, thereby setting or resetting a particular flip-flop, relay, etc. If bit position 8 of the FR contains a 1, a confirming pulse will be sent. Bits 0 through 7 of the FR are ignored. The MD instruction does not affect the C control flip-flops.

**Options:****RM**

I(0V000;V = Index Reg No.)  
 A(20000)  
 W(00000 1)  
 S(20000 1)

**LCJ**

PL(00000 2)  
 PS(00001)  
 S(20401)  
 C(10000)

**Restrictions:**

- (1) The following instruction must not change register F.
- (2) If the following instruction consumes only one cycle of time in its execution, the instruction after that, unless it is another input-output instruction, must not change register F.

**RESTRICTIONS ON INPUT-OUTPUT INSTRUCTIONS MA, WA, MAS, WAS, MSF, AND WSF****A. K (Accumulator) Restrictions**

**8.99** The following instruction may use K (accumulator) in only these specific ways:

- (1) As an index register (but register modifications are not permitted)
- (2) In the instruction KM
- (3) TK...
- (4) Shift (H, HC), rotate (Q, QC, QS, QSC)
- (5) AKR, or SKR (but R may not be identified by K)
- (6) CWR or TR... with K in the R subfield (but register modification is not permitted).

**8.100** If the following instruction consumes only one cycle of time in its execution, in the next instruction, K (accumulator) can only be used in one of the following ways:

- (1) An input-output instruction
- (2) As an index register (but register modifications are not permitted)
- (3) In the instruction KM
- (4) TK...
- (5) AKR or SKR (but R may not be identified by K)

## SECTION 231-001-102

- (6) AKR or SKR (but R may not be identified by K)
- (7) CWR or TR... with K in the R subfield (but register modification is not permitted).

### B. F Register Restrictions

**8.101** The following instruction must not change register F. (This, of course, rules out such instructions as TZRFU, TZRFZ, some input-output instructions, some of the combined instructions, instructions which change F by register modifications, etc.)

**8.102** If the following instructions consume only one cycle of time in its execution, the next instruction, unless it is MF, TCMMF, HMF, QMF or another input-output instruction, must not change register F.

### C. Y Register Restrictions

**8.103** The following instruction must not use register Y.

**8.104** If the following instruction consumes only one cycle of time in its execution, in the next instruction, Y can only be used in an input-output instruction.

### D. L Register Restrictions

**8.105** The following instructions must not use register L. (This, of course, rules out such instructions as PWX/Y/Z, PMX/Y/Z, UWX/Y/Z, UMX/Y/Z instructions specifying masking options, etc.)

**8.106** If the following instruction consumes only one cycle of time in its execution, in the next instruction, register L can only be used in one of the following ways:

- (1) By an input-output instruction
- (2) The instruction ML
- (3) PL masking on an instruction having a memory location as its origin
- (4) In one of the following combined instructions: UNKmj, KMKUS, KMKXS.

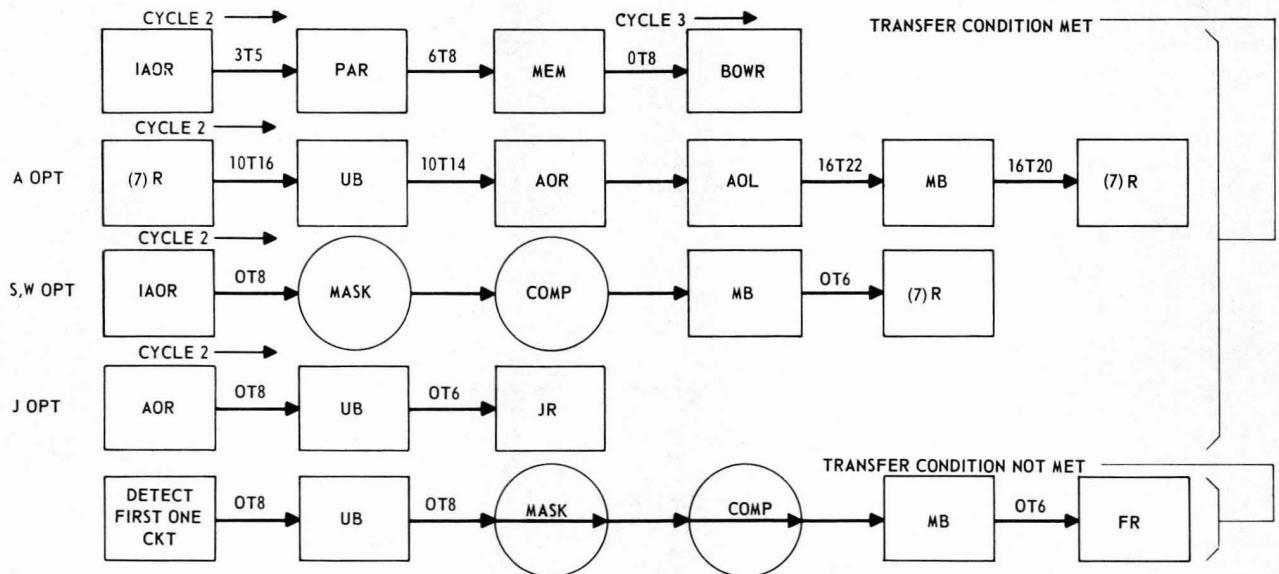
## COMBINED INSTRUCTIONS

**8.107** A combined instruction calls for two or more operations that generally require separate instructions. There are 11 combined instructions with each instruction designed to save time in frequently executed programs such as scans for input data. Some of the operations are conditional and can involve transfers. When no transfers are involved, a combined instruction is executed in a single machine cycle. The following definitions of the instructions do not specify or assume any particular use, but are grouped according to primary functions.

### A. General Purpose Combined Instructions

**8.108 TZRFU(00014)—Transfer if Accumulator is 0 or Find Right-Most 1:** If the contents of the KR are logical 0 (all zeros), program control is transferred to another instruction. In the case of a direct transfer specified by an empty M subfield, the address of this instruction is the resultant DAR address. In the case of an indirect transfer specified by an M in the subfield, the address of the instruction is contained in a location specified by the resultant DAR address. The contents of bits 0 through 19 of this location contain the address of the instruction to which control is transferred. If the

content of the KR is not logical 0, the bit address of the right-most 1 of the KR is placed in the FR. This number will be between 0 and 22, inclusive. Options occur only if transfer is executed.



**Options:**

**RM**

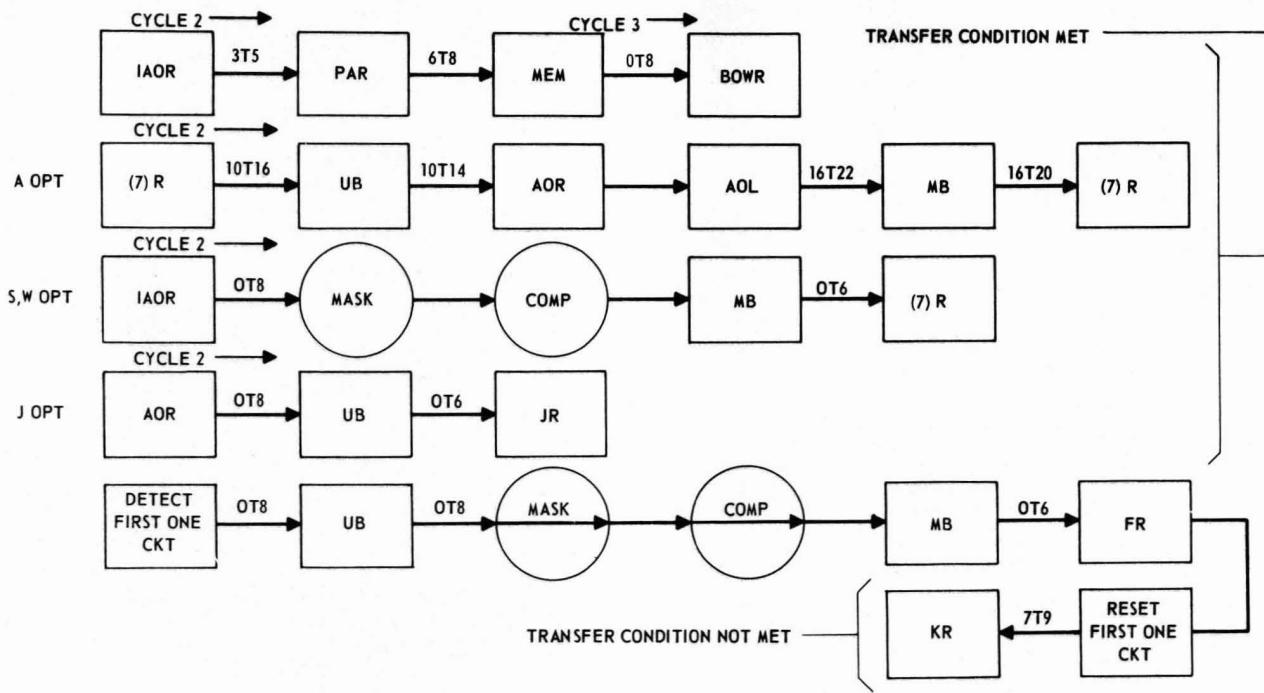
**LCJ**

I(0V000;V = Index Reg No.)  
A(20000)  
W(000001)  
S(200001)  
M(00002)

J(10000)

**Note:** If both the J option and the A option are specified, the return address placed in J is incremented by one if J is equal to I.

