IBM 4700 Finance Communication System

Reference Digest

GC31-2034-1
File No. S370-30

IBM Corporation, Dept. 78C
1001 W.T. Harris Boulevard
Charlotte, NC, USA, 28257

IBM 4700 Finance Communication System

Reference Digest

GC31-2034-1
File No. S370-30

This publication is intended for programmers and operators working with the 4700 Finance Communication System.

You should be familiar with these publications from which this digest was compiled:

IBM 4700 Finance Communication System:

Controller Programming Library, GBOF-1387
Volume 1, GC31-2066
Volume 2, GC31-2067
Volume 3, GC31-2068
Volume 4, GC31-2069
Volume 5, GC31-2070
Volume 6, GC31-2071

Subsystem Operating Procedures, GC31-2032

IBM Corporation, Dept. 78C
1001 W.T. Harris Boulevard
Charlotte, NC, USA, 28257

Second Edition (November 1983)
The organization of this manual has changed from the previous edition.

Changes occur often to the information herein; before using this publication in connection with the operation of IBM systems, consult *IBM System/370 Bibliography of Industry and Application Programs,* GC20-0370, for the editions that are applicable and current.

References in this publication to IBM products, programs, or services do not imply that IBM intends to make these available in all countries in which IBM operates. Any reference to an IBM program product in this publication is not intended to state or imply that only IBM's program product may be used. Any functionally equivalent program may be used instead.

Publications are not stocked at the address given below; requests for IBM publications should be made to your IBM representative or to the IBM branch office serving your locality.

A form for reader's comments is provided at the back of this publication. If the form has been removed, address comments to the IBM Corporation, Information Development, Department 78C, 1001 W.T. Harris Boulevard, Charlotte, NC 28257, USA. IBM may use or distribute any of the information you supply in any way that it believes appropriate without incurring any obligation to you.

Contents

Figures

Chapter 1. 4700 Program Instructions

This chapter contains the program instruction in alphabetic order. The right hand column under *Volume* is the volume number of the *IBM 4700 Finance Communication System Controller Programming Library* manual where you can find the instruction.

Name	Operation	Operand	Volume
[label]	ADDFLD	reg1, { defcon2 | defld2 | (defrf2) | (reg2) | seg2,disp2,len2 }	1
[label]	ADDFLDL	reg1, { defcon2 | defld2 | (defrf2) | (reg2) | seg2,disp2,len2 }	1
[label]	ADDREG	reg1,reg2	1
[label]	ADDZ	{ defld1 | (defrf1) | (reg1) | (seg1,disp1,len1) } , { defcon2 | defld2 | (defrf2) | (reg2) | seg2,disp2,len2 }	1
[label]	AND	{ defld1 | (defrf1) | (reg1) | seg1,disp1 } , { defcon2 | defld2 | (defrf2) | (reg2) | seg2,disp2,len2 }	1
[label]	ANDI	{ defld1 | (defrf1) | (reg1) | seg1,disp1 } , immdata2	1
[label]	APBDUMP	{ { seg | (seg, . . .) } [,FILE] | FILE } [,ID=dumpid]	1
[label]	APCALL	{ defcon2 | defld2 | (defrf2) | (reg2) | seg2,disp2 } [,WAIT={Y | N}]	1
[label]	APOPT	[RELOC={Y | N}] [,SPLIT={Y | N}] [,DIRNAME= name,{NEW | OLD}] [,DISP={NEW | OLD}] [,YL2={Y | N}] [,INDEX={nnn | 0}] [,DISP16={Y | N}]	1
[label]	APRETURN		1
[label]	ASSIGN	{ defld2 | seg2,disp2 | (reg2) | (defrf2) }	3
[label]	BEGIN	APBNM= { name | (name [,{vn | 1}]) } ,DATE=mmddyy [,PC=label] [,DEL=label] [,STP=label] [,APENTRY=label] [,API=label] [,ATD=label] [,ACP=label] [,AST=label] [,NUMOVLY=n] [,ATM=label] [,INSNAME=name] [,DSECT={Y | N}]	1

Name	Operation	Operand	Volume
[label]	BRAN	[ccmask,] branch address	1
[label]	BRANL	[ccmask,] branch address [,reg]	1
[label]	BRANLR	[ccmask,] reg1 [,reg2]	1
[label]	BRANR	[ccmask,] reg	1
[label]	BRANX	reg,branch address	1
[label]	CAFLD	reg1, { defcon2 / defld2 / (defrf2) / (reg2) / seg2,disp2,len2 }	1
[label]	CAFLDL	reg1, { defcon2 / defld2 / (defrf2) / (reg2) / seg2,disp2,len2 }	1
[label]	CAREG	reg1,reg2	1
[label]	CCDI	{ defcon1 / defld1 / (defrf1) / (reg1) / seg1,disp1 },immdata2	1
[label]	CCFLD	seg1, { defcon2 / defld2 / (defrf2) / (reg2) / (seg2,disp2,len2 }	1
[label]	CCFXD	{ defcon1 / defld1 / (defrf1) / (reg1) / seg1,disp1 }, { defcon2 / defld2 / (defrf2) / (reg2) / seg2,disp2,len2 }	1
[label]	CCSEG	seg1,seg2	1
[label]	CLOSSESS	TYPE= { TERM / LU / HOST }	3
[label]	COBLCALL	ap, { defld1 / (defrf1) / (reg1) }, { defld2 / (defrf2) / (reg2) } [, { defld3 / (defrf3) / (reg3) } , . . .]	1
[label]	COMP	{ defld2 / (defrf2) / (reg2) / seg2,disp2 }	1
[label]	COMPDKT	{ defld2 / (defrf2) / (reg2) / seg2,disp2 }	2

Name	Operation	Operand	Volume
[label]	COMPTB	seg1, { defld2 / (defrf2) / (reg2) / seg2,disp2 }	1
[label]	COMPZ	{ defcon1 / defld1 / (defrf1) / (reg1) / seg1,disp1,len1 }, { defcon2 / defld2 / (defrf2) / (reg2) / seg2,disp2,len2 }	1
[label]	CONFIRM	RESP= { ERR / OK }	3
	COPY	copyfilename	1
[label]	CRETN	{ defld2 / (defrf2) / (reg2) / seg2,disp2 }	1
[label]	DCACTL	{ defcon2 / defld2 / seg2,disp2 / (reg2) / (defrf2) }	4
[label]	DECIPHER	{ defld2 / seg2,disp2 / (reg2) / (defrf2) }	5
[label]	DECODE	seg1, { defcon2 / defld2 / (defrf2) / (reg2) / seg2,disp2 } ,DES	5
[label]	DECOMP	{ defld2 / (defrf2) / (reg2) / seg2,disp2 }	1
[label]	DECOMPTB	seg1, { defld2 / (defrf2) / (reg2) / seg2,disp2 }	1
[label]	DEFASEP		3
[label]	DEFCODE	LOCE=label [,LOCD=label] [,SEFPASMB= { Y / N }]	3
[label]	DEFCON	data1 [,data2] . . . [,datan] [,0]	1
label	DEFDEL	(char,mask) , (char,mask) , . . .	1
[label]	DEFDMP	SEG=n	1
[label]	DEFLD	seg, [disp / (abs exp) / *] , { len / (abs exp) } [,BDY=HALF]	1

Name	Operation	Operand	Volume
[label]	DEFLINK	OUT=seg, IN=seg,WKSEG= { (seg, disp) | seg } , ,LKREG=reg WKREG=reg [,DREG=reg] [,SEN1=xxxxyyyy] [,SEN2=xxxxyyyy] [,SEN3=xxxxyyyy] [,SEPASMB= { N | Y }]	3
label	DEFRF	reg, [disp | (abs exp) | (label)], {len | (abs exp)}	1
[label]	DEFSTOR	SEGSIZE= ({seg0 | *}, {seg2 | *} , . . . , {seg13 | *}) [,MAXSTOR=n] [,USE= {STATIC | DYNAMIC}]	1
[label]	DELETE	PLR, { defld2 | (defrf2) | (reg2) | seg2,disp2,len2 | seg2 }	2
[label]	DEVPARM	{ lda | mnem } , { defld2 | (defrf2) | seg2,disp2,len2 | (reg2) | seg2 | (defcon2) } [, { WAIT | NOWAIT }] [, { SETRET= { RETURN [,EXP= { Y | N }] } | SET } | EXTRACT= { RETURN | SET }]	4
[label]	DIVFLD	reg1, { defcon2 | defld2 | (defrf2) | (reg2) | seg2,disp2,len2 }	1
[label]	DIVFLDL	reg1, { defcon2 | defld2 | (defrf2) | (reg2) | seg2,disp2,len2 }	1
[label]	DIVREG	reg1,reg2	1
[label]	DIVZ	{ defld1 | (defrf1) | (reg1) | seg1,disp2,len1 } , { defcon2 | defld2 | (defrf2) | (reg2) | seg2,disp2,len2 }	1
[label]	DPOOL	{ GET | GETX | RETURN | QUERY } , { defld2 | seg2,disp2 | (reg2) | (defrf2) }	4
[label]	DTACCESS	{ defld2 | (defrf2) | (reg2) | seg2,disp2, } [,WAIT= {Y | N}]	1

Name	Operation	Operand	Volume
[label]	DTAFREE	{ defld2 | (defrf2) | (reg2) | seg2,disp2 }	1
[label]	EDIT	seg1,seg2,{label 3 | (reg3)}	1
[label]	ENCIPHER	{ defld2 | seg2,disp2 | (reg2) | (defrf2) }	5
[label]	ENCODE	seg1,{ defcon2 | defld2 | (defrf2) | (reg2) | seg2,disp2 },DES	5
[label]	ENDINIT		1
[label]	ENDOVLY		1
	ENDSEG		1
label	EQUATE	{ value | (abs exp) }	1
[label]	ERRLOG	seg1,{ defld2 | (defrf2) | (seg2) | seg2,disp2 }	1
[label]	EXOR	{ defld1 | (defrf1) | (reg1) | seg1,disp1 },{ defcon2 | defld2 | (defrf2) | (reg2) | seg2,disp,len2 }	1
[label]	EXORI	{ defld1 | (defrf1) | (reg1) | seg1,disp1 },immdata2	1
[label]	EXPS	seg1	1
[label]	FCLENTER	saveaddress [,a1,a2,...,a254]	1
[label]	FCLEXIT	saveaddress [,a1,a2,...,a254]	1
[label]	FINDAP	{ defld2 | (defrf2) | (reg2) | seg2,disp2 }	1
	FINISH		1
[label]	FORMDKT	{ defld2 | (defrf2) | (reg2) | seg2,disp2, }	2
[label]	GETDMS	{ seg2 | (reg2) }	4
[label]	GETFLD	{ seg2 | (reg2) }	4

Name	Operation	Operand	Volume
[label]	INITSEG	[id,] ({seg,disp | defld} ,data) [, ({seg,disp | defld} ,data) . . .]	1
[label]	INOR	{ defld1 / (defrf1) / (reg1) / seg1,disp1 } , { defcon2 / defld2 / (defrf2) / (reg2) / seg2,disp2,len2 }	1
[label]	INORI	{ defld1 / (defrf1) / (reg1) / seg1,disp1 } , immdata2	1
[label]	INTMR	{ defcon2 / defld2 / (defrf2) / (reg2) / seg2,disp2 }	1
[label]	JUMP	[ccmask,] branch address	1
[label]	KEYGEN	{ defld2 / seg2,disp2 / (reg2) / (defrf2) }	5
[label]	LCHAP	[ON / OFF / NEXT]	1
[label]	LCHECK	lda , [,TIO]	4
[label]	LCHECK	CP[,TIO]	3
[label]	LCHECK	{ PLR / DSK }	2
[label]	LCHECK	ST [,TIO]	1
[label]	LCONVERT	{ defld1 / (defrf1) / (reg1) / seg1,disp1 } , { defld2 / (defrf2) / (reg2) / seg2,disp2,len2 } , { TOBITS / TOBYTES }	1
[label]	LDDI	reg1,immdata2	1
[label]	LDFLD	reg1, { defcon2 / defld2 / (defrf2) / (reg2) / seg2,disp2,len2 }	1
[label]	LDFLDC	reg1, { defcon2 / defld2 / (defrf2) / (reg2) / seg2,disp2,len2 }	1
[label]	LDFLDL	reg1, { defcon2 / defld2 / (defrf2) / (reg2) / seg2,disp2,len2 }	1

Name	Operation	Operand	Volume
[label]	LDFP	reg1,seg2	1
[label]	LDKT	{ defcon2 | defld2 | (defrf2) | (reg2) | seg2 | seg2,disp2 }	2
[label]	LDLN	reg1,seg2	1
[label]	LDRA	reg1, { defcon2 | defld2 | (defrf2) | (reg2) | seg2,disp2,len2 }	1
[label]	LDREG	reg1,reg2	1
label	LDSECT	{ SDL={ seg | defld } | BASE={ reg | * } } [,DISP=number]	1
[label]	LDSEG	reg1,seg2	1
[label]	LDSEGC	reg1,seg2	1
[label]	LDSEGLN	reg1,seg2	1
[label]	LDSFP	reg1,seg2	1
[label]	LEJECT		1
	LEND	[PRINT={ OFF | ON }]	1
[label]	LEXEC	reg1,addr [,len | 0]	1
[label]	LEXIT		1
[label]	LHRT	{ defcon2 | defld2 | (defrf2) | (reg2) | seg2,disp2 }	1
[label]	LIFOFF	{ defld2 | (defrf2) | (reg2) | seg2,disp2 },bitnum,branch address [,ELSE=SETOFF]	1
[label]	LIFON	{ defld2 | (defrf2) | (reg2) | seg2,disp2 },bitnum,branch address [,ELSE=SETON]	1
[label]	LLOAD	[reg1], { defcon2 | defld2 | (defrf2) | (reg2) | seg2,disp2 } [,PARM=EXP]	1

Name	Operation	Operand	Volume
[label]	LMERGE	{ seg2 / defld2 / (defrf2) / (reg2) / seg2,disp2 }	1
[label]	Ln	{ * / label / label,length } [,REDEF=label]	1
[label]	LPOST		1
[label]	LREAD	lda,{ seg2 / defld2 / seg2,disp2,len2 / (reg2) / (defrf2) } [{NOTRACK / TRACK}] [{NOWAIT / WAIT}]	4
[label]	LREAD	CP,{ seg2 / defld2 / seg2,disp2 / (reg2) / (defrf2) }	3
[label]	LREAD	ST,{ seg2 / defld2 / (defrf2) / (reg2) / seg2,disp2,len2 }	1
[label]	LREAD	{ TFn / C / P / PLR / L / A / DSID / PBN } , { defld2 / (defrf2) / (reg2) / seg2 / seg2,disp2,len2 }	2
[label]	LRECEIVE	[CANCEL={NO / YES}]	3
[label]	LRETURN		1
[label]	LSEEK	{ seg1,seg2 / seg1,table / (reg1),(reg2) } [,match]	1
[label]	LSEEKP	{ defcon2 / defld2 / (defrf2) / (reg2) / seg2,disp2 }	1

Name	Operation	Operand	Volume
[label]	LSEEKPL	[OPTION=({TBLAD / PRMAD} , {SEQ / BIN} , {NOINST / INST} , {NOENT / ENT} , {EQ / BEL / BEG / BELG} , {NONSI / NSI})] [,NSIRTN= {0 / reg}] [,ARGU= {defld / (seg,disp,len)}] [,SEARCH= {defld / table / seg,disp}] [,MATCH=match] [,ENTRY= {defld / (seg,disp,len)}]	1
[label]	LSEND	[RESP= {NO / DEF / EX}] [,DATA= {CTL / NORM}] [,BRCKT= {NONE / MARK / NORM}] [,CHNGDIR= {YES / NO}]	3
[label]	LSETOFF	{defld1 / (defrf1) / (reg1) / seg1,disp1} ,bitnum	1
[label]	LSETON	{defld1 / (defrf1) / (reg1) / seg1,disp1} ,bitnum	1
[label]	LSORT	{seg2 / defcon2 / defld2 / (defrf2) / (reg2) / seg2,disp2}	1
	LSPACE	[1 / 2 / 3]	1
[label]	LTIME	{SET / GET / ADJ} , {defld2 / (defrf2) / (reg2) / seg2,disp2}	1

Name	Operation	Operand	Volume
[label]	LTIMET	[MONTHS= ({ m1 / JAN } , { m2 / FEB } , ... , { m12 / DEC })] [DAYS= ({ d1 / SUN } , { d2 / MON } , ... , { d7 / SAT })] [IND= ({ AM / m } , { PM / e })]	1
[label]	LTIMEV	seg1, { defcon2 / defld2 / (defrf2) / (reg2) / (seg2,disp2,len2 } [,ltimet]	1
[label]	LTRT	{ defld2 / (defrf2) / (reg2) / seg2,disp2 }	1
[label]	LTRTBEG	[RANGE=([low code],[high code],[mode])]	1
[label]	LTRTENT	(code,[ochr],[func],[mask], [addr],[inlen]) [,(. . .)]	1
[label]	LTRTGEN		1
[label]	LWAIT		1
[label]	LWRITE	lda, { defcon1 / defld1 / seg1,disp1,len1 / (reg1) / seg1 / (defrf1) } [, { NOWAIT / WAIT }]	4
[label]	LWRITE	CP, { seg2 / defld2 / seg2,disp2,len2 / (reg2) / (defrf2) / defcon2 }	3
[label]	LWRITE	ST, { defcon2 / defld2 / (defrf2) / (reg2) / seg2 / seg2,disp2,len2 }	1
[label]	LWRITE	{ TFn / L / PLR } , { defcon2 / defld2 / (defrf2) / (reg2) / seg2,disp2,len2 / seg2 }	2
[label]	MACGEN	{ defld2 / seg2,disp2 / (reg2) / (defrf2) }	5
label	MASK	[fill,] t'characters'	1

Name	Operation	Operand	Volume
abel	MOD	modnum, {wt1 / (wt1,wt2, . . . ,wtn)} [MA / ,AP / UN]	1
label]	MODCHK	seg1,reg2, {label / (reg3)}	1
label]	MPYFLD	reg1, {defcon2 / defld2 / (defrf2) / (reg2) / seg2,disp2,len2}	1
label]	MPYFLDL	reg1, {defcon2 / defld2 / (defrf2) / (reg2) / seg2,disp2,len2}	1
label]	MPYREG	reg1,reg2	1
label]	MPYZ	{defld1 / (defrf1) / (reg1) / seg1,disp1,len1} , {defcon2 / defld2 / (defrf2) / (reg2) / seg2,disp2,len2}	1
label]	MVCZ	{defld1 / (defrf1) / (reg1) / seg1,disp1,len1} , {defcon2 / defld2 / (defrf2) / (reg2) / seg2,disp2,len2}	1
label]	MVDI	{defld1 / (defrf1) / (reg1) / seg1,disp1} , immdata2	1
label]	MVFLD	seg1, {defcon2 / defld2 / (defrf2) / (reg2) / seg2,disp2,len2}	1
[label]	MVFXD	{defld1 / (defrf1) / (reg1) / seg1,disp1} , {defcon2 / defld2 / (defrf2) / (reg2) / seg2,disp2,len2}	1
[label]	MVSEG	seg1,seg2	1
label]	OPENSESS	SESSID= (iden[, {len / 8}]) [,TYPE= {LU / HOST}] [,RESP= {RRN / FME}]	3