**8.109 TZRFZ(00015)—Transfer if Accumulator is 0 or Find and Zero Right-Most 1:** If the contents of the KR are logical 0 (all zeros), program control is transferred to another instruction. In the case of a direct transfer specified by an empty M subfield, the address of this instruction is the resultant DAR address. In the case of an indirect transfer specified by an M in the M subfield, the address of the instruction is contained in a location specified by the resultant DAR address. The contents of bits 0 through 19 of this location contain the address of the instruction to which control is transferred. If all KR bits are not 0, the right-most 1 is made 0 and its bit address is placed in the FR. The number placed in the FR will be between 0 and 22, inclusive. Options occur only if transfer is executed.



Options:

RM

LCJ

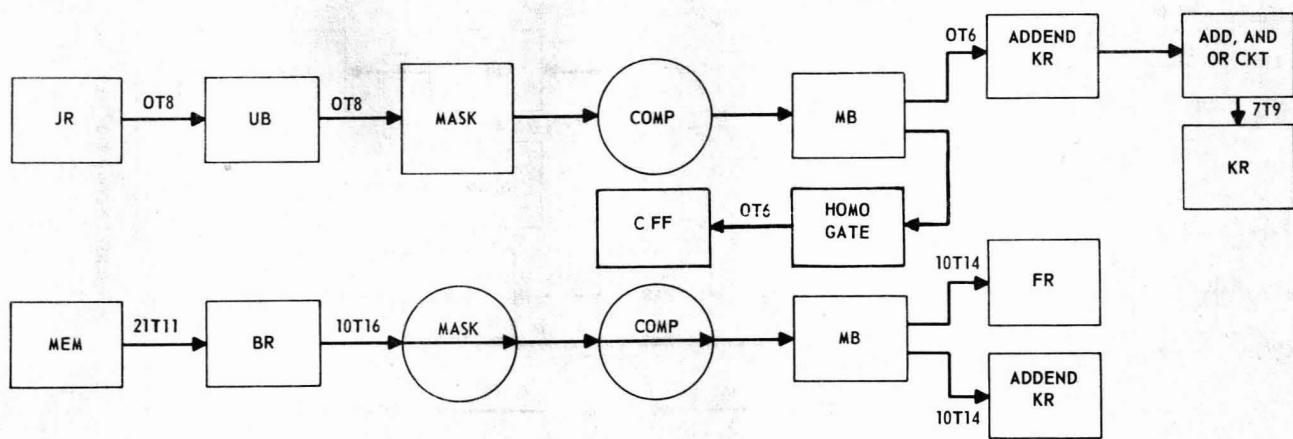
I(0V000 ; Index Reg No.)  
 A(20000)  
 W(00000 1)  
 S(20000 1)  
 M(00002)

J(10000)

**Note:** If both the J option and the A option are specified, the return address placed in J is incremented by 1 if J is equal to I.

#### B. Dial Pulse Scan Combined Instructions

**8.110 JKMSF(00264):** The contents of the JR, after possible product masking and/or complementing, replace the contents of the KR and set the C control flip-flops. This instruction is then completed by the execution of the instruction MSF (memory to scanner and set the FR as covered in 8.96, which is unaffected by the options specified in the LCJ subfield).

**Options:****RM****LCJ**

PL(00000 2)  
 PS(00001)  
 C(10000)

**For MSF**

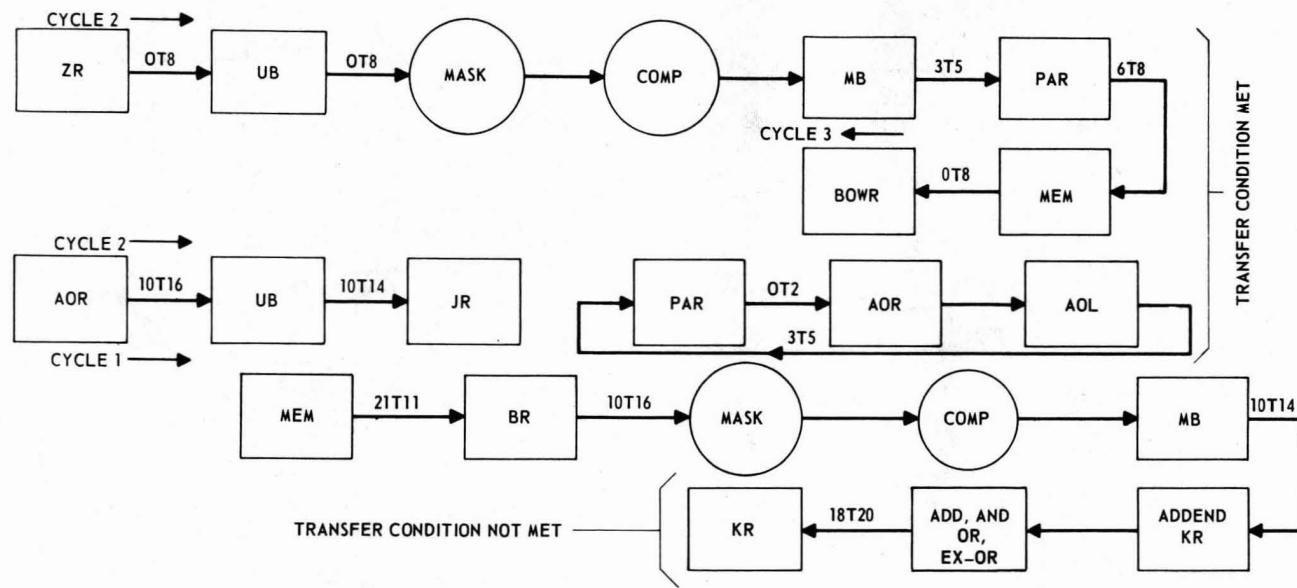
I(0V000; V = Index Reg No.)  
 A(20000)  
 W(00000 1)  
 S(20000 1)

**Restrictions:**

- (1) The following instruction must not be an input-output instruction.
- (2) KR restrictions (8.99 and 8.100).
- (3) FR restrictions (8.101 and 8.102).
- (4) YR restrictions (8.103 and 8.104).
- (5) LR restrictions (8.105 and 8.106).
- (6) Location M cannot be the PS.

**8.111 TAUMK(00244):** If the C control flip-flops do not indicate arithmetic 0, a transfer is made to the address in bits 0 through 19 of the ZR, and the return address is placed in the JR. If the C control flip-flops indicate arithmetic 0, the transfer is not executed and the contents of location M replace the contents of the BR and, after possible product masking and/or complementing, replace the contents of the KR. Options occur only if transfer is not executed.

**SECTION 231-001-102**



## Options:

RM

LCJ

I(0V000;V = Index Reg No.)  
A(20000)  
W(00000 1)  
S(20000 1)

PL(00000 2)  
PS(00001)  
S(20401)  
C(10000)

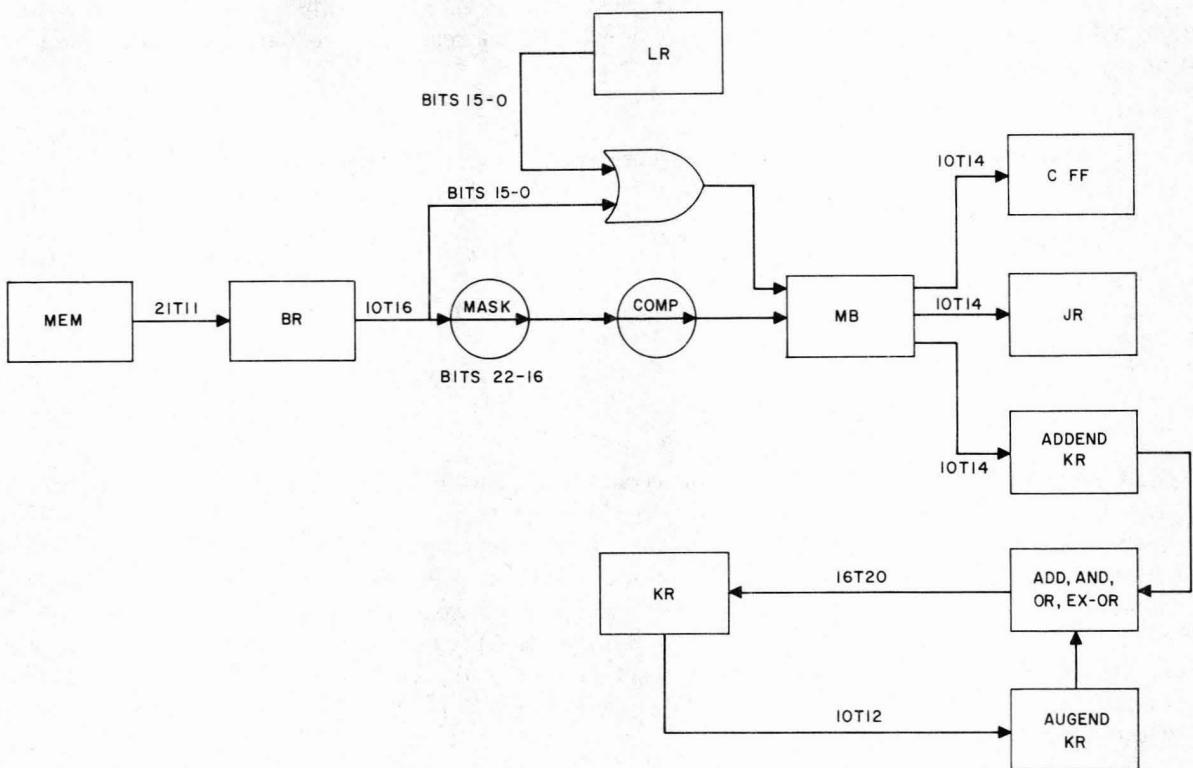
### Restrictions:

(1) The following instruction must not be

- (a) AKR or SKR
- (b) One of the following instructions specifying K to identify R: CWR, AWRP, TR instructions.

(2) TAUMK cannot be executed from the CS.

**8.112 *UMK MJ(00274)*:** The contents of location M replace the contents of the BR. Bits 0 through 15 of the new contents of BR and bits 0 through 15 of the LR are combined by the EXCLUSIVE OR function. Bits 0 through 15 of the resultant word and bits 16 through 22 of the new contents of the BR form a new word that replaces the contents of the JR and sets the C control flip-flops. The same word that replaces the contents of the JR is combined by the logical union (OR) function with the contents of the KR. This result replaces the contents of the KR.



**Options:** RM

I(0V000;V = Index Reg No.)  
 A(20000)  
 W(00000 1)  
 S(20000 1)

**Restrictions:** The following instruction must not

- (1) Specify K in the R subfield
- (2) Be AKR or SKR.

#### C. Line Scan Combined Instructions

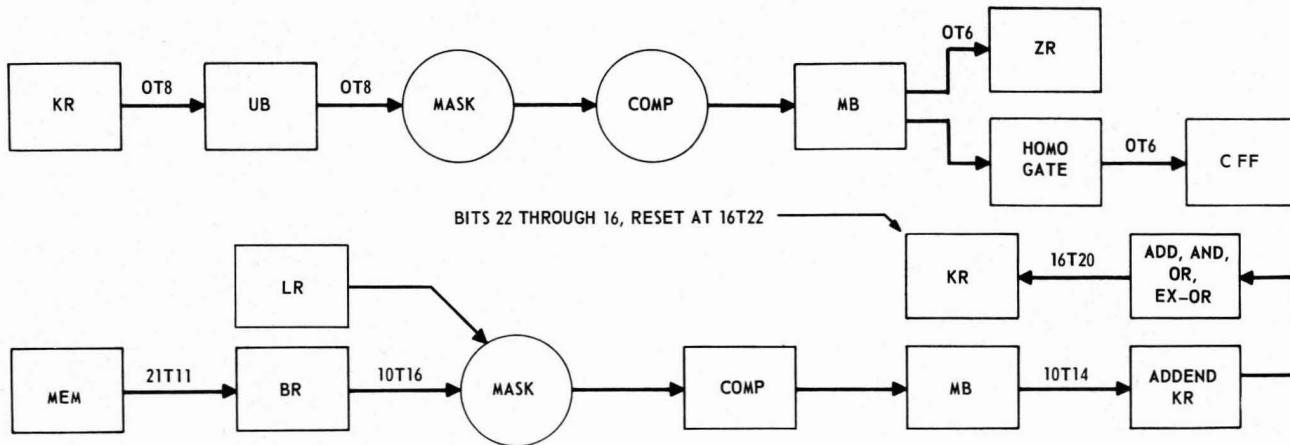
**8.113 KMKUS(10370):** The contents of the KR replace the contents of the ZR and set the C control flip-flops, which are then unaffected by the remaining actions of the instruction. Then, the contents of location M replace the contents of the BR. Bits 0 through 15 of the new contents of the BR and bits 0 through 15 of the contents of the LR are combined by the logical union (OR) function, and the result replaces the contents of bits 0 through 15 of the KR. (However, if bit 22 of the logic register L is a 1, which is the case if a scanner all-seens-well failure occurred on a previous WAS, MAS, KMKUS, or KMKXS, bits 0 through 15 of the accumulator K are made zeros.) Bits 16 through 22 of the KR are made zeros. Bits 16 through 21 of the complemented result replace the contents of bits 16 through 21 of the addend KR. Bits 0 through 15 and bit 22 of the addend KR are ignored. Bits 9, 8, and 7 of the FR must be 110 in order that the contents of bits 16 through 21 of the addend KR be sent through the skew scanner translator to activate 2 of the 36 leads of the peripheral unit address bus. Bits 10, 11, 12, and 13 of the FR control the execute translators in selecting the CPD. A particular point within

## SECTION 231-001-102

a CPD is chosen by bits 14, 15, and 16 of the FR that make a vertical range selection and bits 17, 18, 19, 20, 21, and 22 that make a horizontal range selection, thereby enabling the peripheral unit. A controller within the peripheral unit and a peripheral unit address bus are determined by using bits 10 and 14 of the FR as follows:

FR BIT 10	FR BIT 14	CONTROLLER	PERIPHERAL UNIT ADDRESS BUS
0	0	0	0
0	1	1	1
1	0	1	0
1	1	0	1

If bit position 6 of the FR contains a 1, the contents of the YR are cleared and the YR accepts an enable verify word. This word is a response via the CPD from the peripheral unit actually addressed. If the responding unit is not the unit addressed, an all-seems-well scanner check (wired) will cause a level F interrupt. If bit position 5 of the FR contains a 1, a wired-check is made. If an all-seems-well signal is not received as the result of a wired check, an improper address or scanner malfunction is indicated and a level F interrupt occurs. The contents of the LR are cleared after which (during a period of time extending beyond the cycle time of this instruction) bit positions 0 through 15 accept the information on the scanner answer bus from the peripheral unit. If a scanner all-seems-well failure occurs, bit 22 of the logic register is set.



Options:

RM

I(OV000;V = Index Reg No.)  
 A(20000)  
 W(00000 1)  
 S(20000 1)

**Restrictions:**

- (1) The following instruction must not be an input-output instruction.
- (2) KR (1 through 4 of 8.99 and 1, 4, 5, 6, 7 of 8.100).
- (3) FR (8.101 and 8.102).
- (4) YR (8.103 and 8.104).
- (5) LR (8.105 and 8.106).
- (6) Location M cannot be the PS.

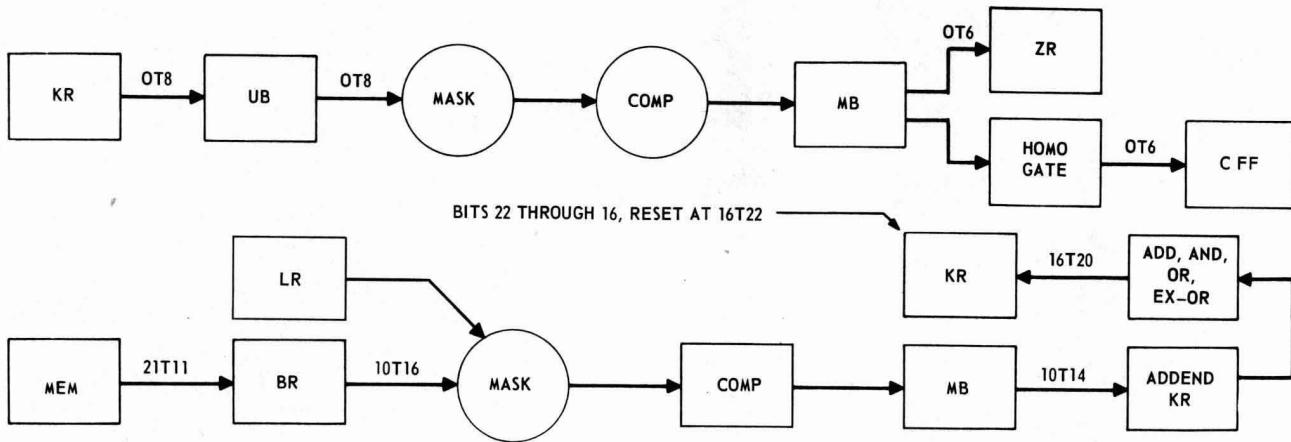
**D. Trunk Scan Combined Instructions**

**8.114 KMKXS(00372):** The contents of the KR first replace the contents of the ZR and set the C control flip-flops, which are then unaffected by the remaining actions of the instruction. Then, the contents of location M replace the contents of the BR. Bits 0 through 15 of the new contents of the BR and bits 0 through 15 of the contents of the LR are combined by the EXCLUSIVE OR function, and the result replaces the contents of bits 0 through 15 of the KR. (However, if bit 22 of the logic register L is a 1, which is the case if a scanner all-seems-well failure occurred on a previous WAS, MAS, KMKUS, or KMKXS, bits 0 through 15 of the accumulator K are made zeros.) Bits 16 through 22 of the KR are made zeros. Bits 16 through 21 of the new contents of the BR replace the contents of bits 16 through 21 of the addend KR. Bits 0 through 15 and bit 22 of the addend KR are ignored. Bits 9, 8, and 7 of the FR must be 110 in order that the contents of bits 16 through 21 of the addend KR be sent through the skew scanner translator to activate 2 of the 36 leads of the peripheral unit address bus. Bits 10, 11, 12, and 13 of the FR control the execute translators in selecting the correct CPD. A particular point within a CPD is chosen by bits 14, 15, and 16 of the FR that make a vertical range selection and bits 17, 18, 19, 20, 21, and 22 that make a horizontal range selection, thereby enabling the peripheral unit. A controller within the peripheral unit and a peripheral address bus are determined by using bits 10 and 14 of the FR as follows:

FR BIT 10	FR BIT 14	CONTROLLER	PERIPHERAL UNIT ADDRESS BUS
0	0	0	0
0	1	1	1
1	0	1	0
1	1	0	1

If bit position 6 of the FR contains a 1, the contents of the YR are cleared and the YR accepts an enable verify word. This word is a response via the CPD from the peripheral unit actually addressed. If the responding unit is not the unit addressed, an all-seems-well scanner check (wired) will cause a level F interrupt. If bit position 5 of the FR contains a 1, a wired-check is made. If an all-seems-well signal is not received as the result of a wired-check, an improper address or scanner malfunction is indicated and a level F interrupt occurs. The contents of the LR are cleared after which (during a period of time extending beyond the cycle time of this instruction) bit positions 0 through 15 accept the information on the scanner answer bus from the peripheral unit. If a scanner all-seems-well failure occurs, bit 22 of the LR is set.