Name	Operation	Operand	Volume
label	OVLYSEC	ORIGIN= {org | ([inst-org] [,const-org])} [,OVLYEP=entry] [,VERSION= {version | 0}] [,INSTR= {N | Y}] [,INSNAME=name]	1
[label]	PAKFLD	seg1, {defcon2 | defld2 | (defrf2) | (reg2) | seg2,disp2,len2}	1
[label]	PAKSEG	seg1,seg2	1
[label]	PAUSE		1
[label]	PINTRANS	{defld1 | seg1,disp1 | (reg1) | (defrf1)}, {defld2 | seg2,disp2 | (reg2) | (defrf2)}	5
[label]	PINVERIF	{defld1 | seg1,disp1 | (reg1) | (defrf1)}, {defld2 | seg2,disp2 | (reg2) | (defrf2)}	5
	PLPCMD	{EDIT | NOEDIT | DELETE | NODELETE}	1
[label]	PRINTI	[ON | OFF]	1
[label]	PUTDMS	{seg2 | (reg2)}	4
[label]	PUTFLD	{seg2 | (reg2)}	4
	REBASE	label	1
[label]	REPLACE	{A | CR | PBN | TFn | C | P | DSID | PLR} {defcon2 | defld2 | (defrf2) | (reg2) | seg2,disp2,len2 | seg2}	2
[label]	RFMK	{defld2 | seg2,disp2 | (reg2) | (defrf2)}	5
[label]	RTMK	{defld2 | seg2,disp2 | (reg2) | (defrf2)}	5
	SAVEBASE	label	1

Name	Operation	Operand	Volume
[label]	SCALE	seg1, { defld2 | (defrf2) | (reg2) | seg2,disp2 }	1
[label]	SCRPAD	{ defld2 | (defrf2) | (reg2) | seg2,disp2 }	1
[label]	SECTION	{ DUMMY | END | INSTR | CONST | AUTO }	1
[label]	SEGCODE	NAME=name [,VERSION= { version | 1 }] [,INSNAME=name]	1
[label]	SEGCOPY	seg1, { defcon2 | defld2 | (defrf2) | (reg2) | seg2,disp2 } , { TO | FROM }	1
[label]	SELECT		1
[label]	SETDSKT	{ defcon2 | defld2 | (defrf2) | (reg2) | seg2,disp2 }	2
[label]	SETFLDI	{ defld1 | (defrf1) | (reg1) | seg1,disp1,len1 } , immdata2	1
[label]	SETFPL	{ defld seg, { [±] F | L } seg, { [±] F | L } { [±] n | (reg2) | S(reg2) } seg, [[±] n | (reg1) | S(reg1) | +0] [{ [±] n | (reg2) | S(reg2) | +0 }] }	1
[label]	SETSFP	{ defcon defld seg [{ [±] n | (reg1) | S(reg1) | [±] F | L }] }	1
[label]	SETX	{ OFF | ON }, { defld2 | (defrf2) | (reg2) | seg2,disp2 }	1

Name	Operation	Operand	Volume
[label]	SETXREG	{(seg1,[reg1]) / (seg2[,reg2])} [, {(seg2[,reg2]) / DUP}]	1
[label]	SHIFTL	reg1,count	1
[label]	SHIFTR	reg1,count	1
[label]	SIGNAL	lda, {defcon1 / defld1 / seg1,disp1 / (reg1) / (defrf1)}	4
label	SINIT		1
[label]	**STATS**	**{defld2 / (defrf2) / (reg2)}**	1
[label]	STFLD	reg1, {defld2 / (defrf2) / (reg2) / seg2,disp2,len2}	1
[label]	STFLDC	reg1, {defld2 / (defrf2) / (reg2) / seg2,disp2,len2}	1
[label]	STOVLY	[C / I / A]	1
[label]	STPLNK	{defld2 / seg2,disp2 / (reg2) / (defrf2) / defcon2}	3
[label]	STPLPS	**{defcon2 / defld2 / seg2,disp2 / (reg2) / (defrf2)}**	4
[label]	STRLNK	{defld2 / seg2,disp2 / (reg2) / (defrf2) / defcon2}	3
[label]	STRLPS	**{defcon2 / defld2 / seg2,disp2 / (reg2) / (defrf2)}**	4
[label]	STSEG	reg1,seg2	1
[label]	STSEGC	reg1,seg2	1
[label]	SUBFLD	reg1, {defcon2 / defld2 / (defrf2) / (reg2) / seg2,disp2,len2}	1

Name	Operation	Operand	Volume
[label]	SUBFLDL	reg1, { defcon2 / defld2 / (defrf2) / (reg2) / seg2,disp2,len2 }	1
[label]	SUBREG	reg1,reg2	1
[label]	SUBZ	{ defld1 / (defrf1) / (reg1) / seg1,disp1,len1 }, { defcon2 / defld2 / (defrf2) / (reg2) / seg2,disp2,len2 }	1
[label]	SWAPTT	{ defld2 / seg2,disp2 / (reg2) / (defrf2) }	4
[label]	TABLE	element1[,element2 . . . ,elementn], LNG=nnn [,SRT=n] where: element is { data / (data1,data2 . . . ,datan) }	1
[label]	TABLE	element1[,element2 . . . ,elementn], LNG=nnn [,SRT=n] where: element is { (data,addr) / (data1,data2 . . . ,datan,addr) }	1
[label]	TESTX	{ defld2 / (defrf2) / (reg2) / seg2,disp2 }	1
[label]	TMEXIT		3
[label]	TSTMSK	{ defcon1 / defld1 / (defrf1) / (reg1) / seg1,disp1 }, { defcon2 / defld2 / (defrf2) / (reg2) / seg2,disp2,len2 }	1
[label]	TSTMSKI	{ defcon1 / defld1 / (defrf1) / (reg1) / seg1,disp1 }, immdata2	1
[label]	UPKFLD	seg 1, { defcon2 / defld2 / (defrf2) / (reg2) / seg2,disp2,len2 }	1

Name	Operation	Operand	Volume
[label]	UPKSEG	seg1,seg2	1
	USEBASE	label, { reg | * }	1
[label]	VERIFY	seg1, { defcon2 / defld2 / (defrf2) / (reg2) / seg2,disp2 }	1
[label]	VIEW	{ defld2 / (defrf2) / (reg2) / seg2,disp2 }	1
[label]	WRTI	{ lda / mnemonic } ,immdata2 [, { WAIT / NOWAIT }]	4

Chapter 2. 4700 Configuration Macros

For more information on the configuration macros refer to the *IBM 4700 Finance Communication System Controller Programming Library* manuals.

Name	Operation	Operand
[label]	APLIST	(apname1,vn[,T|R])
		[,(apname2,[vn][,T|R])][(...)]
label	CHARDEF	X'a.....ghi.....s',
		X'a.....ghi.....s',
		X'a.....ghi.....s',
		X'a.....ghi.....s',
		X'a.....ghi.....s',
		X'a.....ghi.....s',
		X'a.....ghi.....s',
		X'a.....ghi.....s',
		X'a.....ghi.....s',
		X'a.....ghi.....s',
		X'a.....ghi.....s',
		X'a.....ghi.....s',
		X'a.....ghi.....s',
		X'a.....ghi.....s',
		X'a.....ghi.....s',
		X'a.....ghi.....s',
		,DEVICE= {3616 | 4710 |
		(4720,{R | Q})}

Name	Operation	Operand
[label]	COMLINK	[DCL = {nnnn | 9600}]
		[,WRAP = {Y | N}]
		[,CNL = {nnn | 72}]
		[,CNB = {n | 1}]
		[,ACB = ([DCE] [,DSL] [,PRT] [,SWM])]
		[,WRT = {n | 1}]
		[,CTG = [{mm | 0}] [: {0 | ss | 20}]]
		[,TYPE = ([,4502] [,4850] [,1422] [,5656])]
		[,VERSION = {A | B}]
		[,FLAG = {AA | AC | DC}]
		[,RETRIES = {n | 5}]
		[,DEL1 = {ss.t | 03.0}]
		[,DEL2 = {[mm] [:ss] | 03.00}]
		[,SELSEQ = '{c1} [...cn]']
		[,OPTIONS = (option [,...])]
		[,BSC = ({X'gpc' | C'gpc'}, [{m | 256}],[{n | 5}] [,DEF])]
		[,BSCOPT = {(opt1[,opt2])}]

Name	Operation	Operand
label	DCAPORT	(labeldca[,poolname]),PORT = addr
label	DCATRTBL	{scancode,X'n'} [,...],
		TYPE = {ALT | SHIFT},
		DEVICE = 4704,
		CASE = {1 | 2 | 3 | 4}, IND = {Y | N}
label	DCA3262	[PS = {n | 0}] [,TM = {n | 0}]
		[,SWITCH = {Y | N}]
		[,CHAIN = {Y | N}]
		[,UDDPC1 = {value | 00}]
label	DCA3278	MODEL = 8,OUTRTBL = labelout
		[,TRTBHDR = labeltrt]
		[,EOMSET = {X'xx' | X'FF'}]
		[,CURSOR = {Y | N | R}]
		[,MSTRTBL = msintrt]
		[,ERTLS = {Y | N}]
		[,OPTIONS = option]
		[,UDDccc = {value | 00}]
label	DCA3279	MODEL = {8A | 8B},
		OUTRTBL = labelout
		[,TRTBHDR = labeltrt]
		[,EOMSET = {X'xx' | X'FF'}]
		[,CURSOR = {Y | N | R}]
		[,MSTRTBL = msintrt]
		[,ERTLS = {Y | N}]
		[,OPTIONS = option]
		[,UDDccc = {value | 00}]

Name	Operation	Operand
label	DCA3287	[PS = {n | 0}] [,TM = {n | 0}]
		[,SWITCH = {Y | N}]
		[,UDDPC1 = {value | 00}]
label	DCA5210	[PS = {n | 0}] [,TM = {n | 0}]
		[,SWITCH = {Y | N}]
		[,UDDPC1 = {value | 00}]
label	DCA4704	MODEL = {21 | 22},
		OUTRTBL = label-out
		[,TRTLIST = label-list]
		[,TRTBHDR = label-trt]
		[,MSTRTBL = (msintrt,msoutrt)]
		[,PINTBL = pinintrt]
		[,EOMSET = (X'xx' | X'FF')]
		[,CURSOR = {N | Y | R}]
		[,ERTLS = {Y | N }]
		[,OPTIONS = option]
		[,FEATURE = MSRE]
		[,UDDccc = {value | 00}]
label	DEFADDR	(labeldev,sa1[,sa2] ... [,sa15] [,poolname])
		,ADDR = (looplabel,tla,mod)
[label]	DEFSBF	r1,c1
	DEFSPA	SIZE = n [,ID = id]

Name	Operation	Operand
label	DEV3604	MODEL = {1 | 2 | 3 | 4 | 5 | 6 | 7}
		[,OUTRTBL = labelout]
		[,TRTBHDR = labeltrt]
		[,OPTIONS = {(option1,option2,...) }]
		[,MSLITE = {1 | 2 | 3}]
		[,EOMSET = { X'xx' / X'FF' }]
		[,PINTBL = labelpin]
		[,MSTRTBL = (msintrt,msoutrt)]
		[,CURSOR = {Y | N | R}]
		[,ERTLS = {Y | N}]
		[,PRTY = {Y | N}]
		[,UDDccc = {value | 00}]
label	DEV3606	TRTBHDR = labeltrt,
		OUTRTBL = labelout
		[,MSTRTBL = labelms]
		[,PRTY = {Y | N}]
		[,UDDFST = {value | 00}]
label	DEV3608	TRTBHDR = labeltrt,
		OUTRTBL = (labelout1, labelout2)
		[,MSTRTBL = labelms]
		[,AVGML = {n | 100}]
		[,FORMWD = {n | 50}]
		[,OFFSET = { ({begin | 6} , {pages | 0}) }]
		[,OPTIONS = OCR]
		[,PRTY = {Y | N}]
		[,UDDccc = {value | 00}]

Name	Operation	Operand
label	DEV3610	MODEL = {1 | 2 | 3 | 4 | 5 | 12 | 13}
		,OUTRTBL = labelout
		[,PS = {n | 0}] [,WL = {n | 0}]
		[,SHARED = {Y | N}]
		[,CONSHR = {Y | N}]
		[,FT = {CTAUTO | CT | CN}]
		[,TIMEOUT = {5 | n}]
		[,OPTIONS = MSG1586]
		[,DEBLSE = {n | 10}]
		[,PRTY = {Y | N}]
		[,UDDPC1 = {value | 00}]
label	DEV3611	MODEL = {1 | 2}
		,OUTRTBL = labelout
		[,PS = {n | 1}]
		[,CFOLD = ({begin | 0 }[, {skip | 2}])]
		[,OFFSET = ({ steps | 0 }[, {lines | 0}])]
		[,SHARED = {Y | N}]
		[,OPTIONS = MG1586]
		[,PRTY = {Y | N}]
		[,UDDPC1 = {value | 00}]

Name	Operation	Operand
label	DEV3612	MODEL = {1 | 2 | 3 | 12 | 12P | 13 | 13P}
		,OUTRTBL = (labelout1,labelout2)
		[,PS = {({n | 0} , {n | 1})}]
		[,WL = {n | 0}]
		[,CFOLD = ({begin | 0 }[, {skip | 2}])]
		[,OFFSET = ({ steps | 0 }[, {lines | 0}])]
		[,SHARED = {Y | N}]
		[,CONSHR = {Y | N}]
		[,FT = {CTAUTO | CT | CN}]
		[,TIMEOUT = {5 | n}]
		[,OPTIONS = MG1586]
		[,DEBLSE = {n | 10}]
		[,PRTY = {Y | N}]
		[,UDDccc = value | 00]
label	DEV3614	[OPTIONS = {9495 | 9494}]
		[,UDDCTF = { value | 00}]
label	DEV3615	MODEL = {1 | 2}
		,OUTRTBL = labelout
		[,PS = {n | 0}] [,WL = {n | 0}]
		[,SHARED = {Y | N}]
		[,CONSHR = {Y | N}]
		[,FT = {CT | CTAUTO | CN}]
		[,OPTIONS = MG1586]
		[,PRTY = {Y | N}]
		[,UDDPC1 = { value | 00}]

Name	Operation	Operand
label	DEV3616	[OUTBHDR = label]
		[,PS = ({n | 0} , {n | 1})]
		[,WL = {n | 0}]
		[,CFOLD = ({begin | 0 }[, {skip | 2}])]
		[,OFFSET = ({steps | 0 }[, {lines | 0}])]
		[,SHARED = {Y | N}]
		[,CONSHR = {Y | N}]
		[,KEY = {N | Y} , {N | Y}]
		[,NL = {AUTO | SPECIFY}]
		[,PBFOLD = {HORZ | VERT}]
		[,MARGIN = {n | 0}]
		[,LPI = {5 | 6}]
		[,CPI = ({0 | 1} , {0 | 1})]
		[,LINE = ({n | 47/57} , {n | 83/100})]
		[,DEFTRT = labeltrt]
		[,DEFCHAR = lblchar]
		[,PRTY = {Y | N}]
		[,UDDccc = {value | 00}]
label	DEV3624	[UDDCTF = {value | 00}]

Name	Operation	Operand
label	DEV4704	MODEL = {11 | 12},
		OUTRTBL = labelout,
		CHARSET = {US | KATAKANA}
		[,TRTBHDR = labeltrt]
		[,PRTY{Y | N}]
		[,OPTIONS = {(option1,option2,...)}]
		[,EOMSET = { X'xx' | X'FF'}]
		[,PINTBL = labelpin]
		[,MSTRTBL = (msintrt,msoutrt)]
		[,ALITE = {SYS | 1 | 2 | 3}]
		[,CURSOR = {Y | N | R}]
		[,ERTLS = {Y | N}]
		[,UDDccc = {value | 00 }]
label	DEV4710	[OUTBHDR = labelout]
		[,PS = {n | 1}]
		[,WL = {n | 0}]
		[,OFFSET = {lines | 0}]
		[,SHARED = {Y | N}]
		[,KEY = {N | Y}]
		[,NL = {SPECIFY | AUTO}]
		[,CPI = {1 | 0}]
		[,LINE = {n | 40/48}]
		[,DEFTRT = labeltrt]
		[,DEFCHAR = lblchar]
		[,PRTY = {Y | N}]
		[,UDDPC1 = {value | 00}]

Name	Operation	Operand
label	DEV4720	MODEL = {1 | 2 | 3 | 4}
		[,PS = {n | 1}] [,WL = {n | 0}]
		[,OUTBHDR = labelout]
		[,SHARED = {Y | N}]
		[,OFFSET = {[steps | 0][,lines | 0]}]
		[,KEY = {N | Y}]
		[,NL = {SPECIFY | AUTO}]
		[,LPI = {6 | 5}]
		[,CPI = {1 | 0}] [,LINE = {n | 82/99}]
		[,FONT = {Q | R}]
		[,ADVANCE = {Y | N}]
		[,PRTY = {N | Y}]
		[,DEFTRT = labeltrt]
		[,DEFCHAR = lblchar]
		[,SKEW = {3 | 2 | 1 | 0}]
		[,DEFQUAL = labelqchar]
		[,UDDPC1 ={value | 00}]
label	DKATBL	(accent,letter-and-composite,...) [(...)]
[label]	ENDGEN	[([id],seg,[disp],data),...]
[label]	EXTCTR	{ loop (loop[,tla,comp] [,{sa1[,sa{...san] sa1-san}]])} ...,ID = X'nnnn'

Name	Operation	Operand
[label]	FILES	{DKT= {1 | 2} | DSK={1 | 2}}
		[,PF= {blks | 0}]
		[,TF= ({num | 0} , {trks | 1})]
		[,INDX1= ({1,...n | 0})]
		[,INDX2= ({1,...n | 0})]
		[,INDX3= ({1,...n | 0})]
		[,INDX4= ({1,...n | 0})]
		[,COMF= ({1,...n | 0})]
		[,INDXC= ({1,...n | 0})]
		[,BUF= {Y | N}] [,CPYAP= {Y | N}]
		[,EDAM= ([{N | datasets | Y}] [,{buffers | 2}] [,{NOALLOC | ALLOC}] [,{XRCD | NOXRCD}] [,{KEYED | NOKEYED}] [,{MB | NOMB}])
		[,LOGTM= {Y | N}]

Name	Operation	Operand
label	INTRTBL	[(code[,char][,func] [, { EOF | EOFA | EOFB | EOM | EOMA | EOMB } ,mask[,EID]])] ,... [,KYBD = { 3278A | 3278G | 3278K | 36047 | 3606 | 3608 | 4621 | 4622 | 4623 | 4627 | 4661 | 4662 | 4663 | 4664 | 4662J | 4664J | 470450 | 47US62 | 47US77 | 47US107 | 47US112 | 47KAT62 | 47KAT77 | 47KAT112 } [,CASE = {0 | 1 | 2 | 3 | 4}]] [,SCRANGE = (low,high)] [,IND = Y | N]
label	LDA	defaddr,device[,component] ,[(lda1[,lda2][,lda3])] [,A | B]
label	LOOPS	ID = n,TYPE = {L | R},BPS = nnnn
label	MSTRTBL	[([mscode],[char][,func])...,] TYPE = ({ IN | OUT } [,NULL])
[label]	OPTMOD	{name1 | func1} [,{name2 | func2}] ....