**SECTION 231-001-102**



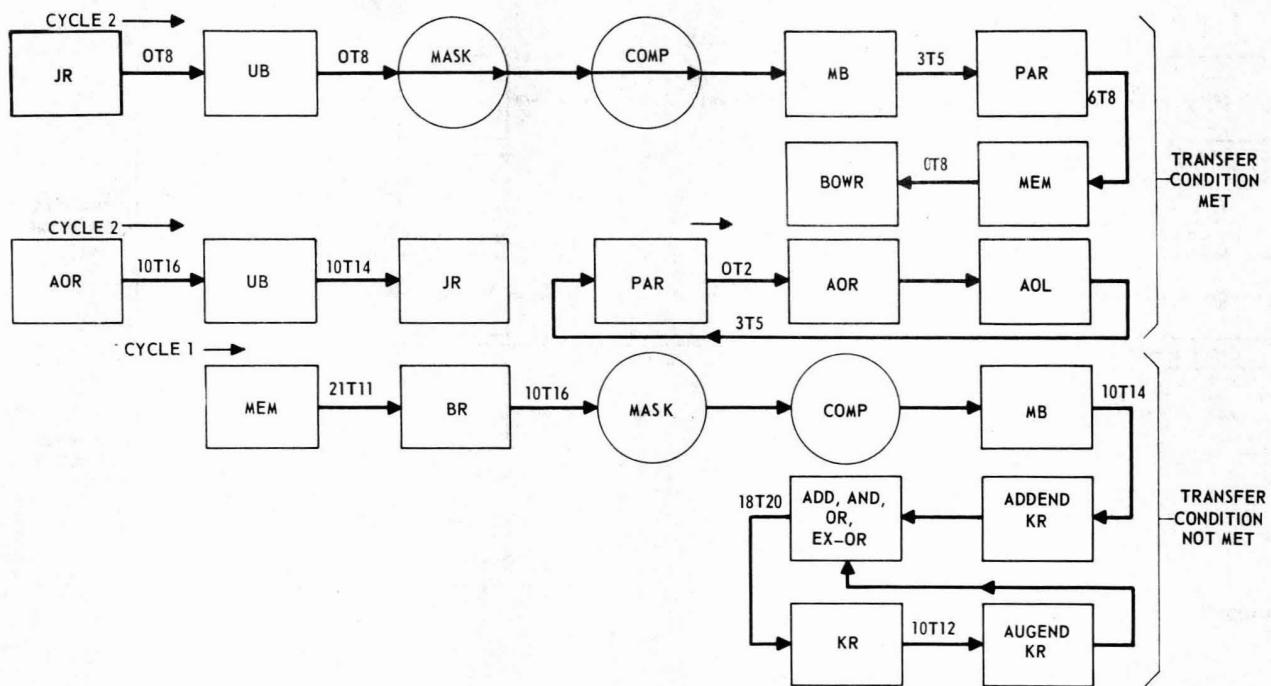
**Options:** RM

I(0V000;V=Index Reg No.)  
 A(20000)  
 W(00000 1)  
 S(20000 1)

**Restrictions:**

- (1) The following instruction must not be an input-output instruction.
- (2) KR (1 through 4 of 8.99 and 1, 4, 5, 6, and 7 of 8.100).
- (3) FR (8.101 and 8.102).
- (4) YR (8.103 and 8.104).
- (5) LR (8.105 and 8.106).
- (6) Location M cannot be the PS.

**8.115 TUPMK(00246):** If the C control flip-flops do not indicate arithmetic 0, a transfer is made to the address specified by bits 0 through 19 of the JR and the return address is placed in the JR. If the C control flip-flops indicate arithmetic 0, the contents of location M replace the contents of the BR and, after possible product masking and/or complementing, are combined by the logical product (AND) function with the contents of the KR. This result replaces the contents of the KR. Options occur only if a transfer is not executed.

**Options:****RM**

I(0V000;V=Index Reg No.)  
 A(20000)  
 W(00000 1)  
 S(20000 1)

**LCJ**

PL(00000 2)  
 PS(00001)  
 S(20401)  
 C(10000)

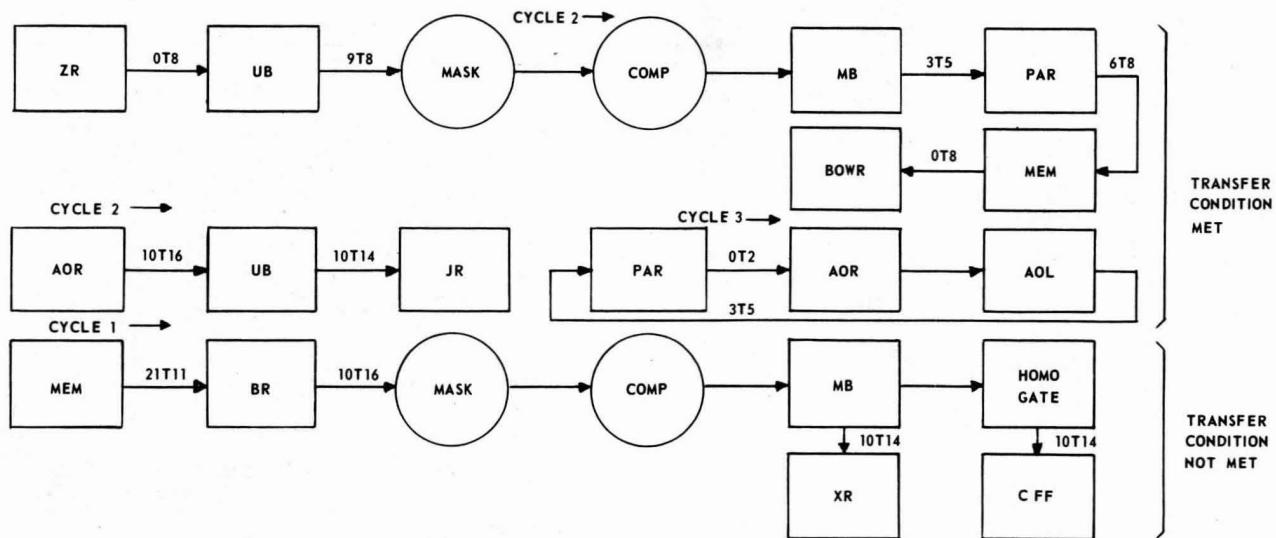
**Restrictions:**

- (1) The following instruction must not
  - (a) Specify K in the R subfield
  - (b) Be AKR or SKR.
- (2) TUPMK cannot be executed from the CS.

**E. Network Combined Instructions**

**8.116 TCGMX(00316):** If the C control flip-flops indicate a state greater than zero, a transfer is made to the address represented in the ZR and the return address is stored in the JR. If the C control flip-flops indicate a state equal to or less than zero, the contents of location M replace the contents of the BR and, after possible product masking and/or complementing, set the C control flip-flops and replace the contents of the XR. Options occur only if transfer is not executed.

**Note:** The contents of the Z register are not changed.

**Options:****RM**

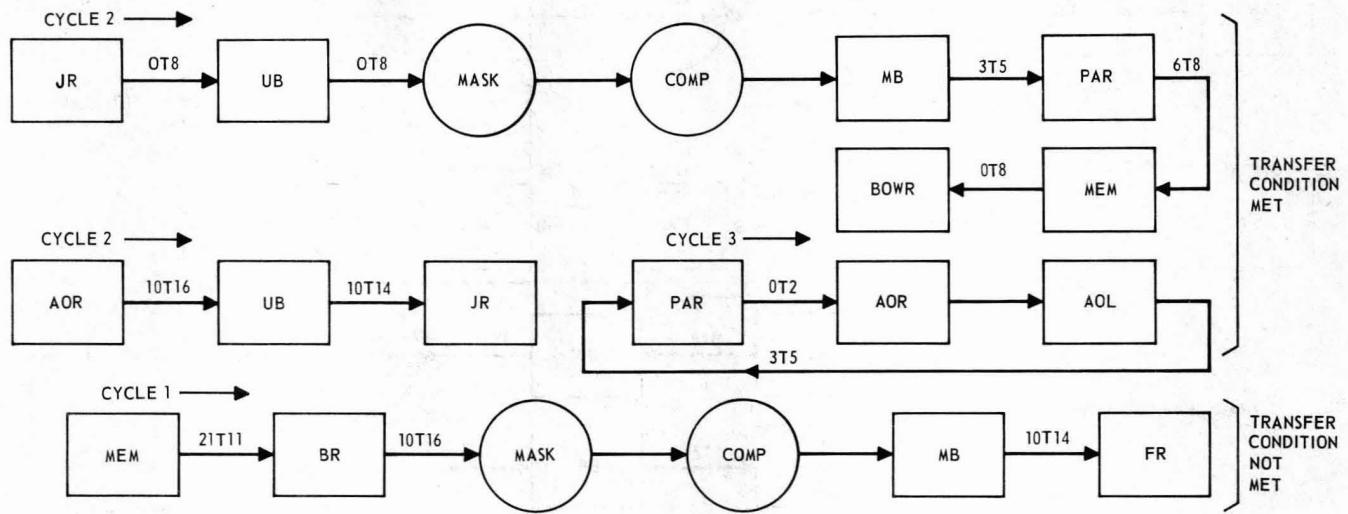
I(0V000;V=Index Reg No.)  
 A(20000)  
 W(00000 1)  
 S(20000 1)

**LCJ**

PL(00000 2)  
 PS(00001)  
 S(20401)  
 C(10000)

**Restrictions:** TCGMX cannot be executed from the call store.

**8.117 TCMMF(00220):** If the sign bit of the C control flip-flops is minus, a transfer is made to the address specified in bits 0 through 19 of the JR and the return address is placed in the JR. If the sign bit of the C control flip-flops is plus, the contents of location M replace the contents of the BR and, after possible product masking and/or complementing, set the C control flip-flops and replace the contents of the FR. Options occur only if transfer is not executed.

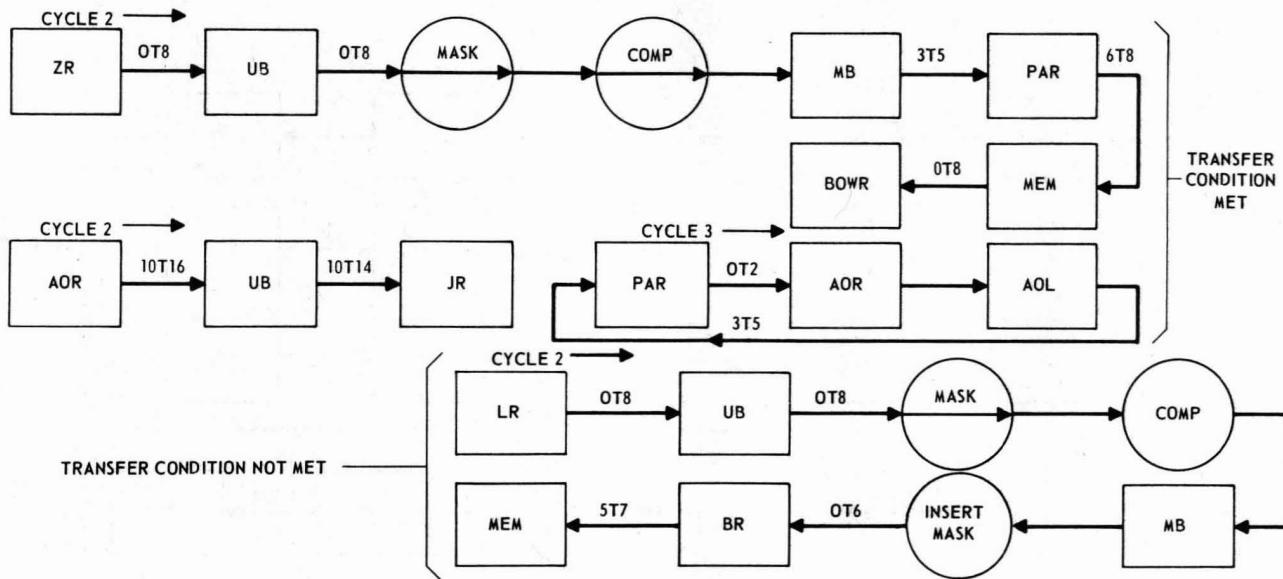
**Options:****RM****LCJ**

I(0V000;V = Index Reg No.)  
 A(20000)  
 W(00000 1)  
 S(20000 1)

PL(00000 2)  
 PS(00001)  
 S(20401)  
 C(10000)

**Restriction:** This instruction cannot be executed from the CS.

**8.118 TAULM(00040):** If the C control flip-flops do not indicate arithmetic 0, a transfer is made to the address specified in bits 0 through 19 of the ZR and the return address is placed in JR. If the C control flip-flops indicate arithmetic 0, the contents of the LR, possibly complemented, replace or are insertion masked into the contents of the BR, and the new contents of the BR replace the contents of location M.



Options:

RM

LCJ

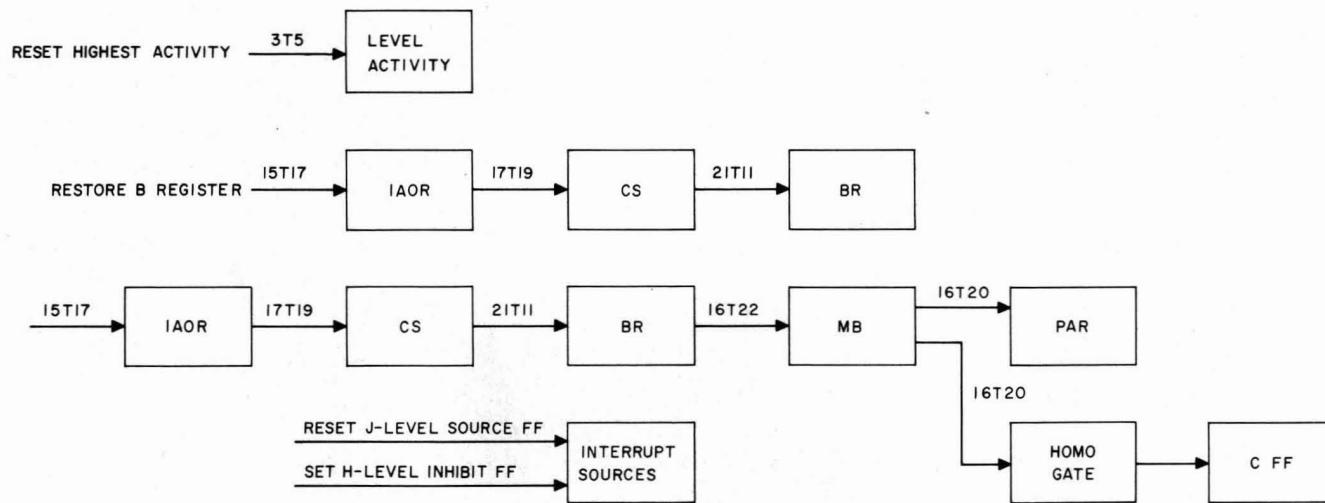
I(0V000;V = Index Reg No.)  
 A(20000)  
 W(00000 1)  
 S(20000 1)

EL(00100 2)  
 ES(00101)  
 C(10000)

Restriction: This instruction cannot be executed from the CS.

#### F. Interrupt Combined Instructions

**8.119 GBNHJ(05560)—GO Back to Normal, Inhibit H, and Reset J:** This instruction (given at the completion of the J level interrupt program in place of EGBN), in addition to performing the functions of the instruction EGBN, will inhibit the H level interrupt (5-millisecond clock pulse) and reset the J level source flip-flop in interrupt source register B8NOIS.



## MAINTENANCE AND SPECIAL PURPOSE INSTRUCTIONS

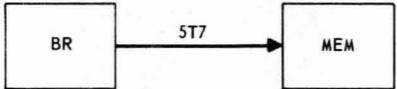
**8.120 BMAP(00106)—BR to Memory and Invert Address Parity:** The parity bit computed (normally odd) on the address of location M is inverted (made even) to cause a CS parity check failure, which prevents the contents of the BR from replacing the contents of location M. The CC does not receive the all-seems-well CS check signal and a level D (CS reread failure) interrupt occurs. This instruction is used by maintenance programs as a test for the CS and CC parity circuits.



**Options:** RM

I(0V000;V = Index Reg No.)  
 A(20000)  
 W(00000 1)  
 S(20000 1)

**8.121 BMOP(10102)—BR to Memory and Invert Overall Parity:** The contents of the BR replace the contents of location M. The overall parity bit (normally odd) computed on location M and the contents of the BR are inverted and stored as the 24th bit of location M. If ES appears in the L subfield, the contents of the DA field replace the contents of the LR, but no insertion masking occurs. This instruction is used by certain maintenance programs as a test for the CS and CC parity circuits.



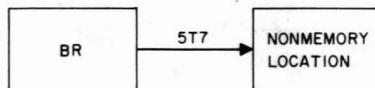
**Options:** RM

LCJ

I(0V000;V = Index Reg No.)  
 A(20000)  
 W(00000 1)  
 S(20000 1)

ES(00001)

**8.122 BN(00142)—BR to Nonmemory:** The contents of the BR are sent to the nonmemory location specified by the address in the instruction. Nonmemory locations are control points in a CS, signal processor internal locations, and registers and control points internal to either CC. If ES appears in the L subfield, the contents of the DA field replace the contents of the LR, but no insertion masking occurs.