Name	Operation	Operand
label	OUTBHDR	{ CHARDEF = (maclbl1,maclbl2) OUTSPEC = (maclbl1,maclbl2,..., maclbln) OUTSPEC = (maclbl1,maclbl2,..., maclbln) ,CHARDEF = (maclbl1,maclbl2) }
label	OUTRTBL	[(pos,X'n'),...] DEVICE = { 3278 | 3604 | 3606 | 3608 | 3610 | 3611 | 3612 | 3615 | 4704 | 4704M2 } ,CHARSET = charset [,DEF = pos]
label	OUTSPEC	[(pos,X'aa'),...] ,DEVICE = {3616 | 4710 | 4720} [,{CHARSET = country | CPGID = idnumber}]
label	PINTBL	[(({ pincode | START | END | FILL | ENCRYPT } , { X'nn' | NOOP | C'c' } [,EOM])] ,...
[label]	PRIDSP	{ stid | ({stid | stid-stid} [,{stid | stid-stid}]...) } [stid ,{stid | stid-stid} [,{stid | stid-stid}]...)...] ,ID = n [,STARTUP = {ON | OFF | N}]

Name	Operation	Operand
label	PSSLST	(name1[,name2])
label	SEGINIT	[([id],seg,[disp],data),...]
[label]	SEGSTOR	CLASS = nnn,
		SEGSIZE = ([seg0],seg2,...,seg12)
		[,{S13 = size | S13ID = class}]
[label]	STARTGEN	ID = name,DATE = mmddyy
		[,VN = {n | 0}]
		[,CTLR = {4701-1 | 4701-2 | 4701-5}]
		[,S13 = {nnnn | 0}]
		[,S15 = {nnnn | 0}]
		[,MSGLITE = {n | 4}]
		[,DSKOP = {Y | N}]
		[,MONITOR = ({ STD | EXP | [,log] } / { EXPMB [,segs][,log] })]
		[,TOALERT = {N | Y}]
		[,TIMEOUT = {m:ss | 2:00}]
		[,PRINT = {NOGEN | GEN}]
		[,STATS = {NOWRAP | WRAP}]
		[,DUMP = {N | Y}]
		[,BSC = {X'usc' | C'usc'}]
		[,CNM = ({id,term} [,tranpl])]

Name	Operation	Operand
[label]	STATION	ID = nn
		[,TERM = {term | (term[lda1,..., lda8] [{A | B}]...)}]
		[,DA = (port[([lda1],[lda2])],...)
		[,MAXSTOR = n]
		[,SS = class]
		,APBNM = {name | (name[,vn | 1])}
		,DELSET = X'nn'
		[,CPU = {N | Y,PU | Y,BIND}]
		[,SHARED = {Y | N}]
		[,STARTUP = {Y | N}]
		[,WRT = {n | 1}]
		[,INSTR = {nnnn | 5000]
		[,ASYNLDA = {ldan | (lda0...lda7)}]
		[,BSC = ({X'usc' | C'usc'} [,DEF])]
		[,RETSTK = {n | 6}]
		[,INTMR = {n | 0}]
		[,DSM = (fct,r1,c1,r2,c2,EAB, DCA,SBF)]
		[,APSTACK = n] [,MAXSEG = n]
		[,DPOOL = poolname]
		[,TRANPL = poollabel]
		[,DTACC = n]
label	TRANPL	(siz1,ct1)[,(siz2,ct2)...(sizn,ctn)]
label	TRTLIST	SHIFT1 = case1, SHIFT2 = case2, SHIFT3 = case3, SHIFT4 = case4, ALTTBL = alt

Chapter 3. Coded Values for 4700 Assembler Language

Coded values for LDAs:

Mnemonic	*LDA*	*Explanation*
CT	0	3614/3624 terminal
KB	0	4704/3604/3278 keyboard
DS	1	4704/3604/3278 display
MS	2	4704/3604 magnetic stripe encoder
JP	3	3615 administrative terminal printer
JP	3	4710/3616 journal printer
PB	4	3611/3612 passbook printer
PB	4	4710/3616 document printer

Coded values for files:

Mnemonic	*Explanation*
A	Absolute address
C	Composite file
CR	Control Record
DSK	Controller diskette
DSID	Data set open with temporary file option
L	Log
P	Permanent file
PBN	Physical Block Number
PLR	Data set logical record
TF1	Temporary files
TF2	Temporary files
TF3	Temporary files
TF4	Temporary files

Coded values for the host or a logical work station:

Mnemonic	*Explanation*
CP	Central Processor (host)
ST	Station

Coded values for masks:

Mnemonic	*Hex Value*	*Explanation*
BL	08	A significant (one) bit is lost.
BU	04	The device is not available (busy).
EQ	01	The values compared are equal.
GE	05	The first operand is greater than or equal to the second operand.
GT	04	The first operand is greater than the second operand.
ID	02	The ID specified is invalid, the name was not found, or the ID was out of the range of valid IDs.
IL	02	An incorrect length is specified.
IO	04	An invalid segment 0 is specified: segment 0, operator A has been set as the default.
IS	04	An invalid segment is specified in the parameter list.
LE	03	The first operand is less than or equal to the second operand.
LT	02	The first operand is less than the second.
MD	08	A modulus error occurred.
ME	04	The tested field and mask are identical.
MO	08	All tested bits are 1's.
MX	02	The tested bits are mixed 1's and 0's.
MZ	01	All tested bits are 0 or the mask bits are all 0.
NE	06	The values compared are not equal.
NG	02	The result or data is negative.
NL	01	No significant (one) bits are lost.
NN	04	The field is not numeric.
NO	08	No segment 0 for operator B exists.
NS	04	There is an invalid device specification.
NZ	02	The result or data is nonzero (logical instructions only).
OK	01	The operation is successful.
OV	08	An overflow occurred.
PS	04	The result or data is positive.
SP	04	There is insufficient segment space.
ST	02	Status is returned.
SU	08	The segment is in use.
TR	08	Truncation occurred.
ZD	08	Division by zero was attempted.
ZO	01	The result or data is zero.

Chapter 4. Device Control Characters

Control Characters Common to All Devices

OD Carriage return
15 New line
25 Line feed
34 Position (3-byte sequence)
Byte 2:
Bit 4 = 0: Position to absolute value
Bit 4 = 1: Position relative to current position
Bit 5 = 0: Position horizontally
Bit 5 = 1: Position vertically
Bits 6 and 7: (display only)
00 Move to new position, change cursor
01 Erase, but do not change cursor
10 Erase to new position, change cursor
11 Reserved
Byte 3: Line or column number
35 Transparent write (byte 2 = length of string data)

4704/3604/3278 Keyboard Display

04 Select
0C New page
0D Return
15 New line
25 Line feed
28 Set attribute
34 Position
35 Transparent write

3606/3608 Financial Services Terminals Keyboard Display

0C Form feed
0D Return
15 New line
17 Program operator communication
25 Line feed
34 Position
35 Transparent write

3608 Printing Financial Services Terminal Printer

04 Select
0C Form feed
0D Return
15 New line
17 Program operator communication
25 Line feed
34 Position
35 Transparent write

4710 Receipt/Validation Printer and 3616 Document and Passbook Printer

0C Form feed
0D Carriage return
0E Select user-defined characters
0F Deselect user-defined characters
11 Select bold print
12 Deselect bold print
15 New line
25 Line feed
34 Position

3610 Document Printer, 3611 Passbook Printer, and 3612 Passbook and Document Printer

04 Select
0C Form feed
0D Carriage return
15 New line
25 Line feed
34 Position
35 Transparent write

3615 Administrative Terminal Printer

04 Select
0C Form feed
0D Carriage return
15 New line
17 Program operator communication

Controlling individual indicators:

3615 Indicator:	*On:*	*Off:*
1	17 C1 01	17 C2 01
2	17 C1 02	17 C2 02
3	17 C1 03	17 C3 03
4	17 C1 04	17 C2 04

Controlling all indicators simultaneously:

	Control Byte 2	3615 Indicator: 1	2	3	4	
17 C3 . . .	00					Turns all indicators off*

	Control Byte 2	3615 Indicator: 1	2	3	4	
17 C3 . . .	10				•	
	20			•		
	30			•	•	
	40		•			
	50		•		•	
	60		•	•		Turns indicators marked • on
	70		•	•	•	
	80	•				
	90	•			•	
	A0	•		•		
	B0	•		•	•	
	C0	•	•			
	D0	•	•		•	
	E0	•	•	•		
	F0	•	•	•	•	

*Unwanted indicators must first be turned off before desired indicators are turned on.

25 Line feed
34 Position
35 Transparent write

3262 Printer

Refer to the *3262 Line Printer Models 3 and 13 Component Description.*

3287 Printer

Refer to the *3287 Printer Models 1 and 2 Component Description.*

4720 Passbook and Document Printer

0C Form feed
0D Carriage return
0E Enter user-defined character set
0F Enter Standard character set
11 Bold printing
12 Regular printing
15 New line
25 Line feed
34CCAA New print position

Chapter 5. Machine Instructions

This section contains the 4700 assembler language (in operation code sequence).

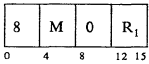
For more information on the machine instructions, refer to the *IBM 4700 Finance Communication System Controller Programming Library* manuals.

The following symbols appear in this section.

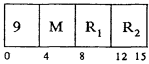
Symbol	Meaning
A(-)	Address of "-" (in adcon form)
D_n	Displacement
I_n	Immediate data
L	Length
M	Mask
N	Bit Number
P	Field pointer
R_n	Register
S_n	Segment
T	First bit in field is a wait modifier (0 for wait, 1 for no wait); remaining bits are logical device address (LDA)
W	First bit in field is a wait modifier (0 for wait, 1 for no wait); remaining bits are reserved
X	Code - see Volume 1
Y	File - Field is set as follows (see Volume 2 for explanation):

Hex Digit	*Mnemonic*
0	C
1	TF1
2	TF2
3	TF3
4	TF4
5	PLR
6	PBN
9	DSID
D	A
E	P
F	L

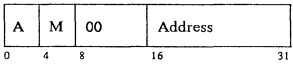
BRANR



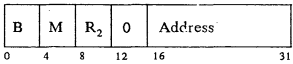
BRANLR



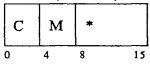
BRAN



BRANL

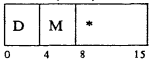


JUMP (forward)



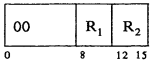
*=number of bytes being jumped

JUMP (back)

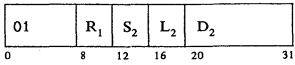


*=number of bytes being jumped

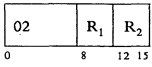
ADDREG



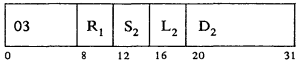
ADDFLD



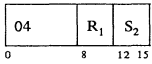
SUBREG



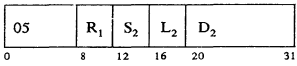
SUBFLD



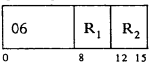
LDSEG



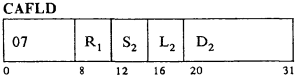
LDFLD



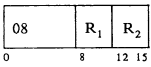
CAREG



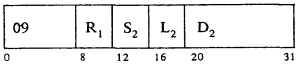
CAFLD



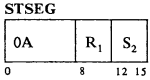
MPYREG



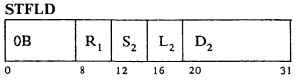
MPYFLD



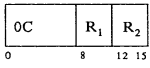
STSEG



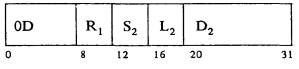
STFLD



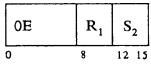
DIVREG



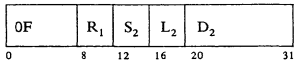
DIVFLD



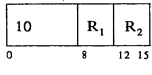
LDSEGC



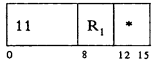
LDFLDC



LDREG,

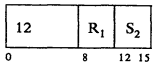


SHIFTR

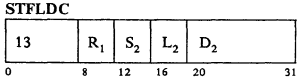


*Count value = 0-15 for shift positions 1-16

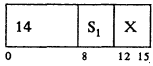
STSEGC



STFLDC

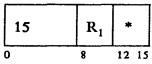


SETFPL



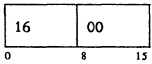
The above instruction format is generated if neither the primary field pointer (PFP) nor the field length indicator (FLI) is numeric.

SHIFTL

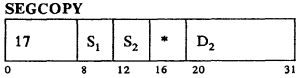


*Count value = 0-15 for shift positions 1-16

SELECT

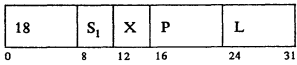


SEGCOPY

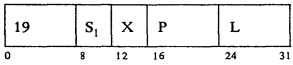


* Mask – 0 = copy to S_1; 1 = copy from S_1

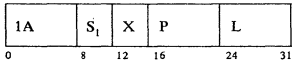
SETFPL



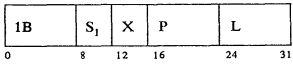
The above instruction format is generated if neither the primary field pointer (PFP) nor the field length indicator (FLI) is specified with register notation.



The above instruction format is generated if the primary field pointer (PFP) is not specified with register notation, and the field length indicator (FLI) is specified with register notation.

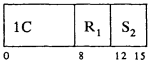


The above instruction format is generated if the primary field pointer (PFP) is specified with register notation, and the field length indicator (FLI) is not specified with register notation.

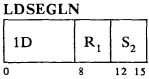


The above instruction format is generated if the primary field pointer (PFP) and the field length indicator (FLI) are specified with register notation.

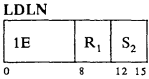
LDSFP



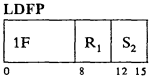
LDSEGLN



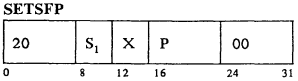
LDLN



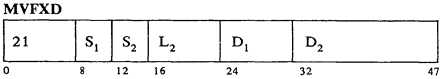
LDFP



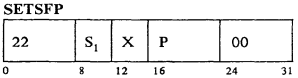
SETSFP



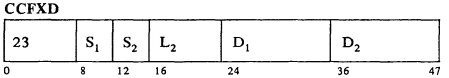
MVFXD



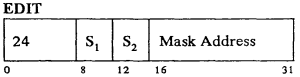
SETSFP



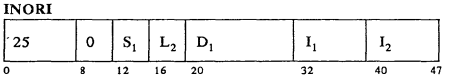
CCFXD



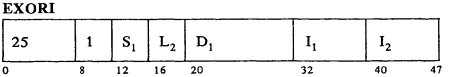
EDIT



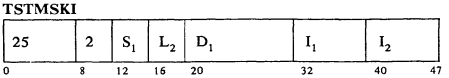
INORI



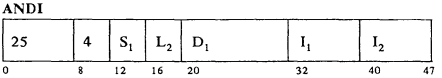
EXORI



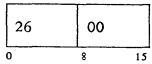
TSTMSKI



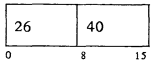
ANDI



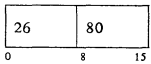
PAUSE



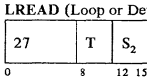
LWAIT



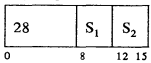
LPOST



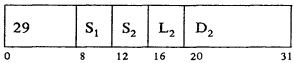
LREAD (Loop or Device)



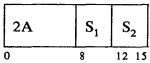
MVSEG



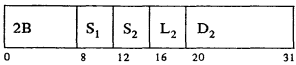
MVFLD



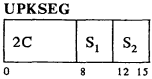
CCSEG



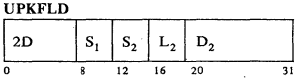
CCFLD



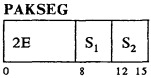
UPKSEG



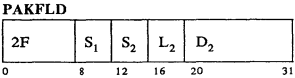
UPKFLD



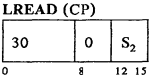
PAKSEG



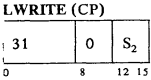
PAKFLD



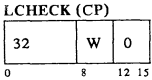
LREAD (CP)



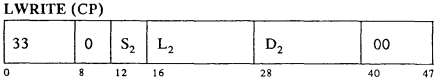
LWRITE (CP)



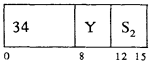
LCHECK (CP)



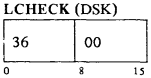
LWRITE (CP)



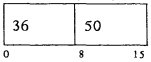
LREAD



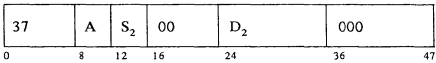
LCHECK (DSK)



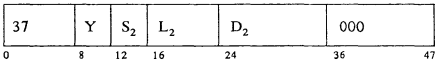
LCHECK (PLR)



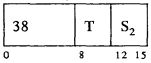
LDKT



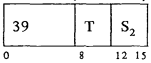
LWRITE



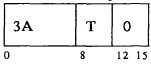
LREAD (Terminal-NOTRACK)



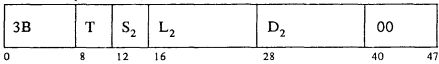
LWRITE (Terminal)



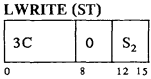
LCHECK (Loop Check-Terminal Device)



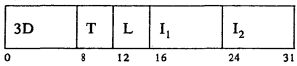
LWRITE (Terminal)



LWRITE (ST)

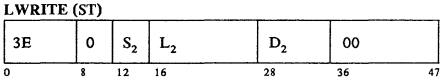


WRTI

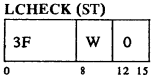


L - 1 or 2

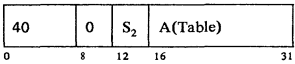
LWRITE (ST)



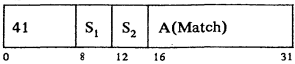
LCHECK (ST)



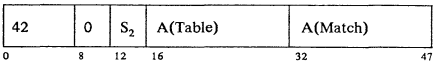
LSEEK



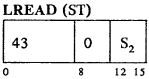
LSEEK



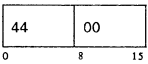
LSEEK



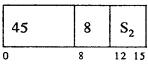
LREAD (ST)



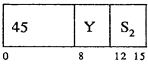
LRETURN



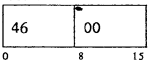
DELETE



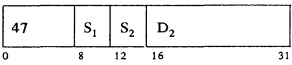
REPLACE



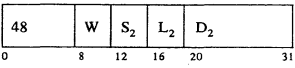
LEXIT



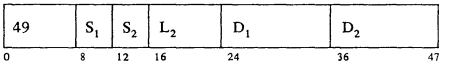
VERIFY



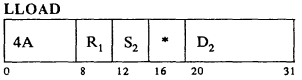
SIGNAL



AND

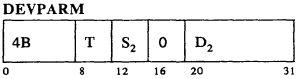


LLOAD

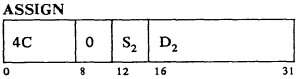


*Mask – 0 for normal LLOAD, 1 for expanded LLOAD

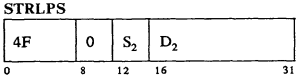
DEVPARM



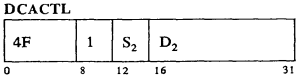
ASSIGN



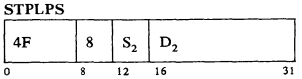
STRLPS



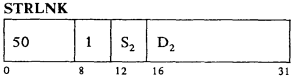
DCACTL



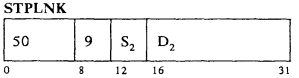
STPLPS



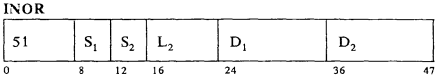
STRLNK



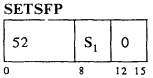
STPLNK



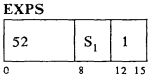
INOR



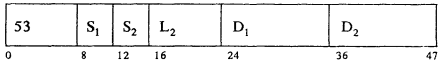
SETSFP



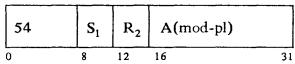
EXPS



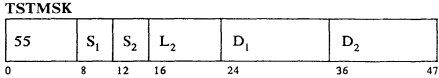
EXOR



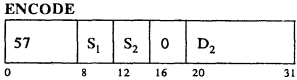
MODCHK



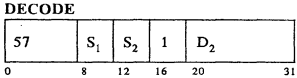
TSTMSK



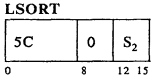
ENCODE



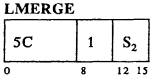
DECODE



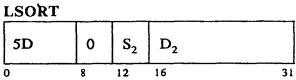
LSORT



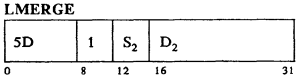
LMERGE



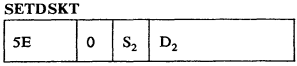
LSORT



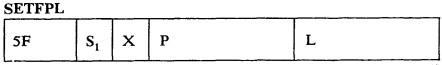
LMERGE



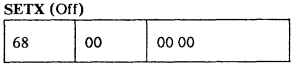
SETDSKT



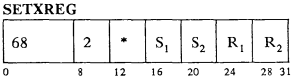
SETFPL



SETX (Off)

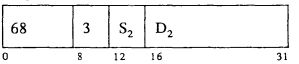


SETXREG

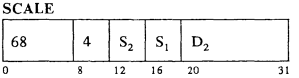


*Mask – 0 = no operands; 1 = operand 2 only; 2 = operand 1 only; 3 = both operands.