Options:

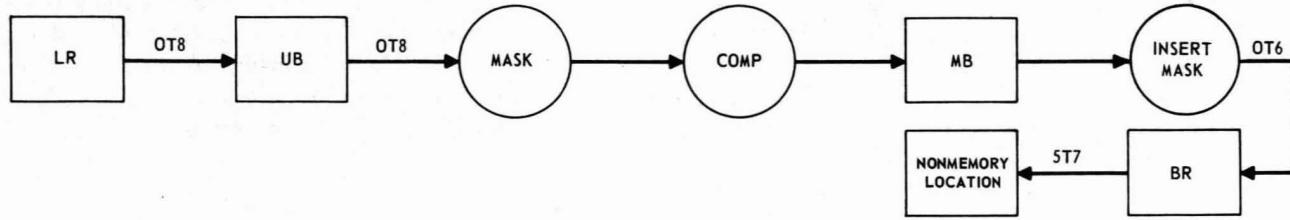
RM

LCJ

I(0V000;V= Index Reg No.)  
W(00000 1)

ES(00001)

**8.123 LN(00140)—LR to Nonmemory:** If neither ES nor PS masking is specified in the instruction, the contents of the LR, after possible complementing, replace or are insertion masked (EL) into the contents of the BR. The new BR contents replace the contents of the specified nonmemory location. If either ES or PS masking is called for in the instruction, the contents of the DA field replace the contents of the LR. The new contents of the LR, after possible product masking and/or complementing, replace or are insertion masked into the contents of the BR. The new contents of the BR replace the contents of the specified nonmemory location.



Options:

RM

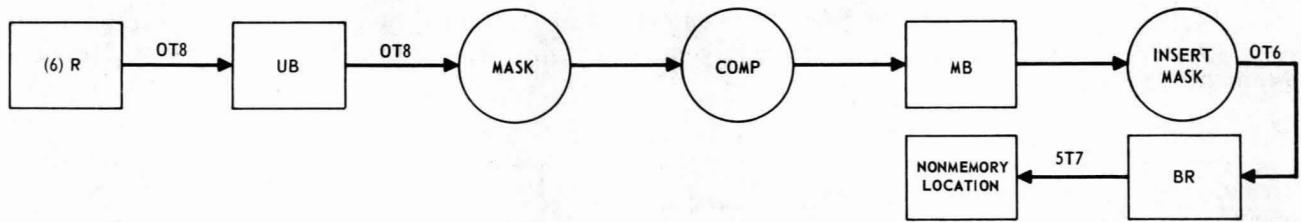
LCJ

I(0V000;V= Index Reg No.)  
W(00000 1)

EL(00000 2)  
ES(00001)  
PS(00005)  
C(10000)

**8.124 FN(00150), JN(00160), KN(00152), XN(00162), YN(00170), ZN(00172)—Register to Nonmemory:**

The contents of the FR, JR, KR, XR, YR, or ZR, whichever is specified in the operation code after possible product masking and/or complementing, replace or are insertion masked into the contents of the BR. The new BR contents replace the contents of the specified nonmemory location. The nonmemory location represents some inspection point, flip-flop, etc.

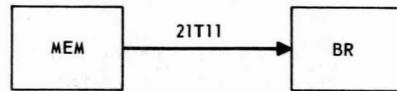
**Options:****RM**

I(0V000;V = Index Reg No.)  
W(00000 1)

**LCJ**

EL(00000 2)  
ES(00001)  
PL(00004 2)  
PS(00005)  
C(10000)

**8.125 MBOP(10202)—Memory to BR with Even Overall Parity:** If the resultant DAR address specifies a CS location M, the overall parity bit stored in the 24th bit of this location (which, when stored, was computed on the CS memory address and the contents of the location) is expected to be even (instead of odd) when read by the CS reading check circuitry. If the parity is even, the contents of CS location M replace the contents of the BR. If the parity is odd, a level D (CS re-read failure), interrupt occurs. If the resultant DAR address specifies a PS location M, parity is checked and the instruction executed as if the instruction were MB. If PS is specified in the L subfield, the contents of the DA field replace the contents of LR, but no actual masking takes place.

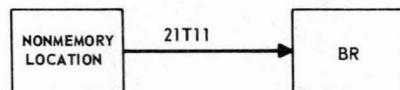
**Options:****RM**

I(0V000;V = Index Reg No.)  
A(20000)  
W(00000 1)  
S(20000 1)

**LCJ**

PS(00001)

**8.126 NB(00206)—Nonmemory to BR:** The contents of the specified nonmemory location replace the contents of the BR. The C control flip-flops are not affected.

**Options:****RM**

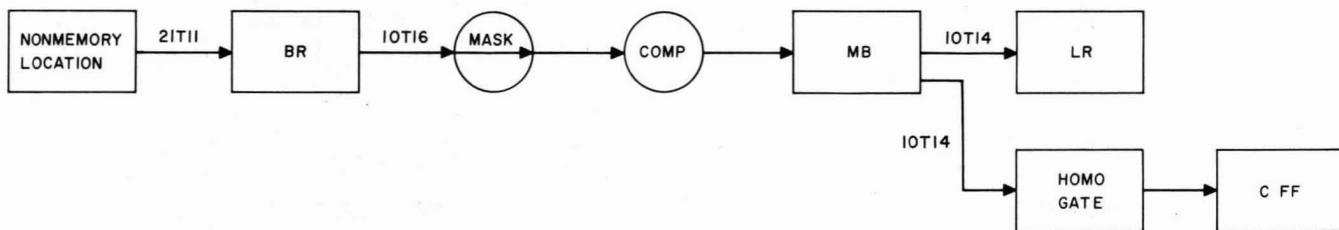
I(0V000;V = Index Reg No.)  
A(20000)  
W(00000 1)  
S(20000 1)

**LCJ**

(no masking  
takes place)

## SECTION 231-001-102

**8.127 NL(00324)—Nonmemory to LR:** The contents of the specified nonmemory location replace the contents of the BR and, after possible complementing, set the C control flip-flops and replace the contents of the LR.



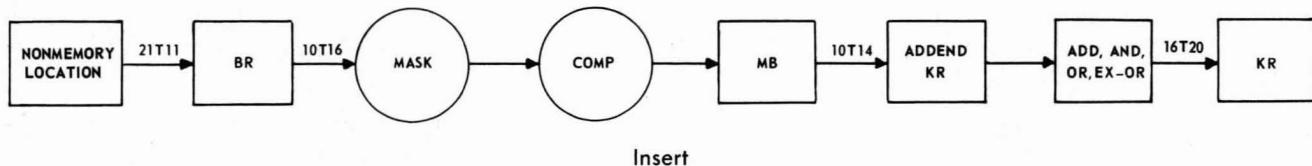
**Options:** RM

I(0V000;V = Index Reg No.)  
 A(20000)  
 W(00000 1)  
 S(20000 1)

LCJ

C(10000)

**8.128 NK(00354)—Nonmemory to Accumulator:** The contents of the specified nonmemory location replace the contents of the BR and, after possible product masking and/or complementing, replace the contents of the KR.



Insert

**Options:** RM

I(0V000;V = Index Reg No.)  
 A(20000)  
 W(00000 1)  
 S(20000 1)

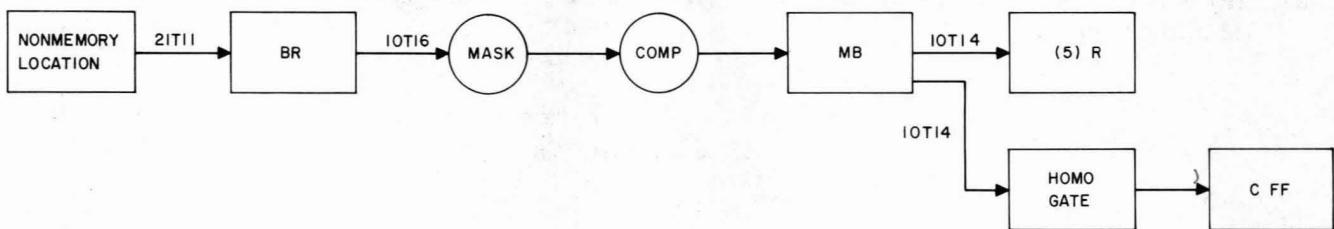
LCJ

PL(00000 2)  
 PS(00001)  
 C(10000)

**Restrictions:** The following instruction must not be

- (1) AKR or SKR
- (2) One of the following instructions with K in the R subfield: CWR, AWRP, TR instructions.

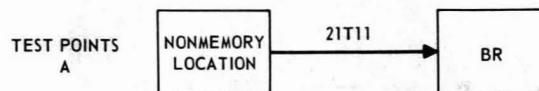
**8.129 NF(00226), NJ(00326), NX(00314), NY(00304), and NZ(00334)—Nonmemory to Register:** The contents of the specified nonmemory location replace the contents of the BR and, after possible product masking and/or complementing, set the C control flip-flops and replace the contents of the FR, JR, XR, YR, or ZR, whichever is specified in the operation code.

**Options:****RM****LCJ**

I(0V000;V = Index Reg No.)  
 A(20000)  
 W(00000 1)  
 S(20000 1)

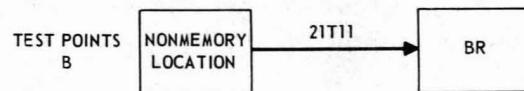
PL(00000 2)  
 PS(00001)  
 C(10000)

**8.130 NBT(A)(00200)—Nonmemory to Data BR; Test Points A:** Bits 0 through 11 of the resultant DAR number replace the contents of the address register in a CS identified by bits 12 through 17 of this resultant DAR number. Certain bits of the new contents of the address register activate group A test points, which generate a response that replaces the contents of the BR. (Bits 18 through 20 of the resultant DAR number must be 0.)