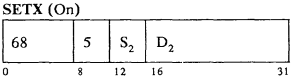
TESTX



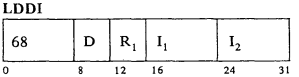
SCALE



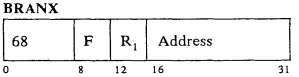
SETX (On)



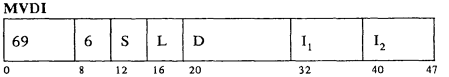
LDDI



BRANX

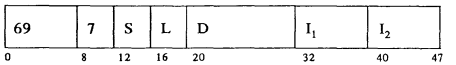


MVDI

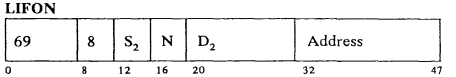


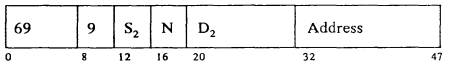
If only one byte of data is specified, X'00' is added in bits 40—47.

CCDI



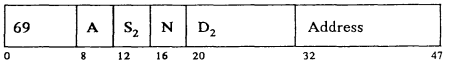
LIFON

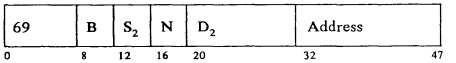




Change bit switch (N) if no branch taken.

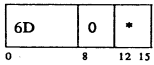
LIFOFF





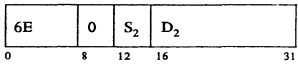
Change bit switch (N) if no branch taken.

LCHAP

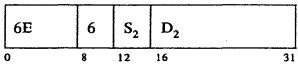


*Bits 12-13 = 0 for off, 2 for on, 1 for next; Bits 14-15 are reserved.

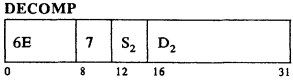
LTRT



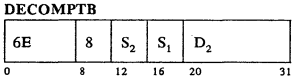
STATS



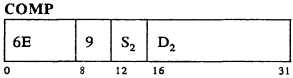
DECOMP



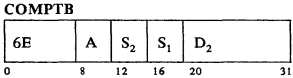
DECOMPTB



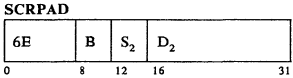
COMP



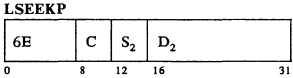
COMPTB



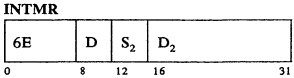
SCRPAD



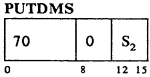
LSEEKP



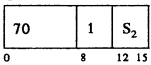
INTMR



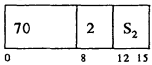
PUTDMS



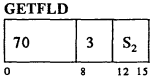
GETDMS



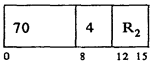
PUTFLD



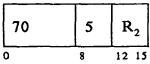
GETFLD



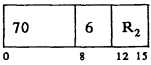
PUTDMS



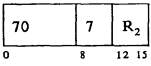
GETDMS



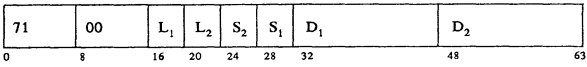
PUTFLD



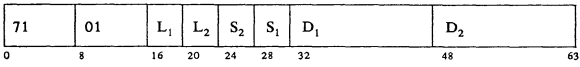
GETFLD



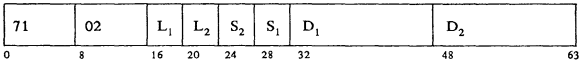
ADDZ



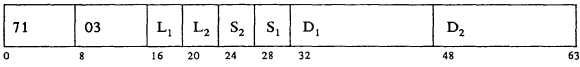
SUBZ



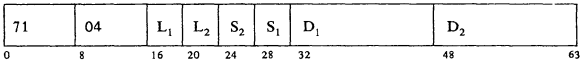
MPYZ



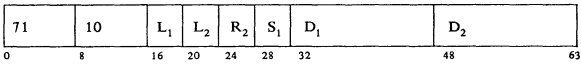
DIVZ



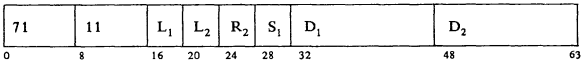
MVCZ



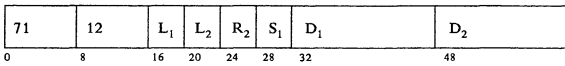
ADDZ



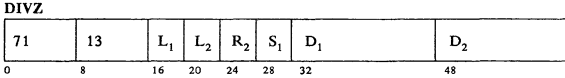
SUBZ



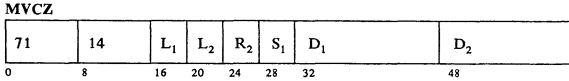
MPYZ



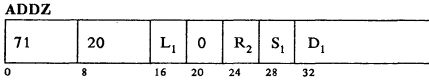
DIVZ



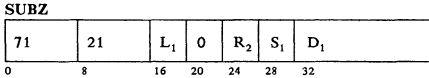
MVCZ



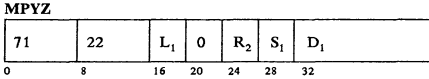
ADDZ



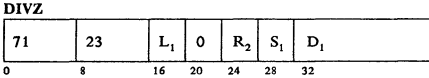
SUBZ



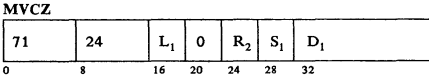
MPYZ



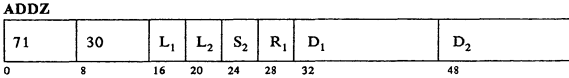
DIVZ



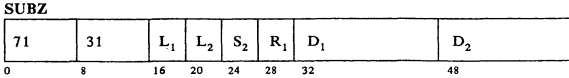
MVCZ



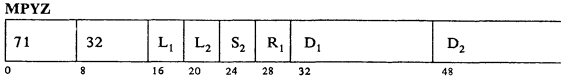
ADDZ



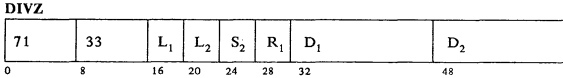
SUBZ



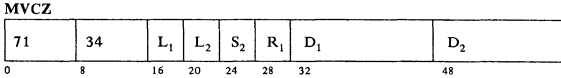
MPYZ



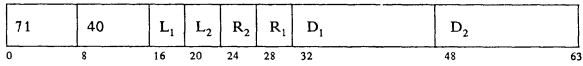
DIVZ



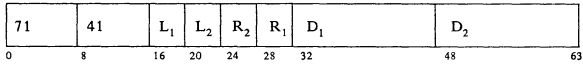
MVCZ



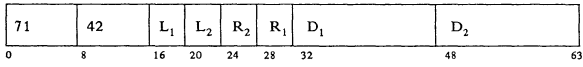
ADDZ



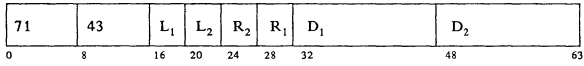
SUBZ



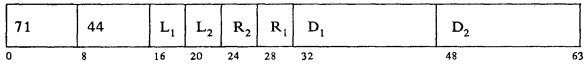
MPYZ



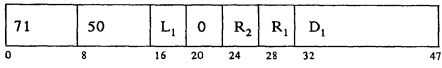
DIVZ



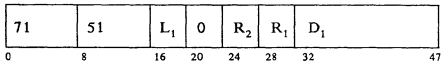
MVCZ



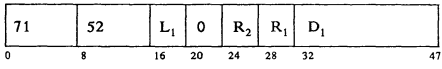
ADDZ



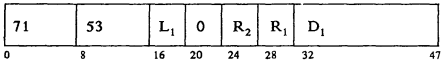
SUBZ



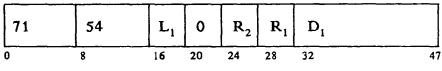
MPYZ



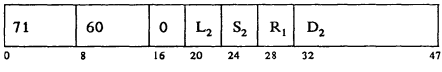
DIVZ



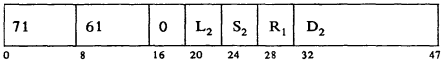
MVCZ



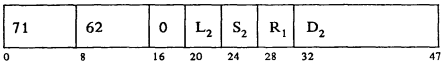
ADDZ



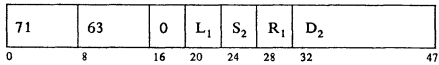
SUBZ



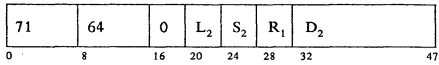
MPYZ



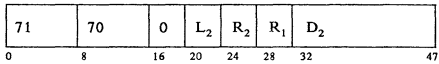
DIVZ



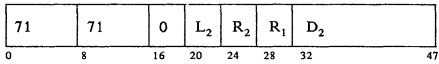
MVCZ



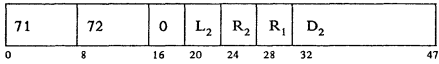
ADDZ



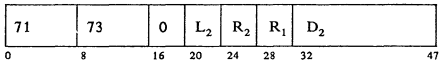
SUBZ



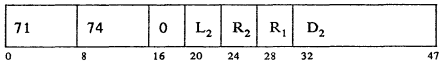
MPYZ



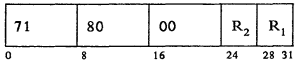
DIVZ



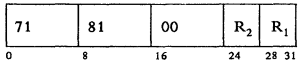
MVCZ



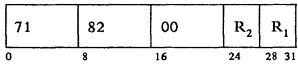
ADDZ



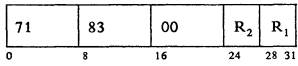
SUBZ



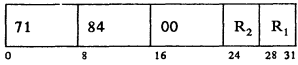
MPYZ



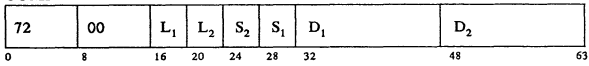
DIVZ



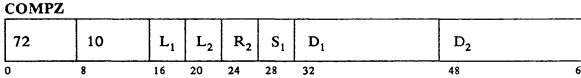
MVCZ



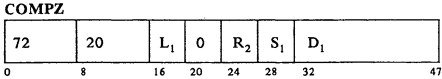
COMPZ



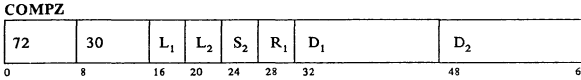
COMPZ



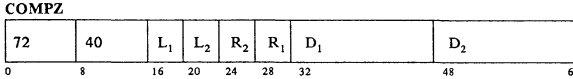
COMPZ



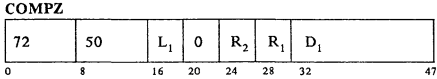
COMPZ



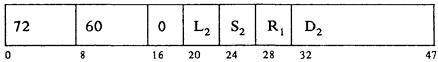
COMPZ



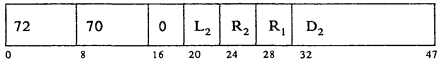
COMPZ



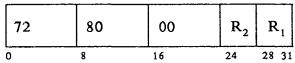
COMPZ



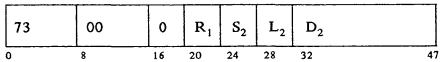
COMPZ



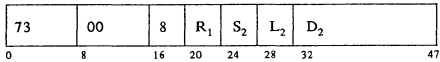
COMPZ



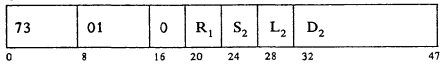
ADDFLD



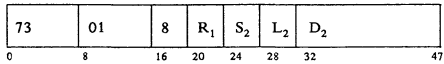
ADDFLDL



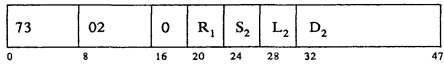
CAFLD



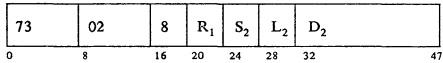
CAFLDL



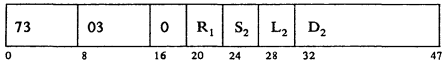
DIVFLD



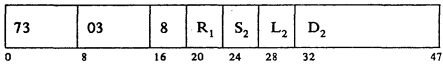
DIVFLDL



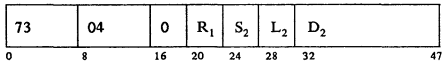
LDFLD



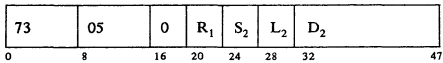
LDFLDL



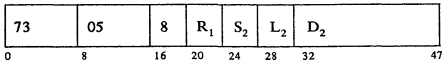
LDFLDC



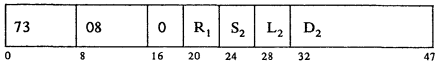
MPYFLD



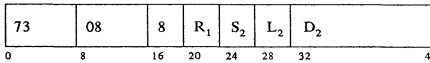
MPYFLDL



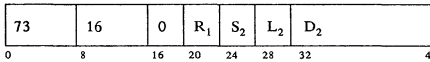
SUBFLD



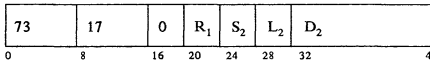
SUBFLDL



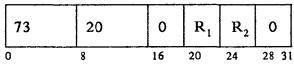
STFLD



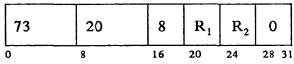
STFLDC



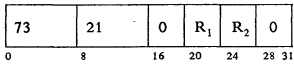
ADDFLD



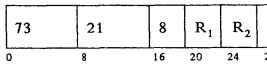
ADDFLDL



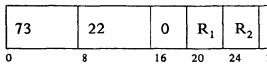
CAFLD



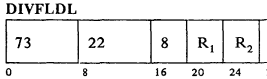
CAFLDL



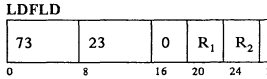
DIVFLD



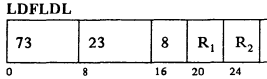
DIVFLDL



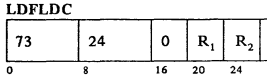
LDFLD



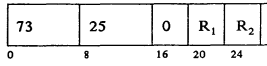
LDFLDL



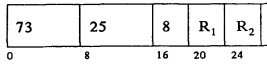
LDFLDC



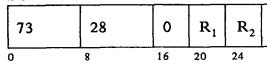
MPYFLD



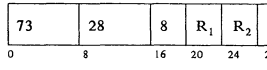
MPYFLDL



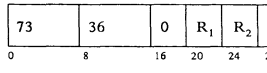
SUBFLD



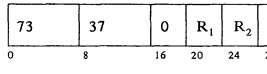
SUBFLDL



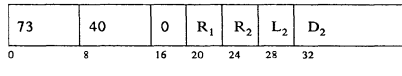
STFLD



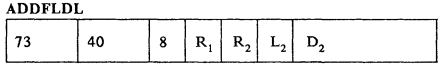
STFLDC



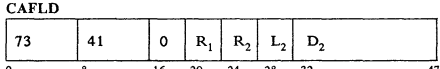
ADDFLD



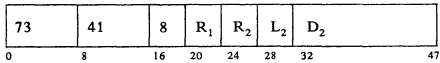
ADDFLDL



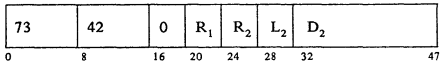
CAFLD



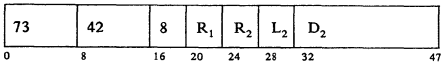
CAFLDL



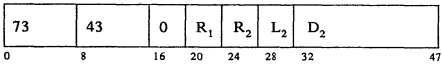
DIVFLD



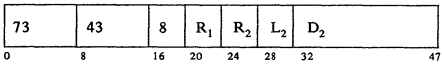
DIVFLDL



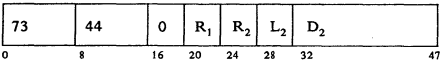
LDFLD



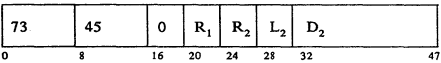
LDFLDL



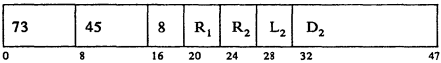
LDFLDC



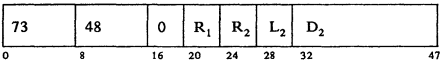
MPYFLD



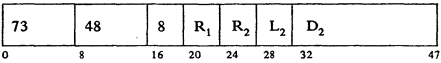
MPYFLDL



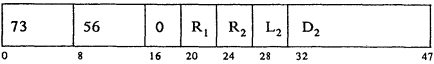
SUBFLD



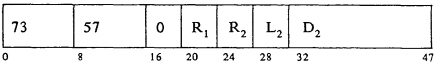
SUBFLDL



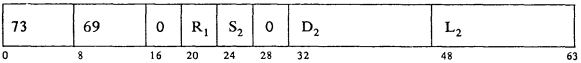
STFLD



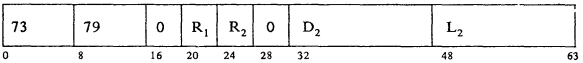
STFLDC



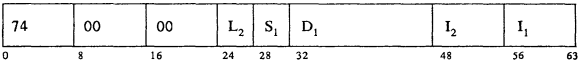
LDRA



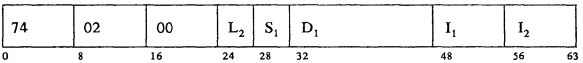
LDRA



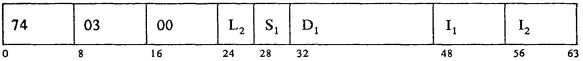
ANDI



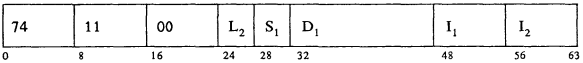
EXORI



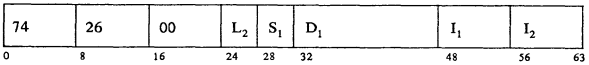
INORI



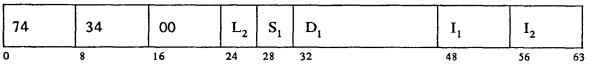
CCDI



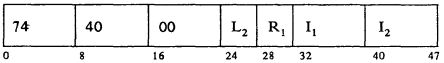
TSTMSKI



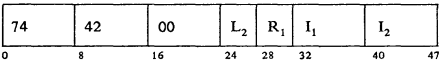
MVDI



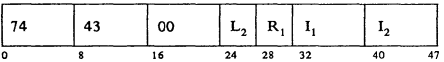
ANDI



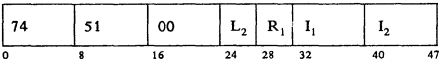
EXORI



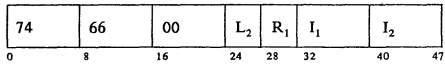
INORI



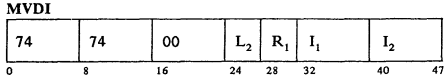
CCDI



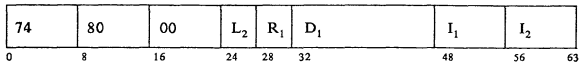
TSTMSKI



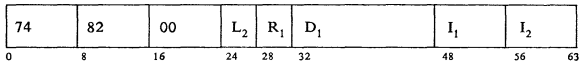
MVDI



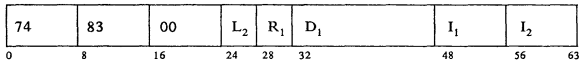
ANDI



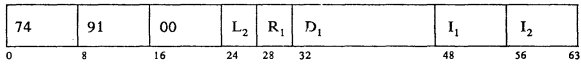
EXORI



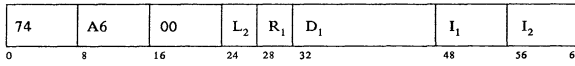
INORI



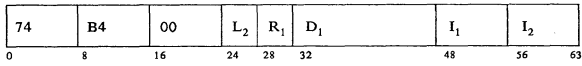
CCDI



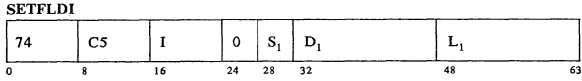
TSTMSKI



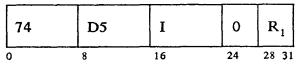
MVDI



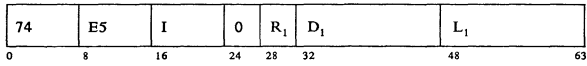
SETFLDI



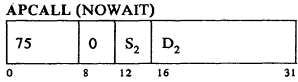
SETFLDI



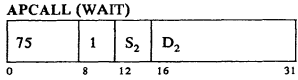
SETFLDI



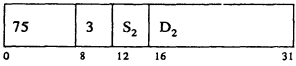
APCALL (NOWAIT)



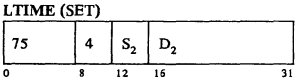
APCALL (WAIT)



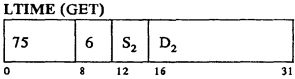
CRETN



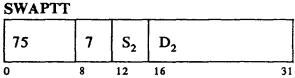
LTIME (SET)



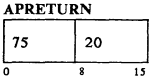
LTIME (GET)



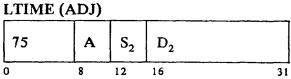
SWAPTT



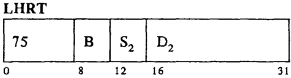
APRETURN



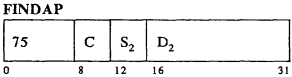
LTIME (ADJ)



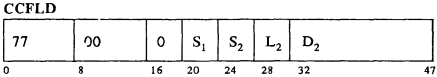
LHRT



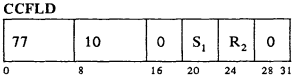
FINDAP



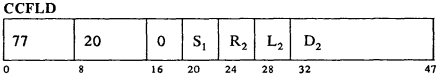
CCFLD



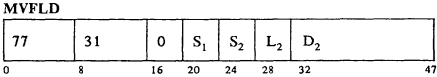
CCFLD



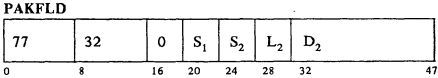
CCFLD



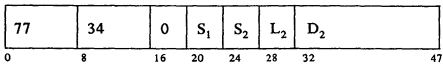
MVFLD



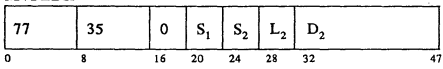
PAKFLD



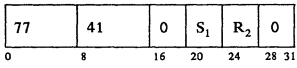
UPKFLD



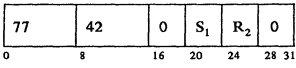
MVFLDR



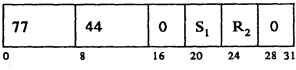
MVFLD



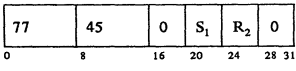
PAKFLD



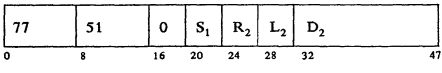
UPKFLD



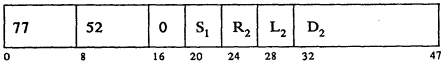
MVFLDR



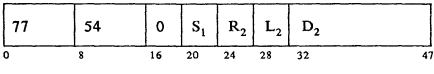
MVFLD



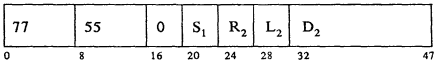
PAKFLD



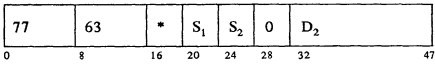
UPKFLD



MVFLDR

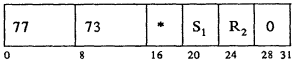


SEGCOPY



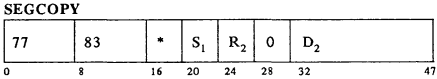
*Mask – 0 = copy to S_1; 1 = copy from S_1

SEGCOPY



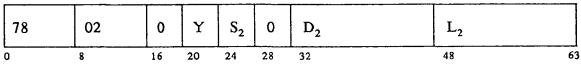
*Mask – 0 = copy to S_1; 1 = copy from S_1

SEGCOPY

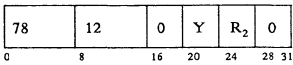


*Mask – 0 = copy to S_1; 1 = copy from S_1

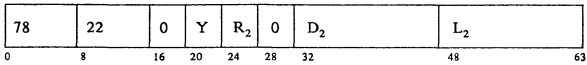
LREAD



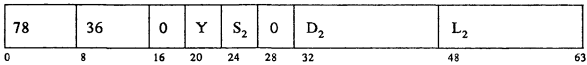
LREAD



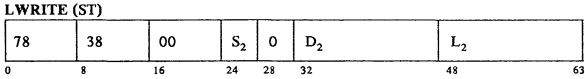
LREAD



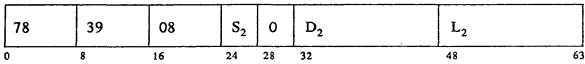
LWRITE



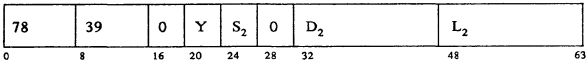
LWRITE (ST)



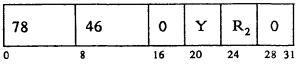
DELETE



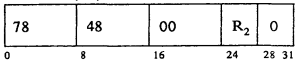
REPLACE



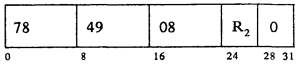
LWRITE



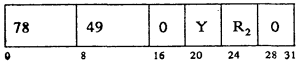
LWRITE (ST)



DELETE



REPLACE



LWRITE

78	56	0	Y	R_2	0	D_2	L_2
0	8	16	20	24	28	32	48 63

LWRITE (ST)

78	58	00	R_2	0	D_2	L_2
0	8	16	24	28	32	48 63

DELETE

78	59	08	R_2	0	D_2	L_2
0	8	16	24	28	32	48 63

REPLACE

78	59	0	Y	R_2	0	D_2	L_2
0	8	16	20	24	28	32	48 63

LREAD (CP)

78	63	00	S_2	0	D_2	L_2
0	8	16	24	28	32	48 63

LREAD

78	64	0	T	S_2	0	D_2	L_2
0	8	16	20	24	28	32	48 63

LREAD (CP)

78	73	00	R_2	0
0	8	16	24	28 31

LREAD

78	74	0	T	R_2	0
0	8	16	20	24	28 31

LREAD (CP)

78	83	00	R_2	0	D_2	L_2
0	8	16	24	28	32	48 63

LREAD

78	84	0	T	R_2	0	D_2	L_2
0	8	16	20	24	28	32	48 63

LREAD (CP)

78	95	00	S_2	0	D_2	L_2
0	8	16	24	28	32	48 63

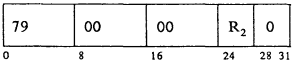
LWRITE (CP)

78	A5	00	R_2	0
0	8	16	24	28 31

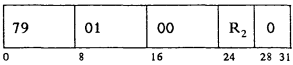
LWRITE (CP)

78	B5	00	R_2	0	D_2	L_2
0	8	16	24	28	32	48 63

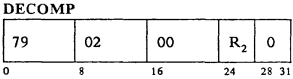
ASSIGN



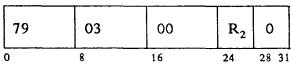
COMP



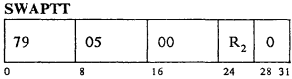
DECOMP



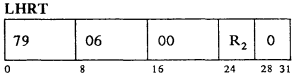
SCRPAD



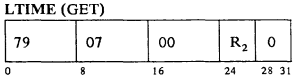
SWAPTT



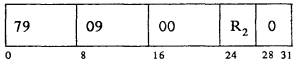
LHRT



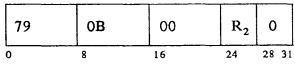
LTIME (GET)



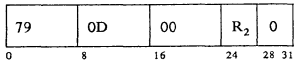
LTRT



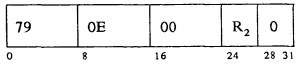
TESTX



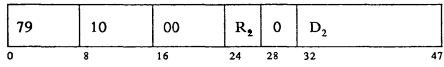
FINDAP



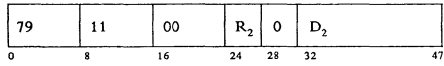
VIEW



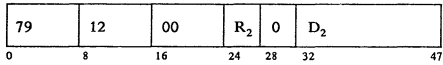
ASSIGN



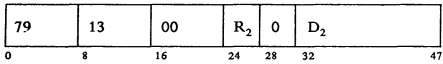
COMP



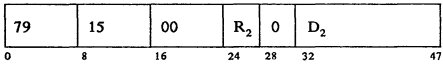
DECOMP



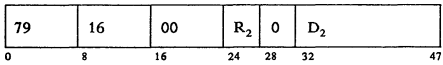
SCRPAD



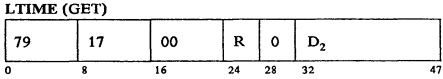
SWAPTT



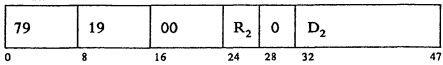
LHRT



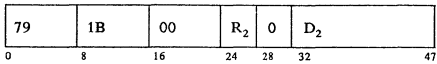
LTIME (GET)



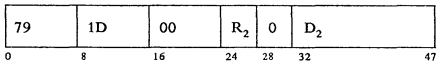
LTRT



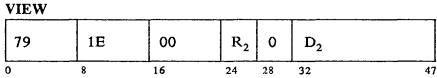
TESTX



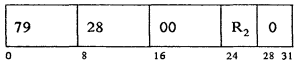
FINDAP



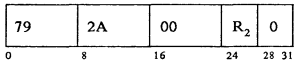
VIEW



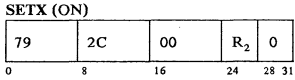
LMERGE



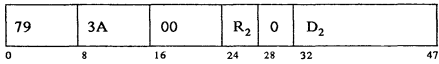
CRETN



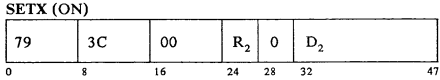
SETX (ON)



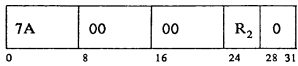
CRETN



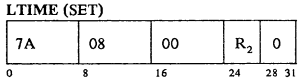
SETX (ON)



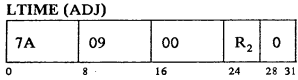
LSEEKP



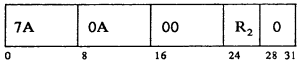
LTIME (SET)



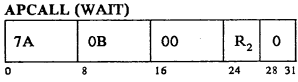
LTIME (ADJ)



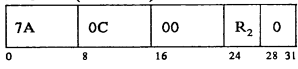
INTMR



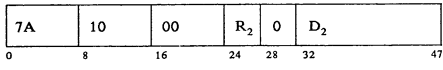
APCALL (WAIT)



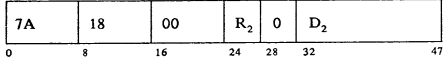
APCALL (NOWAIT)



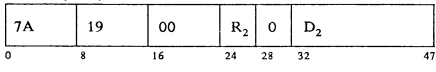
LSEEKP



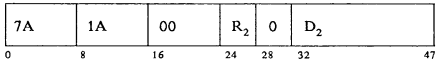
LTIME (SET)



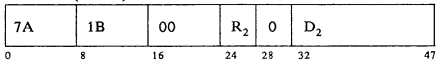
LTIME (ADJ)



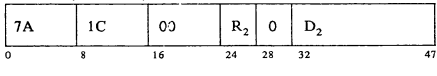
INTMR



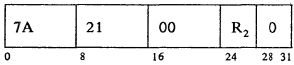
APCALL (WAIT)



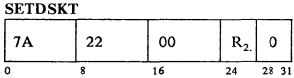
APCALL(NOWAIT)



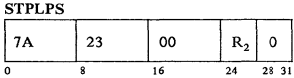
LSORT



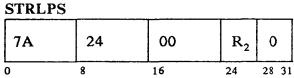
SETDSKT



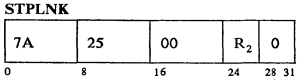
STPLPS



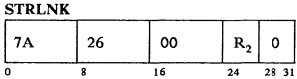
STRLPS



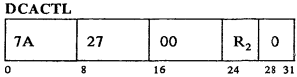
STPLNK



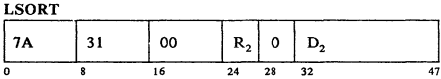
STRLNK



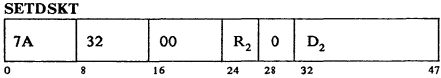
DCACTL



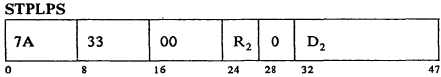
LSORT



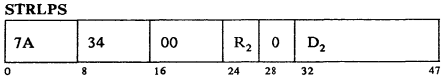
SETDSKT



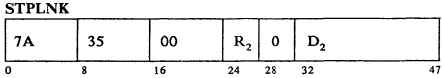
STPLPS



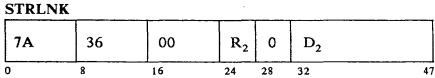
STRLPS



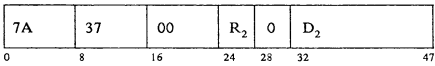
STPLNK



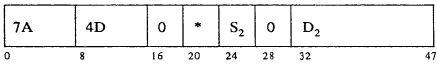
STRLNK



DCACTL

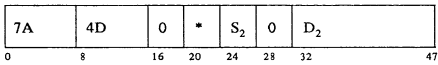


DTACCESS



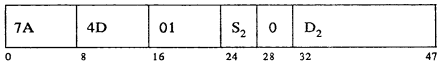
* = 0 if the LWAIT Parameter = Y
8 if the LWAIT Parameter = N

SEGALLOC

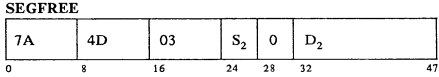


* = 2 if the LWAIT Parameter = Y
A if the LWAIT Parameter = N

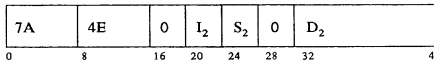
DTAFREE



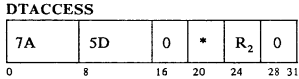
SEGFREE



DPOOL

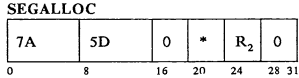


DTACCESS



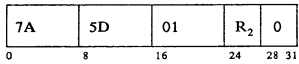
* = 0 if the LWAIT Parameter = Y
8 if the LWAIT Parameter = N