**Options:****RM**

I(0V000;V = Index Reg No.)  
 A(20000)  
 W(00000 1)  
 S(20000 1)

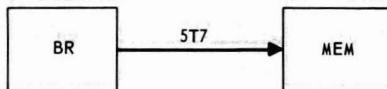
**8.131 NBT(B)(00204)—Nonmemory to Data BR; Test Points B:** Bits 0 through 11 of the resultant DAR number replace the contents of the address register in a CS identified by bits 12 through 17 of this resultant DAR number. Certain bits of the new contents of the address register activate group B test points which generate a response that replaces the contents of the BR. (Bits 18 through 20 of the resultant DAR number must be 0.)

**Options:****RM**

I(0V000;V = Index Reg No.)  
 A(20000)  
 W(00000 1)  
 S(20000 1)

## SECTION 231-001-102

**8.132 BG(00402)—BR to G Unit of Memory:** The contents of the BR replace the contents of location M in the G unit.



**Options:**

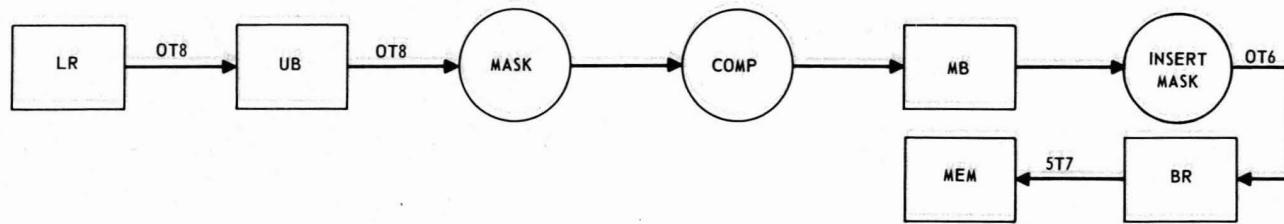
**RM**

I(0V000;V = Index Reg No.)  
A(20000)  
W(00000 1)  
S(20000 1)

**LCJ**

ES(00001)  
(no effect except to set LR)

**8.133 LG(00400)—LR to G Unit of Memory:** If neither ES nor PS masking is specified in the instruction, the contents of the LR, after possible complementing, replace or are insertion masked (EL) into the contents of the BR. The new contents of the BR replace the contents of location M in the G unit. If either ES or PS masking is called for in the instruction, the contents of the DA field replace the contents of the LR. The new LR contents, after possible complementing, replace or are insertion masked into the contents of the BR. The new contents of the BR replace the contents of location M in the G unit.



**Options:**

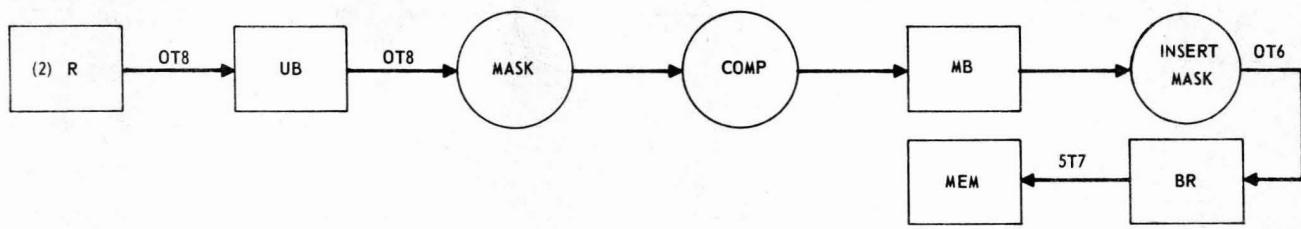
**RM**

I(0V000;V = Index Reg No.)  
A(20000)  
W(00000 1)  
S(20000 1)

**LCJ**

ES(00101)  
EL(00100 2)  
PS(00105)  
C(10000)

**8.134 FG(00410) and KG(00412)—Register to G Unit of Memory:** The contents of either the FR or KR, whichever is specified in the operation code after possible product masking and/or complementing, replace or are insertion masked into the contents of the BR. The new contents of the BR replace the contents of location M in the G unit.

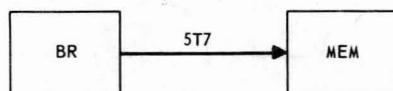
**Options:****RM**

I(0V000;V = Index Reg No.)  
 A(20000)  
 W(00000 1)  
 S(20000 1)

**LCJ**

EL(00100 2)  
 ES(00101)  
 PL(00104 2)  
 PS(00105)  
 C(10000)

**8.135 BH(00502)—BR to H Unit of Memory:** The contents of the BR replace the contents of location M in the H unit.

**Options:****RM**

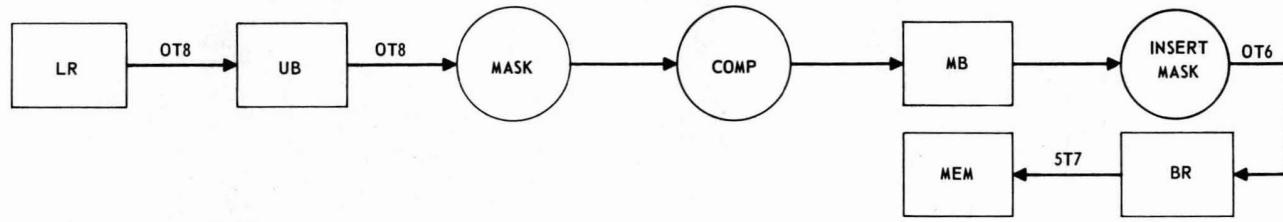
I(0V000;V = Index Reg No.)  
 A(20000)  
 W(00000 1)  
 S(20000 1)

**LCJ**

ES(00001)  
 (no effect except to set LR)

**8.136 LH(00500)—LR to H Unit of Memory:** If neither ES nor PS masking is specified in the instruction, the contents of the LR, after possible complementing, replace or are insertion masked (EL) into the contents of the BR. The new contents of the BR replace the contents of location M in the H unit. If either ES or PS masking is called for in the instruction, the contents of the DA field replace the contents of the LR. The new contents of the LR, after possible complementing, replace or are insertion masked into the contents of the BR. The new contents of the BR replace the contents of location M in the H unit.

**SECTION 231-001-102**



**Options:**

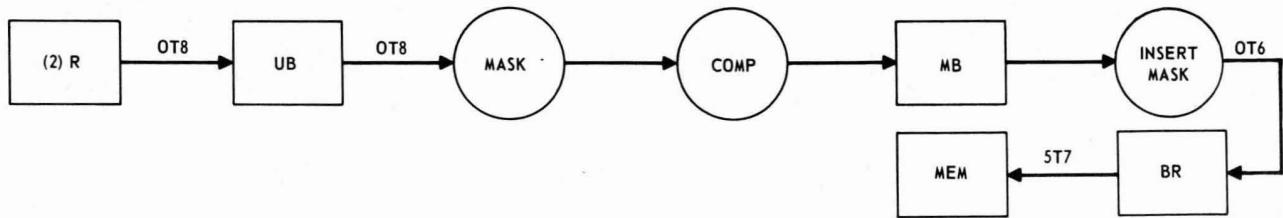
**RM**

I(0V000;V = Index Reg No.)  
A(20000)  
W(00000 1)  
S(20000 1)

**LCJ**

EL(00000 2)  
ES(00001)  
PS(00005)  
C(10000)

**8.137 FH(00510) and KH(00512)—FR or KR to H Unit of Memory:** The contents of either the FR or KR, whichever is specified in the operation code, after possible product masking and/or complementing, replace or are insertion masked into the contents of the BR. The new contents of the BR replace the contents of location M in the H unit.



**Options:**

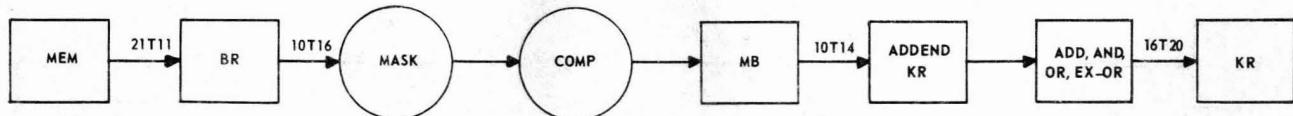
**RM**

I(0V000;V = Index Reg No.)  
A(20000)  
W(00000 1)  
S(20000 1)

**LCJ**

EL(00000 2)  
ES(00001)  
PL(00004 2)  
PS(00005)  
C(10000)

**8.138 GKC(00272)—G Unit of Memory to Accumulator (Corrected):** The contents of location M in the G unit, corrected by error detection and correction circuitry, replace the contents of the BR and, after possible product masking and/or complementing, replace the contents of the KR. The correction involved is the normal process occurring on all nonmaintenance instructions.

**Options:****RM****LCJ**

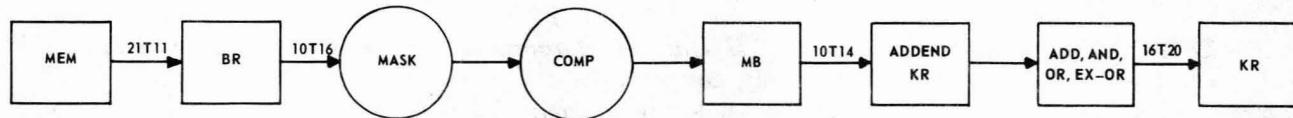
I(0V000 ;V = Index Reg No.)  
 A(20000)  
 W(00000 1)  
 S(20000 1)

PL(00000 2)  
 PS(00001)  
 C(10000)

**Restriction:** The following instruction must not be

- (1) AKR or SKR
- (2) One of the following instructions with K in the R subfield: CWR, AWRP, TR instructions.