SEGALLOC

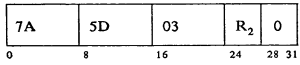


* = 2 if the LWAIT Parameter = Y
A if the LWAIT Parameter = N

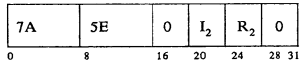
DTAFREE



SEGFREE



DPOOL



DTACCESS

7A	6D	0	*	R_2	0	D_2
0	8	16	20	24	28	32 47

* = 0 if the LWAIT Parameter = Y (default)
8 if the LWAIT Parameter = N

SEGALLOC

7A	6D	0	*	R_2	0	D_2
0	8	16	20	24	28	32 47

* = 2 if the LWAIT Parameter = Y (default)
A if the LWAIT Parameter = N

DTAFREE

7A	6D	01	R_2	0	D_2
0	8	16	24	28	32 47

SEGFREE

7A	6D	03	R_2	0	D_2
0	8	16	24	28	32 47

DPOOL

7A	6E	0	I_1	S_2	0	D_2
0	8	16	20	24	28	32 47

DEVPARM

7B	00	0	T	S_2	0	D_2
0	8	16	20	24	28	32 47

LDKT

7B	02	00	S_2	0	D_2
0	8	16	24	28	32 47

SIGNAL

7B	03	0	I_1	S_2	L_2	D_2
0	8	16	20	24	28	32 47

DEVPARM

7B	10	0	T	R_2	0
0	8	16	20	24	28 31

LDKT

7B	12	00	R_2	0
0	8	16	24	28 31

SIGNAL

7B	13	0	I_1	R_2	0
0	8	16	20	24	28 31

DEVPARM

7B	20	0	T	R_2	0	D_2
0	8	16	20	24	28	32 47

LDKT

7B	22	00	R_2	0	D_2
0	8	16	24	28	32 47

SIGNAL

7B	23	0	I_1	R_2	L_2	D_2
0	8	16	20	24	28	32 47

LLOAD

7B	34	*	R_1	S_2	0	D_2
0	8	16	20	24	28	32 47

*0 for normal LLOAD; 1 for expanded LLOAD

LLOAD

7B	44	*	R_1	R_2	0
0	8	16	20	24	28 31

*0 for normal LLOAD; 1 for expanded LLOAD

LLOAD

7B	54	*	R_1	R_2	0	D_2
0	8	16	20	24	28	32 47

*0 for normal LLOAD; 1 for expanded LLOAD

ENCIPHER

7B	65	00	S_2	0	D_2
0	8	16	24	28	32 47

FORMDKT

7B	66	00	S_2	0	D_2
0	8	16	24	28	32 47

COMPDKT

7B	67	00	S_2	0	D_2
0	8	16	24	28	32 47

DECIPHER

7B	68	00	S_2	0	D_2
0	8	16	24	28	32 47

MACGEN

7B	6A	00	S_2	0	D_2
0	8	16	24	28	32 47

RFMK

7B	6B	00	S_2	0	D_2
0	8	16	24	28	32 47

RTMK

7B	6C	00	S_2	0	D_2
0	8	16	24	28	32 47

KEYGEN

7B	6D	00	S_2	0	D_2
0	8	16	24	28	32 47

ENCIPHER

7B	75	00	R_2	0
0	8	16	24	28 31

FORMDKT

7B	76	00	R_2	0
0	8	16	24	28 31

COMPDKT

7B	77	00	R_2	0
0	8	16	24	28 31

DECIPHER

7B	78	00	R_2	0
0	8	16	24	28 31

MACGEN

7B	7A	00	R_2	0
0	8	16	24	28 31

RFMK

7B	7B	00	R_2	0
0	8	16	24	28 31

RTMK

7B	7C	00	R_2	0
0	8	16	24	28 31

KEYGEN

7B	7D	00	R_2	0
0	8	16	24	28 31

ENCIPHER

7B	85	00	R_2	0	D_2
0	8	16	24	28	32 47

FORMDKT

7B	86	00	R_2	0	D_2
0	8	16	24	28	32 47

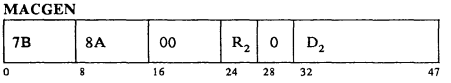
COMPDKT

7B	87	00	R_2	0	D_2
0	8	16	24	28	32 47

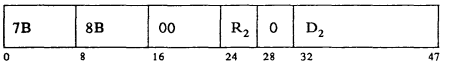
DECIPHER

7B	88	00	R_2	0	D_2
0	8	16	24	28	32 47

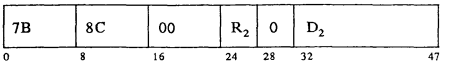
MACGEN



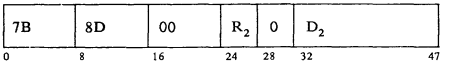
RFMK



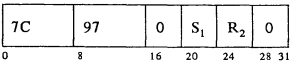
RTMK



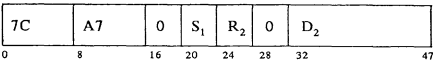
KEYGEN



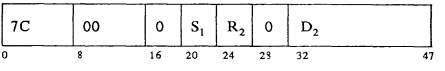
VERIFY



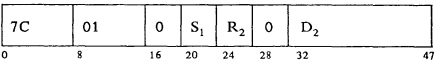
VERIFY



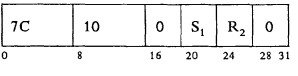
COMPTB



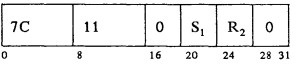
DECOMPTB



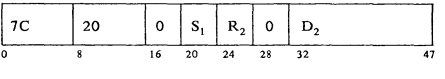
COMPTB



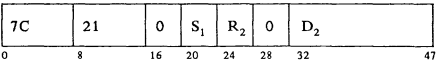
DECOMPTB



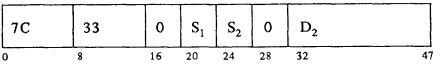
COMPTB



DECOMPTB



DECODE



ENCODE

7C	35	0	S_1	S_2	0	D_2
0	8	16	20	24	28	32 47

DECODE

7C	43	0	S_1	R_2	0
0	8	16	20	24	28 31

ENCODE

7C	45	0	S_1	R_2	0
0	8	16	20	24	28 31

DECODE

7C	53	0	S_1	R_2	0	D_2
0	8	16	20	24	28	32 47

ENCODE

7C	55	0	S_1	R_2	0	D_2
0	8	16	20	24	28	32 47

SCALE

7C	66	0	S_1	S_2	0	D_2
0	8	16	20	24	28	32 47

SCALE

7C	76	0	S_1	R_2	0
0	8	16	20	24	28 31

SCALE

7C	86	0	S_1	R_2	0	D_2
0	8	16	20	24	28	32 47

LTIMEV

7D	00	L_2	S_2	S_1	A(Table)	D_2
0	8	16	24	28	32	48 63

LTIMEV

7D	10	00	R_2	S_1	A(Table)
0	8	16	24	28	32 47

LTIMEV

7D	20	L_2	R_2	S_1	A(Table)	D_2
0	8	16	24	28	32	48 63

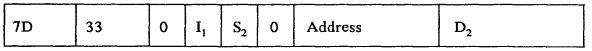
LIFOFF

7D	31	[illegible]	I_1	S_2	0	Address	D_2
0	8	16	20	24	28	32	48 63

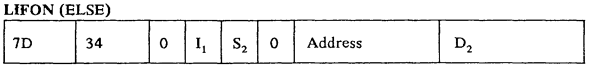
LIFOFF (ELSE)

7D	32	0	I_1	S_2	0	Address	D_2
0	8	16	20	24	28	32	48

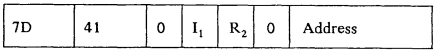
LIFON



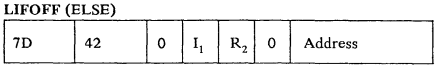
LIFON (ELSE)



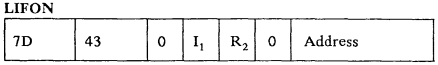
LIFOFF



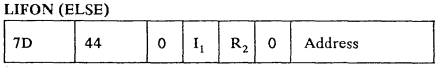
LIFOFF (ELSE)



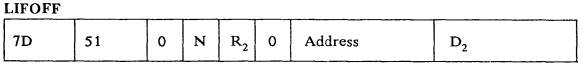
LIFON



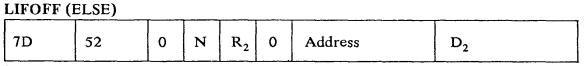
LIFON (ELSE)



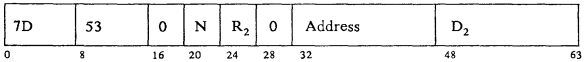
LIFOFF



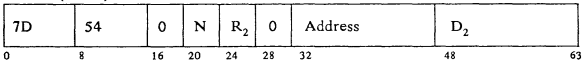
LIFOFF (ELSE)



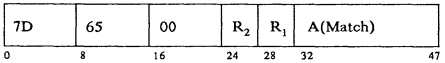
LIFON



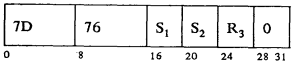
LIFON (ELSE)



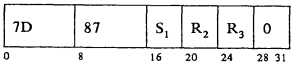
LSEEK



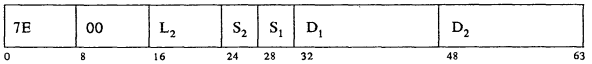
EDIT



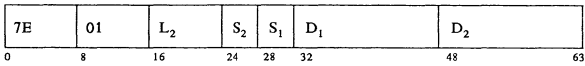
MODCHK



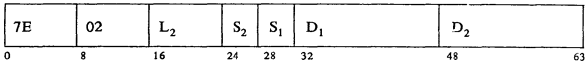
AND



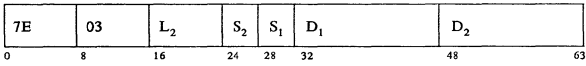
EXOR



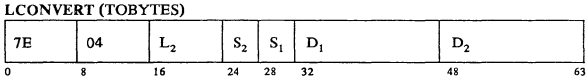
INOR



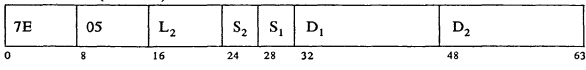
MVFXD



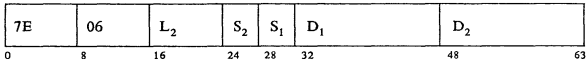
LCONVERT (TOBYTES)



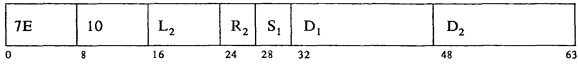
LCONVERT (TOBITS)



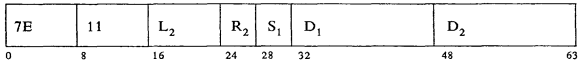
MVFXDR



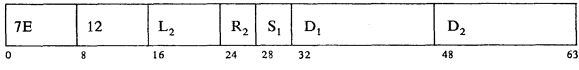
AND



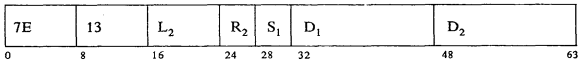
EXOR



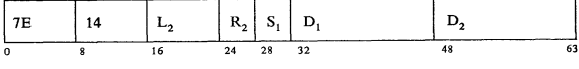
INOR



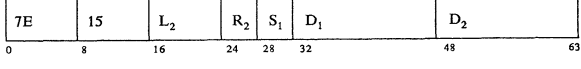
MVFXD



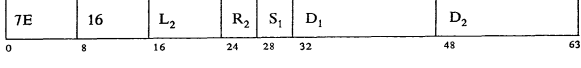
LCONVERT (TOBYTES)



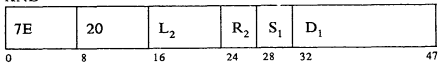
LCONVERT (TOBITS)



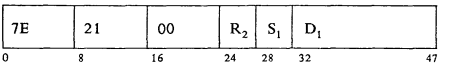
MVFXD



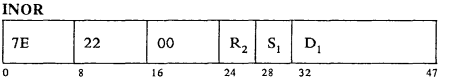
AND



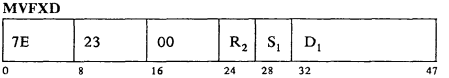
EXOR



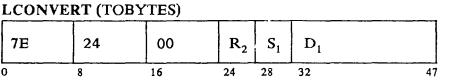
INOR



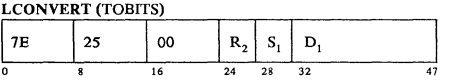
MVFXD



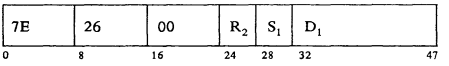
LCONVERT (TOBYTES)



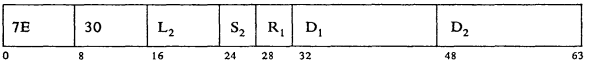
LCONVERT (TOBITS)



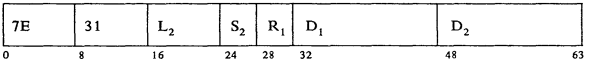
MVFXD



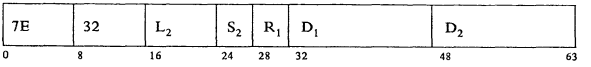
AND



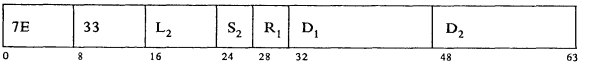
EXOR



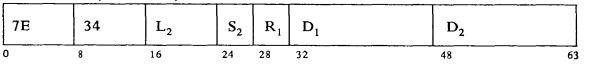
INOR



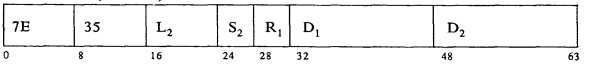
MVFXD



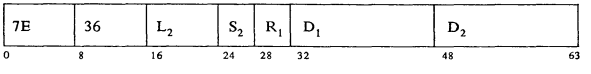
LCONVERT (TOBYTES)



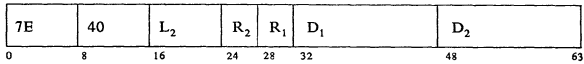
LCONVERT (TOBITS)



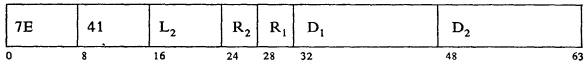
MVFXD



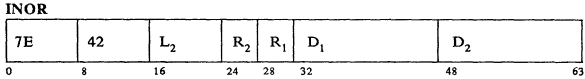
AND



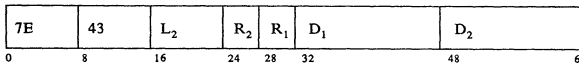
EXOR



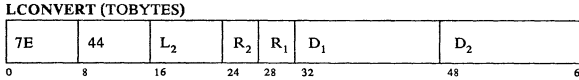
INOR



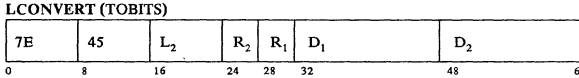
MVFXD



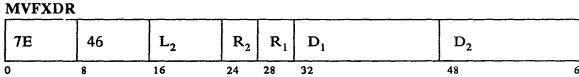
LCONVERT (TOBYTES)



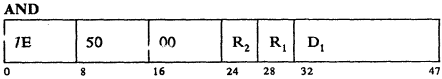
LCONVERT (TOBITS)



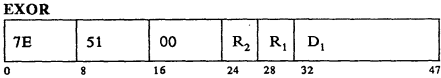
MVFXDR



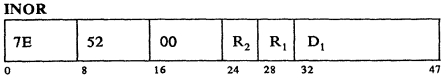
AND



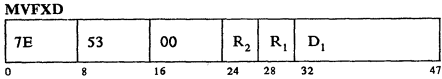
EXOR



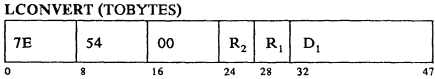
INOR



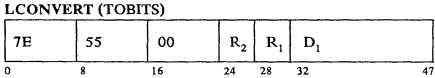
MVFXD



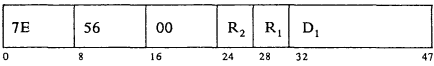
LCONVERT (TOBYTES)



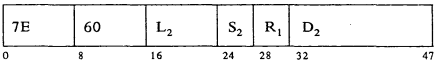
LCONVERT (TOBITS)



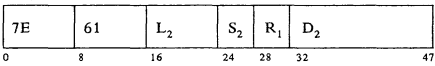
MVFXDR



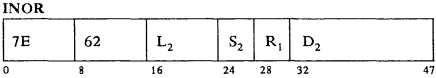
AND



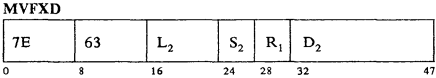
EXOR



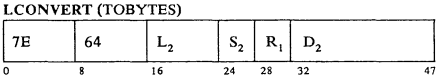
INOR



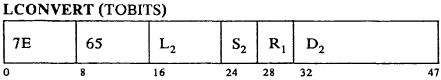
MVFXD



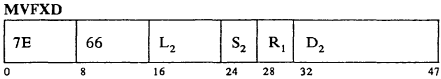
LCONVERT (TOBYTES)



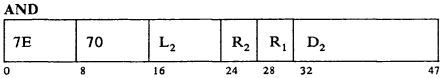
LCONVERT (TOBITS)



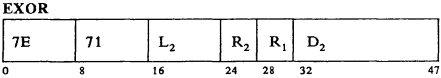
MVFXD



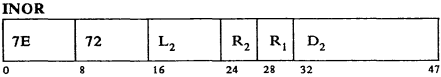
AND



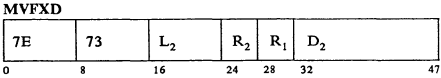
EXOR



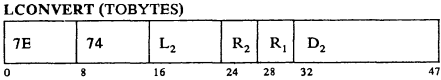
INOR



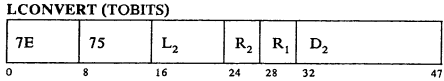
MVFXD



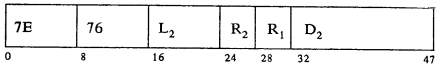
LCONVERT (TOBYTES)



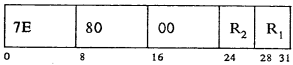
LCONVERT (TOBITS)



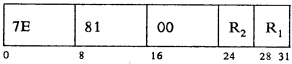
MVFXD



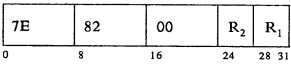
AND



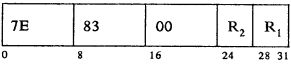
EXOR



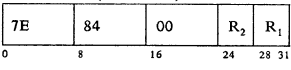
INOR



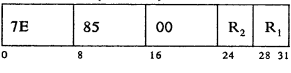
MVFXD



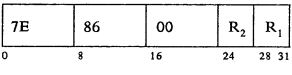
LCONVERT (TOBYTES)



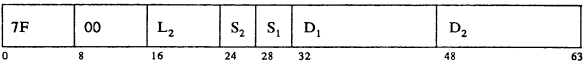
LCONVERT (TOBITS)



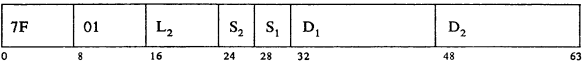
MVFXD



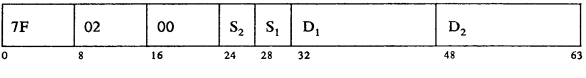
CCFXD



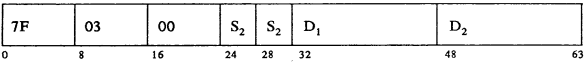
TSTMSK



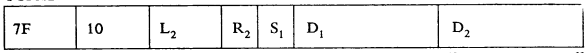
PINVERIF



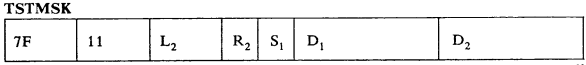
PINTRANS



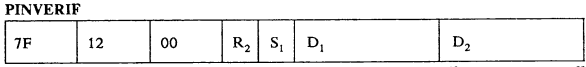
CCFXD



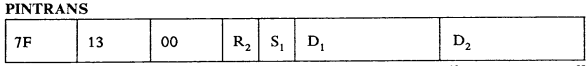
TSTMSK



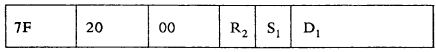
PINVERIF



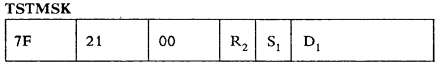
PINTRANS



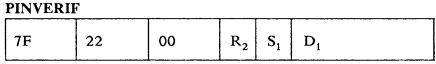
CCFXD



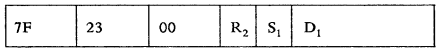
TSTMSK



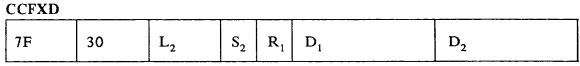
PINVERIF



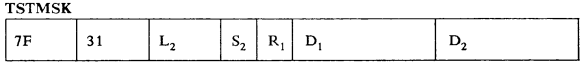
PINTRANS



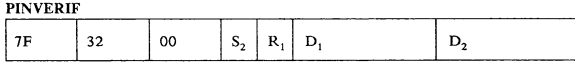
CCFXD



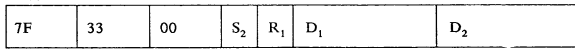
TSTMSK



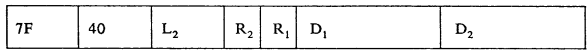
PINVERIF



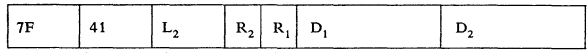
PINVERIF



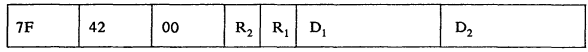
CCFXD



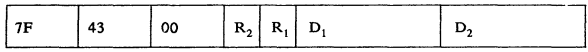
TSTMSK



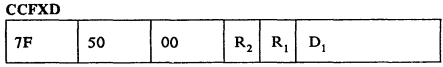
PINVERIF



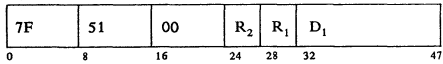
PINTRANS



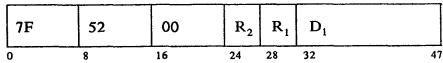
CCFXD



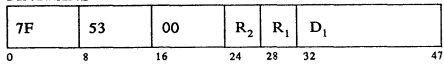
TSTMSK



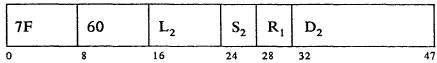
PINVERIF



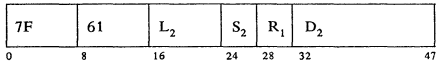
PINTRANS



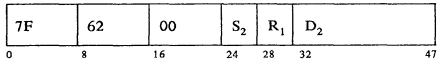
CCFXD



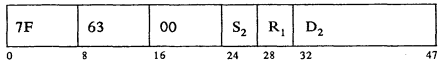
TSTMSK



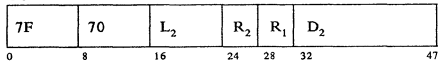
PINVERIF



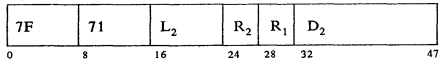
PINTRANS



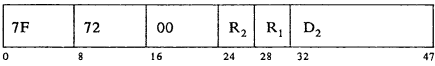
CCFXD



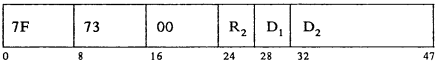
TSTMSK



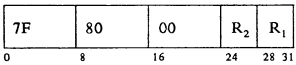
PINVERIF



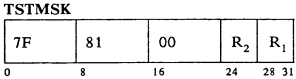
PINTRANS



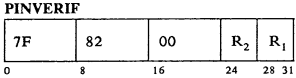
CCFXD



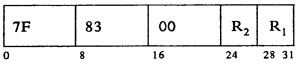
TSTMSK



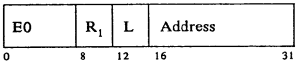
PINVERIF



PINTRANS



LEXEC



Chapter 6. System Monitor Commands

A detailed description is in the *IBM 4700 Finance Communication System: Subsystem Operating Procedures.*

000 – Log Off the System Monitor

Command	Operands
000	

There are no operands.