**8.139 HKU(00276)—H Unit of Memory to Accumulator (Uncorrected):** The contents of location M in the H unit, uncorrected by error detection and correction circuitry, replace the contents of the BR and, after possible masking and/or complementing, replace the contents of the KR.

**Options:****RM****LCJ**

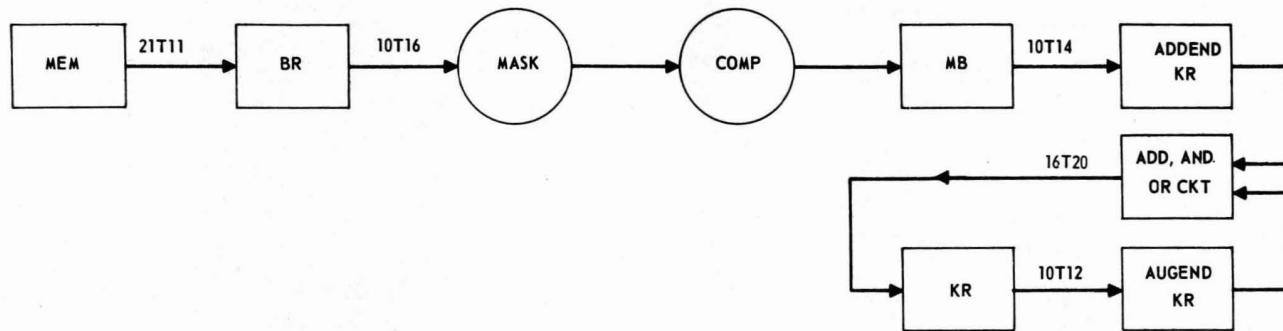
I(0V000 ;V = Index Reg No.)  
 A(20000)  
 W(00000 1)  
 S(20000 1)

PL(00000 2)  
 PS(00001)  
 C(10000)

**Restrictions:** The following instruction must not be

- (1) AKR or SKR
- (2) One of the following instructions with K in the R subfield: CWR, AWRP, TR instructions.

**8.140 XGKU(00252)—EXCLUSIVE OR of G Unit of Memory (Uncorrected) with Accumulator:** The contents of location M in the G unit, uncorrected by the error detection and correction circuitry, replace the contents of the BR and, after possible product masking and/or complementing, are combined by the EXCLUSIVE OR function with the contents of the KR to form a number that replaces the contents of the KR.

**Options:****RM**

I(0V000;V = Index Reg No.)  
 A(20000)  
 W(00000 1)  
 S(20000 1)

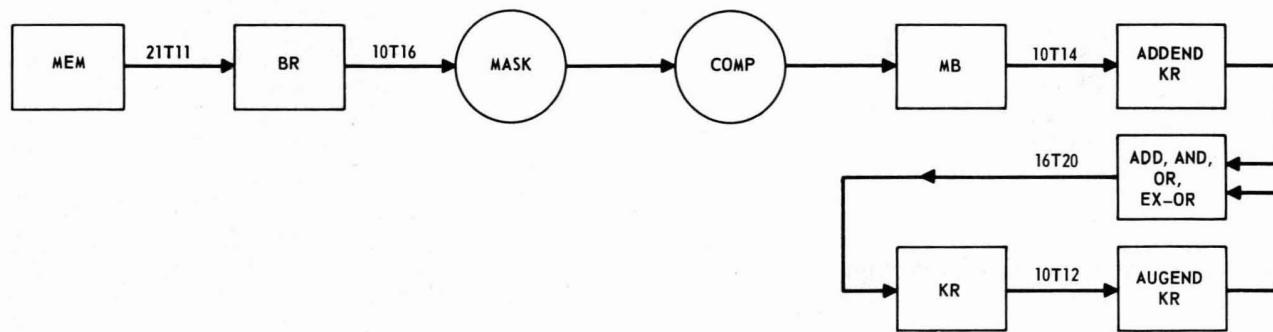
**LCJ**

PL(00000 2)  
 PS(00001)  
 C(10000)

**Restrictions:** The following instruction must not

- (1) Specify K in the R subfield
- (2) Be AKR or SKR.

**8.141 XHCK(00256)—EXCLUSIVE OR of H Unit of Memory (Corrected) with Accumulator:** The contents of location M in the H unit, corrected by the error detection and correction circuitry, replace the contents of the BR and, after possible product masking and/or complementing, are combined by the EXCLUSIVE OR functions with the contents of the KR to form a number that replaces the contents of the KR.

**Options:****RM**

I(0V000;V = Index Reg No.)  
 A(20000)  
 W(00000 1)  
 S(20000 1)

**LCJ**

PL(00000 2)  
 PS(00001)  
 C(10000)

**Restrictions:** The following instruction must not

- (1) Specify K in the R subfield
- (2) Be AKR or SKR.

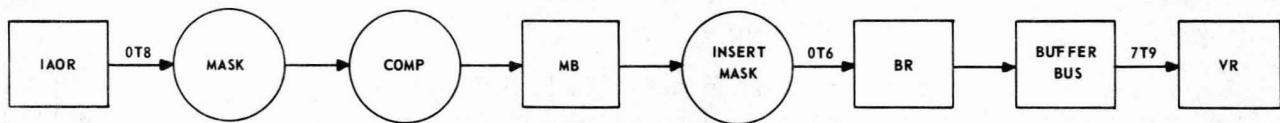
**8.142 WNPS(00006)—Word to Location N in PS:** Bit positions 0 through 11 of the resultant DAR number W represent six pairs of binary digits. Each pair is sent to its corresponding flip-flop at fixed nonmemory location N (specified by bits 12 through 15) in a PS identified by bit positions 16 through 19 of the same resultant DAR number W. Bit position 20 of the resultant DAR number W is ignored.



**Options:** RM

I (0V000 ;V = Index Reg No.)

**8.143 WV(00610)—Word to Miscellaneous Pulse Source Register:** The resultant DAR number W, after possible product masking and/or complementing, replaces or is insertion masked into the contents of the BR and is sent to the miscellaneous pulse source register (VR) located on the buffer bus. The VR consists of control leads and/or flip-flops for various control functions, such as stopping and starting a signal processor and resetting the 5-millisecond clock. Wherever a 1 is sent, the corresponding position in VR will generate a pulse. Where zeros are sent, the corresponding positions in the VR are unaffected. To stop the signal processor, a pulse is sent from the VR every cycle. To start the signal processor, this train of pulses from the VR is discontinued. Bit positions 0 through 18 of the VR are each assigned a control function. Bit positions 19 through 23 are unequipped. Only three bit positions of the VR can be read by a program instruction bit positions 0 (stop signal processor 0), 2 (stop signal processor 1), and 8 (E stop). When read into a CC register, the remaining bits will appear as zeros. This instruction does not affect the C control flip-flops.



**Options:** RM

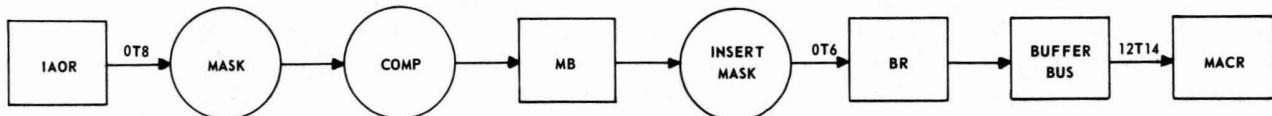
LCJ

I (0V000 ;V = Index Reg No.)

EL(00002)  
ES(00003)  
PL(00004)  
PS(00005)  
C(10000)

## SECTION 231-001-102

**8.144 EMMS(00630)—Mismatch Sampling:** The resultant DAR number W, possibly complemented, replaces the contents of the BR and are then gated into the match control register (MACR) located on the buffer bus.



Options:

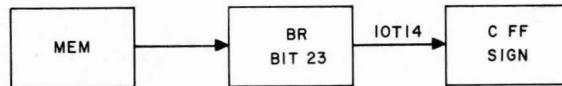
RM

LCJ

I(0V000;V = Index Reg No.)

C(10000)

**8.145 MBCS(00230)—Memory to BR and Parity to Sign of C Control Flip-Flops:** The contents of CS location M replace the contents of the BR. The overall parity bit stored in the 24th bit of this memory location (which when stored was computed on the address of location M and its contents) replaces the contents of the sign bit of the C control flip-flops.



Options:

RM

LCJ

I(0V000;V = Index Reg No.)  
A(20000)  
W(00000 1)  
S(20000 1)

S(00401)

## 9. GLOSSARY

**9.01** Alphabetic listing of instruction mnemonic codes are shown in Table A. Symbols and abbreviations used in the flow diagrams are shown in Fig. 3.

**Note:** Letters appearing in the operation codes (Table B) in the operation field have significant meaning. The definition of letters will not be provided in the glossary. The information is given in Parts 5, 6 and 7.

**9.02** Abbreviations other than those shown in 9.01 are given below:

AND—Logical product

CC—Central control

DA—Data or address

EL—Insertion masking option in the L subfield

ES—Insertion masking option in the L subfield

L—May be located in the encoded instruction column of the PR that indicates the DA field is a relocatable address

OR—Logical union

PIDENT—Program identification

PL—Product masking option in the L subfield

PR—Program listing

S—Logic register option in the L subfield

V—May be located in the encoded instruction column of the PR that indicates the

#—Sharp sign located in the PR.

PS—Product masking option in the L subfield  
location contains a reference into a transfer table