001 – Display Current Log Messages

Command	Operands
001	[X] [msno]

X =the message you want is displayed in hexadecimal.

msno =the 4-digit message number of the most recent message you want to see. (You need not enter leading zeros.)

002 – Display Complete Log Message

Command	Operands
002	[X] [X]msno
	0[000]

X = displays up to 504 bytes of message text in hexadecimal. If you do not enter X, the system displays up to 252 characters in EBCDIC.

msno = decimal message ID of the message to be displayed. Enter 9999 to display the most recent message.

0000 = (or only 0) to display the two diskette status bytes.

003 – Display One Entry from the Trace Area

Command	Operands
003	x y

x = 0 to display a terminal entry, or a 1 to to display a loop entry.

y = 0 to display the first entry in the area, or a 1 to display the most recent entry.

For a loop trace entry, the output format is:

t aaaa aaaa l ss cc xxxxxxxx dddddd

Where:

t	Is the entry type: S = Start of table F = Next entry in table B = Previous entry in table C = Current entry
aaaa aaaa	Is the log entry.
l	Is the loop number.
ss	Is the slot number, 1 less than the device address.
cc	Is the component number.
xxxxxxxx	Is the loop trace type.
dddddd	Is the data or command type: (RESET, LPM, REJ IN, SETIND, SETMOD, READ, WR RED, WR ECH, SENSE, ATTN, REJOUT, or END OP)

For a terminal trace entry, the command displays:

t xxxx xxxx xxxx xxxx xxxxxxxx xxxx

Where:

t	Is the type of entry: S = Start of table F = Next entry B = Previous entry C = Current entry
xx. . .xx	Is the trace data.

005 – Assign a Device as the Output Printer or Display

Command	Operands
005	[lssdd x]

l = the loop number

ss = the terminal or port address

dd = the component address (dd)

For a DCA device, enter the character A as the *l* operand. Always enter the leading 0 for the *ss* part of the operand. The values for *dd* are:

Fixed Addresses:

1	4704/3604/3278/3279 keyboard
2*	4704/3604/3278/3279 display
3	Magnetic stripe encoder
4*	3610/3612 document printer; 3262/3287/5210 printer
5	3611/3612 passbook printer
6	3606/3608 keyboard, display, magnetic stripe reader
7	3608 printer
8	3614/3624 terminal

Variable Addresses:

1-15* 4710/3615 printer (must be same value as the subaddress switch settings on the printer).

2-14* Even numbers only, 4720 and 3616 journal printer station (must be set same as subaddress switch settings).

*You can assign only these components as output printers and they must be in continuous-forms mode. Note that you cannot use odd numbers 3-15 for the 005 or 006 commands. The 3616 document printer is not a continuous-forms printer; you cannot use it for an output printer.

x = 0 for the Operator A part of device, or a 1 for the Operator B part of the device.

006 – Assign Device as Output Printer or Display

Command Operands

006 [lssdd x]

l = the loop number

ss = the terminal or port address

dd = the component address for a DCA device, enter the character A as the *l* operand. Always enter the leading 0 for the *ss* part of the operand. The values for *dd* are:

Fixed Addresses:

1 4704/3604/3278/3279 keyboard
2* 4704/3604/3278/3279 display
3 Magnetic stripe encoder
4* 3610/3612 document printer; 3262/3287/5210 printer
5 3611/3612 passbook printer
6 3606/3608 keyboard, display, magnetic stripe reader
7 3608 printer
8 3614/3624 terminal

Variable Addresses:

1-15* 4710/3615 printer (must be same value as the subaddress switch settings on the printer).

2-14* Even number only, 4720, 3616 journal printer station (must be set same as subaddress switch settings).

* You can assign only these components as output printers and they must be in continuous-forms mode. Note that you cannot use odd numbers 3-15 for the 005 or 006 commands. The 3616 document printer is not a continuous-forms printer; you cannot use it for an output printer.

x = 0 for the Operator A part of the device, or a 1 for the Operator B part of the device.

007 – Assign a Test Compenent

Command	Operands
007	[lssdd x]

lssdd = the loop number

ss = the terminal or port address

dd = the component address

For a DCA device, enter the character A as the *l* operand. Always enter the leading 0 for the *ss* part of the operand. The values for *dd* are:

Fixed Addresses:

1	4704/3604/3278/3279 keyboard
2	4704/3604/3278/3279 display
3	Magnetic stripe encoder
4	3610/3612 document printer; 3262/3287/5210 printer
5	3611/3612 passbook printer
6	3606/3608 keyboard, display, magnetic strip reader
7	3608 printer
8	3614/3624 terminal

Variable Addresses:

1-15	4710/3615 printer (must be same value as the subaddress switch settings on the printer).
2-14	Even numbers only, 4720, 3616 journal printer station (must be set same as subaddress switch settings).
3-15	Odd numbers only, 3616 document print station.

x = 0 for the Operator A part of the device, or a 1 for the Operator B part of the device.

This command displays the assignment parameter list in hexadecimal:

lsd0 xx yy zz

Where:

l Loop number, or A for DCA
s Terminal or port address
d Component address
xx C1 for A side; C2 for B side
yy Work station ID of previous owner
zz Logical device address of previous owner

008 – Assign Device Component

Command Operands

008 lssdd x yy zz

l = the loop number

ss = the terminal or port address

dd = the component address

For a DCA device, enter the character A as the *l* operand. Always enter the leading 0 for the *ss* part of the operand. The values for *dd* are:

Fixed Addresses:

1 4704/3604/3278/3279 keyboard
2 4704/3604/3278/3279 display
3 Magnetic stripe encoder
4 3610/3612 document printer; 3262/3287/ 5210 printer
5 3611/3612 passbook printer
6 3606/3608 keyboard, display, magnetic stripe reader
7 3608 printer
8 3614/3624 terminal

Variable Addresses:

1-15 4710/3615 printer (must be same value as the subaddress switch settings on the printer).
2-14 Even numbers only, 4720, 3616 journal printer station (must be set same as subaddress switch settings).
3-15 Odd numbers only; 3616 document print station ONLY FOR THE 007 AND 008 COMMAND. Do *not* use the document print station for the 005 or 006 command.

x = a 0 for the Operator A part of the device, or a 1 for the Operator B part of the device. If you omit this operand, or enter a value other than 0 or 1, the system uses the Operator A part of the device.

To change the logical unit address, omit this operand.

yy = the hexadecimal work station ID of the work station to which the device is to be assigned, or a 0 to assign the device component to the free pool.

zz = the logical device address of the device in hexadecimal.

The output format is:

lsd0 xx yy zz

Where:

l Loop number, or A for DCA
s Terminal or port address
d Component address
xx C1 for A side; C2 for B side
yy Work station ID of previous owner (0 if free pool)
zz Logical device address of previous owner

009 – Assign a Subaddress to a 3606/3608

Command	Operands
009	id

id = the subaddress of the device, in the range 0 to 15.

The output format is:

lsd0

Where:

l Is the loop number of the device.
s Is the terminal address of the device.
d Is the component address of the device.

010 – Display Statistical Counters

Command	Operands
010	lssdd

l = the loop number

ss = the terminal or port address

dd = the component address

The values for *dd* are:

Fixed Addresses;

1 4704/3604/3278/3279 keyboard
2 4704/3604/3278/3279 display
3 Magnetic stripe encoder
4 3610/3612 document printer; 3262/3287/5210 printer
5 3611/3612 passbook printer
6 3606/3608 keyboard, display, magnetic stripe reader
7 3608 printer
8 3614/3624 terminal

Variable Addresses:

1-15 4710/3615 printer (must be same value as the subaddress switch settings on the printer).
2-14 Even number only, 4720, 3616 journal printer station (must be set same as subaddress switch settings).
3-15 Odd numbers only, 3616 document print station.

For other devices, code *lssdd* as follows:

9001 Host link
9002 Primary diskette
9003 Secondary diskette
9006 Encryption facility
9007 Disk drive 1
9008 Disk drive 2
x000 Loop control; x = loop number
9100 DCA adapter
A0p0 DCA port; p = port number

The output format is:

lsdm tt ss xxx xxx xxx . . . xxx

Where:

l Loop number (1-4, A for DCA)
s Terminal or port, in hexadecimal
d Component address, in hexadecimal
m Modulus value of terminal, speed of loop, link type, or 0.

For other devices, *lsdm* is displayed as:

901x host link (x = link module loaded)

0 No link module loaded
2 SDLC-SNA link loaded
5 BSC
7 X.21
8 Multi-Use loop

9020 Primary diskette drive
9030 Secondary diskette drive
9060 Encryption facility
9070 Disk drive 1
9080 Disk drive 2
9A00 DCA adapter counters
Ap00 DCA port counters (p=port number)
x00z Loop control
x = loop number
z = loop speed:
1 = 4800 bps
2 = 2400 bps
4 = 1200 bps

tt = Component types
- 01 = Host link
- 02 = Diskette
- 03 = ALA line
- 04 = Disk
- 06 = Encryption facility
- 80 = Loop control
- 81 = 4704/3604/3278/3279 keyboard
- 82 = 4704/3604/3278/3279 display
- 83 = 3610, 3611, or 3612 printer
- 84 = 3262/3287/5210 printer
- 86 = Magnetic stripe encoder (3604 and 4704-1)
- 87 = 3614/3624 terminal
- 88 = 3606 or 3608 keyboard, display, and magnetic stripe reader.
- 89 = 3608 printer
- 8A = 3615 printer
- 92 = 3616 printer
- 95 = DCA adapter
- 9A = 4710 printer
- AB = Magnetic stripe encoder (4704-2/3)
- B0 = 4720 printer

ss = Work station ID (does not apply to host, diskette, or loop)

xxx = Counter value, in decimal

011 – Change X.21 Network Selection Sequence

Command	Operands
011	[X] selseq

selseq = up to 24 EBCDIC characters or up to 48 hexadecimal characters (24 pairs, each pair representing a character). You can use all decimal digits, the comma (,), the minus sign (-), the slash (/), the period (.), and the ending delimeter – a plus sign (+).

This command works on X.21 networks. For any other network type, the system ignores 011.

012 – Print Statistical Counters

Command	Operands
012	loops

loops = the number of loops attached to the controller.

This command displays:

lsdm tt ss xxx xxx xxx . . . xxx

Where:

l Loop number (1 - 4, A for DCA)
s Terminal or port, in hexadecimal
d Component address, in hexadecimal
m Modulus value of terminal, speed of loop, link type, or 0

For other devices, *lsdm* is displayed as:

901x host link (x = link module loaded)

- 0 No link module loaded
- 2 SDLC-SNA link loaded
- 5 BSC
- 7 X.21
- 8 Multi-use loop

9020 Primary diskette drive
9030 Secondary diskette drive
9070 Disk drive 1
9080 Disk drive 2
9A00 DCA adapter counters
Ap00 DCA port counters (p=port number)
x00z Loop control

x = loop number
z = loop speed:

1 = 4800 bps
2 = 2400 bps
4 = 1200 bps

tt = Component types

- 01 = Host link
- 02 = Diskette
- 03 = ALA line
- 04 = Disk
- 06 = Encryption facility
- 80 = Loop control
- 81 = 4704/3604/3278/3279 keyboard
- 82 = 4704/3604/3278/3279 display
- 83 = 3610, 3611, or 3612 printer
- 84 = 3262/3287/5210 printer
- 86 = Magnetic stripe encoder (3604 and 4704-1)
- 87 = 3614/3624 terminal
- 88 = 3606 or 3608 keyboard, display, and magnetic stripe reader
- 89 = 3608 printer
- 8A = 3615 printer
- 92 = 3616 printer
- 95 = DCA adapter
- 9A = 4710 printer
- AB = Magnetic stripe encoder (4704-2/3)
- B0 = 4720 printer

ss = Work station ID (does not apply to host, diskette, or loop)

xxx = Counter value, in decimal

013 – Display Machine Feature Switches

Command Operands

013

There are no operands.

The 013 command display the switch settings as:

```
     xxxx xxxx xxxx xxxx xxxx
     |                      |
Bit  0                     19
```

Each x represents a bit in the machine switch setting.

0	Reserved
1	Reserved
2	(On) 2 diskette adapters
3	(On) DCA adapter
4	Reserved
5-7	Number of loops
	001 - 1
	010 - 2
	011 - 3
	100 - 4
8	Reserved
9-11	Host link type
	000 No host link
	001 X.21
	010 SDLC-SNA
	011 Multi-use loop
	101 Binary synchronous
12-13	Number of disk drives
	00 No disk drives
	10 1 disk drive
	11 2 disk drives
14-19	Reserved

014 – CNM – Control Work Station Timers

Command Operands

014 func [wkst] [tmr]

func = the value indicating the function you want to perform.

Value	Function
2	Read
3	Read and reset
4	Activate
5	Deactivate

wkst = the ID of the work station associated with the timers. If you omit this operand, or specify 00, all work stations are included.

tmr = the value representing the timers you want included. If you enter 0, all timers for the work station are included. If you omitted the work station ID, DO NOT enter this operand.

This command displays data in this format:

f ss tt eeeeeee iii mmmmm xxxxxx vvvvv

Where:

f	Status of timer 0 = Active, but not running 4 = Acitve and running 8 = Not running; deactivated C = Deactivated while running
ss	Work station ID
tt	Timer number
eeeeeee	Total time of all intervals since last reset
iii	Intervals timed since last reset
mmmmm	Shortest interval since last reset
xxxxxx	Longest interval since last reset
vvvvv	Average interval since last reset

Note: If the system displays only part of your data, press Enter to display the rest. (Host operators, issue the system ATTN command.)

015 – Display and Set Time-Of–Day Clock

Command	Operands
015	func yyyy/mm/dd hh:mm:ss yy

func = the value indicating the function you want performed.

Value	Function
0	Set clock value.
1	Display time-of-day.
2	Stop clock from 1 to 99 seconds; use yy to indicate 1 to 99.
3	Advance time of day from 1 to 99 seconds; use yy to indicate 1 to 99.
4	Display high-resolution timer; the rightmost 2 digits are a decimal fraction.

yyyy/mm/dd hh:mm:ss = enter a string to set the clock to a new value. In the string, use these values:

yyyy	Year (for example, 1982)
mm	Month
dd	Day of the month
hh	Hour of the day
mm	Minutes
ss	Seconds
/ or :	You can enter any delimiters; they are retained by the system for subsequent time and date displays. Each occurrence can be a different value. (See the sample command at the end of this section.)

yy = a number in the range 1 to 99 indicating the number of seconds to advance (func=3) or delay (func=2) the clock.

You can issue the 045 command to request your own user translation table for the complete alphameric character set.

016 – Measure System Use

Command	Operands
016	func

func = a 0 to start measuring usage, or a 1 to stop measuring and display the measurement. You can log off the system monitor after you start the measurement; the measurement continues until you stop it by entering 016 1.

020 – Ripple Test

Command	Operands
020	rpt lin aaa bbb

rpt = the decimal number of times to print or display the test line. If you enter 000, the test continues until you press Reset twice. If a remote operator enters 000, the test runs 20 times.

lin = the decimal number of characters to present on each line. You can specify up to 255 characters per line.

aaa = the output position of the first of a sequence of characters to be used in the test. This operand is either the physical position on the print wheel or print belt, or the logical position in the display output translation table. See the OUTRTBL macro in the *4700 Controller Programming Library* for information describing output positions.

bbb = the output position of the last character in the sequence to be used in the test.

021 – Print or Display Character Group On Test Component

Command	Operands
021	rpt aaa bbb

rpt = the decimal number of times to print or display the test line. If you enter 000, the test continues until you press Reset twice. If a remote operator enters 000, the test runs 20 times.

aaa = the output position of the first of a group of consecutive characters to be used in the test. This operand is either the physical position on the print wheel or print belt, or the logical position in the display output translation table. See the OUTRTBL macro in the *4700 Controller Programming Library* for information describing output positions.

bbb = the output position of the last character in the group to be used in the test.

023 – Present Test Pattern

Command	Operands
023	rpt lin aaa bbb

rpt = the decimal number of times to print or display the test line. If you enter 000, the test continues until you press Reset twice. If a remote operator enters 000, the test runs 20 times.

lin = the decimal number of characters to present on each line. You can specify a line length up to 255 characters.

aaa = the output position of the first of two characters to be used in the test. This operand is either the physical position on the print wheel or print belt, or the logical position in the display output translation table. See the OUTRTBL macro in the *4700 Controller Programming Library* for information describing output positions.

bbb = the output position of the second character to be used in the test. This character can be the same as the first character.

024 – Read from Test Component

Command	Operands
024	rpt x

rpt = the number of times to run the test. To run the test continuously, enter 000 (or 0); the test stops when you press Reset twice. For remote operators, the test runs 20 times.

x = the type of output you want.

0	Decimal output; asynchronous device
1	Hexadecimal output; asynchronous device
2	Decimal output; synchronous device
3	Hexadecimal output; synchronous device

025 – Print or Display Text On Test Component

Command	Operands
025	rpt text

rpt = the decimal number of times to print or display the line of text. If you enter 000, the test continues until you press Reset twice. If a remote operator enters 000, the test runs 20 times.

text = a string of decimal or hexadecimal characters to be presented at the test component.

Use the character X to precede and follow an even number of hexadecimal characters; use two Xs (XX) to present the character X.

You can intermix decimal and hexadecimal data in the same text, but be sure to enclose any hexadecimal data in Xs.

028 – Track Text On Display Component

Command	Operands
028	text

text = the message to be displayed.

029 – Send a Message to a Station

Command	Operands
029	sid text

sid = the work station ID, in decimal, of the receiving work station.

text = the message text you want to send.

Before you use 029, issue an 045 command to request your user translation table for the complete alphameric character set.

030 – Write a Controller Log Message to Diskette

Command	Operands
030	text

text = the message that you want written to the log.

Before you issue this command, issue the 045 command to request your user-assigned translation table for the translation of the complete alphameric character set.

The 030 command writes your message to the log in the form:

10 hhmm 030 text

031 – Read and Display Primary Diskette Record

Command	Operands
031	[X] ttrr [s]

X = the diskette address in hexadecimal; omit the X to enter the address in decimal.

ttrr = the track (tt) and record (rr) numbers in either decimal or hexadecimal.

s = a 0 to read from side 0, or a 1 to read from side 1. If you omit this operand, the system uses side 0.

If you specify the track and record numbers in hexadecimal, the system ignores your side specification if you specified side 1 in the *rr* field.

This command displays the first 64 bytes of the record in hexadecimal. Press the Enter key to display the rest of the record, or the rest of the records on this track. (Host operator, use the system ATTN command.) Or, enter a decimal number in the range 1-9 to skip 1 to 9 records.

032 – Change a Diskette Record On Primary Drive

Command	Operands
032	[X] ttrr [X] dsp data

X = the operand in hexadecimal; omit the X to enter the operand in decimal.

ttrr = the track (tt) and record (rr) numbers in either decimal or hexadecimal.

dsp = the displacement into the record. You can begin the change up to 255 bytes into the record.

To enter the displacement in hexadecimal, enter Xdd; the character X followed by the hexadecimal displacement, up to XFF.

data = up to 8 hexadecimal bytes (16 input characters) of data to replace the data in the record. If you enter an odd number of characters, the system sets the leftmost 4 bits to 0000.

033 – Print Dump or Storage Map from Primary Diskette Drive

Command	Operands
033	[X] [address]

X = the address in hexadecimal; omit the X to enter the address in decimal.

address = the starting address in decimal or hexadecimal. The command prints the stand-alone dump, beginning at the address you specify.

If you omit this address, the command prints the storage map.

034 – Perform Seek Test On Primary Diskette Drive

Command	Operands
034	[X] ttrr [X] ttrr [s]

X = the address in hexadecimal; omit the X to enter the address in decimal.

ttrr = the track (tt) and record (rr) numbers in either decimal or hexadecimal.

s = a 0 to read from side 0, or a 1 to read from side 1. If you omit this operand, the system uses side 0.

If you specify the track and record numbers in hexadecimal, the system ignores your side specification if you specified side 1 in the *rr* field.

The system reads first from the first address, then from the second address, alternating until you press Reset twice (host operators: the test is performed 20 times).

035 – Display Dump or Storage Map from Primary Diskette

Command	Operands
035	[X] address

X = the address in hexadecimal; omit the X to enter the address in decimal.

address = the starting address in decimal or hexadecimal. The command displays the stand-alone dump, beginning at the address you specify.

If you omit this address, the command displays the storage map.

036 – Identify Dump

Command	Operands
036	[text]

text = up to 32 characters you want to associate with a dump. If you omit this operand, the ID area for the dump is set to zeros.

037 – Compress Diskette On Primary Diskette

Command	Operands
037	[dsname func alin size]

dsname = the name of a data set to be altered. Omit this operand, and all other operands, to compress the entire diskette.

func = the value that indicates what you want done to the data set named in the first operand. Do not specify this operand without naming a specific data set. The allowable function codes are:

Code	Function
0	No change.
1	Add to the data set.
2	Subtract from the data set.
3	Delete the entire data set.
4	Extend the end of the data set to the end of the extent +1.
5	Reduce the data set by moving the end of extent to end of data +1.

alin = a 0 if you do not want to specify alignment, or a 1 if you want the data set aligned on a track boundary. If you specify a 1, the data set must already begin and end on track boundaries before you issue the command.

The system ignores requests to compress sequential data sets.

size = the number of sectors to be added or subtracted when using function codes 1 and 2. You can add or subtract from 0 to 65535 sectors.

038 – Print Operating Information from Primary Drive

Command Operands

038

There are no operands.

039 – Format a Diskette On Primary Drive

Command Operands

039 [lgth type volid crc seq]

lgth = the code that indicates the record length:

Code	Record Length
0	128 bytes per record
1	256 bytes per record
F	Use existing diskette record length

type = the code that indicates the diskette type.

Code	Diskette Type
1	Diskette 1
2	Diskette 2
3	Diskette 2D
F	Assume the type already inserted in the drive (default value)

volid = a one- to six-character volume identification. Use this operand ONLY if you want to change the existing volume ID. If you omit this operand, the system retains the existing volume ID.

Notes:

1. You can change only the volume ID by omitting the preceding operands. However, you can do this *ONLY* if the volume ID is longer than one character. If the volume ID is only one character, enter all preceding operands.
2. The following two operands, *crc* and *seq*, are optional operands. If you enter them, you must first enter a volume ID.

crc = the code that indicates the type of record to be written to the diskette.

0 Format with delete control records. This is the default.
1 Format with data records.

Specify 1 (data records) for multi-block I/O on the diskette.

seq = to modify the physical sequence numbering of records on a type 2D diskette. You can modify the sequence numbering to get more diskette accesses per revolution of the diskette. Replace *seq* with a number in the range 0 to 13.

040 – Start and Stop Loops

Command	Operands
040	func [08 00 loop]

func = a 0 to start loops, or a 1 to stop loops during error recovery.

08 00 = enter these operands as shown.

loop = the number of the loop.

041 – Start or Stop Host Link

Command	Operands
041	x [yy zz cc Xff aaaa]

X = a 0 to start the host link if it is stopped, or to perform a wrap test, or a 1 to stop the link. (If you enter 041 0, but the link is already running, the system ignores the entire request.)

yy = a combination of these codes to describe the host link.

Code	Link Characteristic
01	NRZI encoding
02	Not NRZI encoding
04	Nonwrappable modem
08	Wrappable modem
10	High-speed line
20	Low-speed line
40	Wrap-test the adapter or modem once (ignoring other parameters)

Note: The remaining operands are link parameters.

zz = a combination of these codes to characterize the link.

Code	Link Characteristic
01	Control request to send
02	Permanent request to send
04	Data set ready
08	Connect data set to line
10	Nonswitched line
20	Switched line (02 is ignored)
40	Extended command indicator
80	An entry for the control unit address follows

cc = the control unit address (only used if the CUA switches are all off).

Xff = enter the X, followed by two hexadecimal characters describing the link:

Code	Link Characteristic
01	BSC3 single message mode (used during diskette creation).
02	BSC3 batch message mode
04	X.21 direct call
08	X.21 auto-call
10	X.21 auto-answer

If you do not select any X.21 function above, the system uses the functions specified in a previous COMLINK or STRLINK.

40	Perform an X.21 wrap test. Set all other bits in this operand to 0's.
80	The *aaaaa* operand contains a node ID.

aaaaa = the five-character, hexadecimal transmission ID (XID) assigned to this controller.

042 – Start or Stop Primary Diskette

Command	Operands
042	func

func = a 0 to start the diskette drive, or a 1 to stop the diskette drive.

043 – Change Test Component Operating Parameters

Command	Operands
043	ff xx xx xx . . . xx

ff = one or two hexadecimal characters to be used as the flag byte. See the flag byte information, listed for each device, later in the description of this command.

xx = one or two hexadecimal characters representing up to nine parameter bytes. See the parameter byte information, listed for each device, later in the description of this command.

Device Operating Parameters

Following, listed by device, are the flag byte and data byte settings allowable for each device.

4710 Journal Printer

Flag byte: X'02' = continuous-forms mode (restricts concurrent sharing)
X'82' = continuous-forms mode (specifies concurrent sharing)
Data byte 1: Page size in lines
Data byte 2: Warning line number
Data byte 3: Line length
Data byte 4: Device characteristics,
- bits 0-6 unused
- bit 7 0 = 10 CPI
  1 = 12 CPI

4710 Document Printer

Flag byte: X'01' Cut-forms mode
Data byte 1: Page size in lines
Data byte 2: Warning line number
Data byte 3: Reserved
Data byte 4: Line offset in number of lines
Data byte 5: Line length
Data byte 6: Device characteristics:
- bit 0 unused
- bit 1 unused
- bit 2 0 = nonshared
  1 = shared
- bit 3 0 = start key required
  1 = autostart
- bit 4 unused
- bit 5 0 = no auto-new line
  1 = auto-new line
- bit 6 unused
- bit 7 0 = 10 CPI
  1 = 12 CPI

* Both bits 2 and 3 must not be 1

4720 Printer – Passbook Mode

Flag byte: X'00' Passbook mode
Data byte 1: Page size in lines
Data byte 2: Centerfold begin
Data byte 3: Centerfold Skip
Data byte 4: Vertical increment offset
Data byte 5: Vertical line offset
Data byte 6: Horizontal character offset
Data byte 7: Line length
Data byte 8: Device characteristics:

bit 0	0 = horizontal fold passbook
	1 = vertical fold passbook
bit 1	unused
bit 2	0 = nonshared
	1 = shared
bit 3	0 = start key required
	1 = autostart
bit 4	unused
bit 5	0 = auto-new line
	1 = line length exceeded check
bit 6	0 = 5 LPI
	1 = 6 LPI
bit 7	0 = 10 CPI
	1 = 12 CPI

Data byte 9: Extended device characteristics:

bit 0-5	reserved
bit 6-7	allowable skew:
	00 = 1.37 mm per 100 mm line
	01 = 0.68 mm per 100 mm line
	10 = 1.37 mm per 100 mm line
	11 = 1.37 mm per 100 mm line

4720 Printer – Cut-forms Mode

Flag byte: X'01' Cut-forms mode
Data byte 1: Page size in lines
Data byte 2: Warning line number
Data byte 3: Vertical increment offset
Data byte 4: Vertical line offset
Data byte 5: Line length
Data byte 6: Device characteristics:

bit 0	unused
bit 1	unused
bit 2	0 = nonshared
	1 = shared
bit 3	0 = start key required
	1 = autostart
bit 4	0 =no extended characteristics
	1 = extended characteristics (byte 7)
bit 5	0 = line length exceeded check
	1 = auto-new line
bit 6	0 = 5 LPI
	1 = 6 LPI
bit 7	0 = 10 CPI
	1 = 12 CPI

Data byte 7: Extended device characteristics:

bit 0	0 = regular font
	1 = quality font
bit 1	0 = no advance
	1 = advance journal
bit 2-5	unused
bit 6-7	allowable skew:
	00 = 1.37 mm per 100 mm line
	01 = 2.05 mm per 100 mm line
	10 = 2.74 mm per 100 mm line
	11 = 3.42 mm per 100 mm line

4720 Printer – Journal Mode

Flag byte: X'02' Journal mode
Data byte 1: Page size in lines
Data byte 2: Warning line number
Data byte 3: Line length
Data byte 4: Device characteristics:

bits 0-5	unused
bit 6	0 = 5 LPI
	1 = 6 LPI
bit 7	0 = 10 CPI
	1 = 12 CPI

4704/3604/3278/3279 Keyboard

Flag byte: The bits of this byte set the end-of-message (EOM) selection mask.
Data bytes: Not used

The bits of the flag byte specify which EOM characters defined in the translation table are to be active. If this byte is X'00', the current EOM selection mask is not changed.

4704/3604/3278/3279 Display

Flag byte: X'00'
Data byte 1: Model number, or: X'08' for 3278-2, 3279-2; X'11' or X'12' for 4704-1; X'21' or X'22' for 4704-2 and 4704-3
Data byte 2: Number of characters per lines
Data byte 3: Number of lines on screen
Data byte 4: Line number of cursor
Data byte 5: Column number of cursor

Note: Data bytes 2-5 are for information return only; you cannot change them.

3608 Printer

Flag byte: X'00' = no changes
X'80' = first line is 10-pitch
X'C0' = first line is 7-pitch
Data byte 1: Maximum form width (in tenths of an inch)
Data byte 2: First offset (in tenths of an inch)
Data byte 3: Page spacing (in tenths of an inch)

3610 and 3612 Document Printer

Flag byte:	X'01' = cut-form mode
	X'02' = continuous form, no concurrent sharing
	X'41' = cut-form mode automatic start
	X'82' = continuous form, concurrent sharing
Data byte 1:	Page size (number of lines)
Data byte 2:	Warning line (line number)
Data bytes 3 to 5:	Not used

3611 and 3612 Passbook Printer

The page size plus center-fold skip plus line offset cannot exceed 42. The number of steps and the number of lines offset are for passbook registration.

Flag byte:	X'00'
Data byte 1:	Page size (number of lines)
Data byte 2:	Center-fold start (line number)
Data byte 3:	Center-fold skip (number of lines)
Data byte 4:	Step offset (number of stepper motor steps)
Data byte 5:	Line offset (number of lines)

3615 Printer:

Setting the Parameters:

Flag byte:	X'01' = cut-forms feed option
	X'02' = continuous-forms mode
	X'41' = cut-form mode, automatic start
	X'82' = continuous form, concurrent sharing
Data byte 1:	Page size (number of lines)
Data byte 2:	Warning line (line number)

Retrieving the Parameters:

Flag byte:	X'16' = continuous form, no concurrent sharing
	X'17' = cut-form mode, no automatic start
	X'57' = cut-form mode, automatic start
	X'96' = continuous form, concurrent sharing
Data byte 1:	Page size (number of lines)
Data byte 2:	Warning line (line number)
Data byte 3:	Reserved
Data byte 4:	Reserved
Data byte 5:	Reserved
Data byte 6:	'02' = unidirectional
	'03' = bidirectional
Data byte 7:	'02' = 10 CPI
	'03' = 12 CPI

3616 Journal Printer

Flag byte: X'02' = continuous-forms mode (restricts concurrent sharing)
X'82' = continuous-forms mode (specifies concurrent sharing)
Data byte 1: Page size in lines
Data byte 2: Warning line number
Data byte 3: Line length
Data byte 4: Device characteristics,

bits 0-1	unused
bit 2*	0 = nonshared
	1 = shared
bit 3*	0 = start key required
	1 = autostart
bits 4-6	unused
bit 7	0 = 10 CPI
	1 = 12 CPI

* Bits 2 and 3 should not both be 1.

Data byte 5: Reserved
Data byte 6: Reserved
Data byte 7: Reserved
Data byte 8: Reserved

3616 Document Printer (Passbook Mode)

Flag byte: X'00' = Passbook mode
Data byte 1: Page size lines
Data byte 2: Center-fold starting line number
Data byte 3: Center-fold skip in number of lines
Data byte 4: Step offset in number steps
Data byte 5: Line offset in number of lines (see note)
Data byte 6: Left margin column number
Data byte 7: Line length

Note: Line offsets are required to space from the clamped position of a passbook to the first line.

Data byte 8: Device characteristics:

bit 0	0 = horizontal-fold passbook
	1 = vertical-fold passbook
bit 1	unused
bit 2*	0 = nonshared
	1 = shared
bit 3*	0 = start key required
	1 = autostart
bit 4	unused
bit 5	0 = no automatic new line
	1 = automatic new line
bit 6	0 = 5 LPI
	1 = 6 LPI
bit 7	0 = 10 CPI
	1 = 12 CPI

*Bits 2 and 3 should not both be 1.

3616 Document Printer (Cut-form mode)

Flag byte: X'01' Cut-form mode
Data byte 1: Page size in lines
Data byte 2: Warning line number
Data byte 3: Step offset in number of steps
Data byte 4: Line offset in number of lines
Data byte 5: Line length
Data byte 6: Device characteristics:

bit 0	unused
bit 1	0 = no data chaining 1 = data chaining
bit 2*	0 = nonshared 1 = shared
bit 3*	0 = start key required 1 = autostart
bit 4	unused
bit 5	0 = no automatic new line 1 = automatic new line
bit 6	0 = 5 LPI 1 = 6 LPI
bit 7	0 = 10 CPI 1 = 12 CPI

* Bits 2 and 3 should not both be 1.

Data byte 7: Reserved
Data byte 8: Reserved

3262/3287/5210 Printer

Flag byte: X'02' Continuous-form mode
Data byte 1: Device Characteristics

X'0x' Only segment (no chaining)
X'2x' Last segment of chain
X'4x' First segment of chain
X'6x' Middle segment of chain
X'x1' FM Header data stream for structured fields follows

044 – Write to Magnetic Stripe Encoder

Command	Operands
044	text

text = up to 36 characters consisting of the numbers 0-9 and the letters C, D, and E. The characters C and D have special meaning to the system: C indicates the end of an inquiry; D is the field separator. E is unassigned; you can use it for your needs.

045 – Change Keyboard Translation Table

Command	Operands
045	table

table = a 0 to use the IBM-supplied universal translation table, or a 1 to use the application-program translation table.

046 – Print Controller Log Messages

Command Operands

046 [first last]

first = the message number of the first log message you want printed. (Message 1 is the first message.) If you omit the operands, the system prints all messages.

last = the message number of the last message you want printed. If you omit the operands, the system prints all messages.

To print all messages in the log, enter 046 with no operands.

047 – Display Storage and Diskette Change Counters

Command Operands

047

There are no operands.

This command displays the counters in this format:

stor dskt

where *stor* is the value of the storage counter, and *dskt* is the value of the diskette counter.

048 – Change Control Operator Display Screen Attribute

Command Operands

048 mod

mod = one of these screen attribute codes:

Code	Screen Size
01-07	3604-1 thru 3604-7
08	3278-2 or 3279-2
11	480-character 4704-1
12	1920-character 4704-1
21	480-character 4704-2
22	1920-character 4704-2

049 – Change Number of Lines Displayed

Command Operands

049 lines

lines = the new number of lines to be displayed at your terminal. The valid range is 1 to 24.

051 – Wrap Test 3614/3624 with Prepared Text

Command	Operands
051	rpt x

rpt = the number of times to perform the test; the number of times to transmit and receive the line of prepared text. If you enter 000, the test continues until you press the Reset key twice (remote operators: the test runs 20 times).

x = a 0 to stop on comparison errors, or a 1 to continue the test when comparison errors occur.

052 – Wrap Test 3614/3624 with User Text

Command	Operands
052	rpt x text

rpt = the number of times to perform the test; the number of times to transmit and receive the text. If you enter 000, the test continues until you press the Reset key twice (remote operators: the test runs 20 times).

x = a 0 to stop on comparison errors, or a 1 to continue the test when comparison errors occur.

text = up to 50 hexadecimal characters (50 pairs) to be used in the test. If you enter an odd number of characters, the command sets the end of the output to X'00'.

053 – Display 3614 or 3624 Error Log

Command	Operands
053	

There are no operands.

055 – Display Test

Command	Operands
055	mod code

mod = the model number of the display to be tested (1-6).

code = a 0 to run the test on the assigned test device, or a 1 to run the test on your control-operator terminal. (Be sure that the command works on your device.) If you omit this operand, the test is run at the test device.

060 – Bypass Checking On Write to Test Component

Command	Operands
060	func

func = 0 to check all writes to the test component, or a 1 to bypass the checking.

061 – Print Keyboard and Display Messages

Command	Operands
061	func

func = a 0 to stop printing, or a 1 to print all display and keyboard messages on the output printer.

Input (keyboard) data is preceded by two asterisks (**) on the printer. If only the ** is printed, only the Enter key was pressed, with no translatable data.

062 – Ignore Error Conditions At Test Component

Command	Operands
062	func error

func = a 0 to stop on the condition noted in operand 2, or a 1 to continue in spite of the error.

error = a combination of the following codes. The first operand specifies whether the system will stop for, or ignore, the conditions specified here.

Code	Condition
80	Intervention required
40	Unit exception
10	Prior operation
02	Unit check
FF	All of the above

063 – Set Diskette Parameters

Command	Operands
063	func val

func = one, or a combination, of the following function codes.

Code	Function
01	Reset the temporary file and change the session ID number.
02	Enter a start-mode value on the next automatic startup only. The second operand contains a start-mode value.
04	Enter new start-mode value on all automatic startups. The second operand contains a start-mode value.
08	Dump on system failure.
10	Reload the system after a system failure; do not dump.
20	Set primary diskette ready.
40	Set primary diskette not ready; later diskette requests get 8001 status.
80	Load the system, as if you pressed controller's Reset switch.

val = the start-mode values only if you enter 02 or 04 on the first operand.

Code	Function
0	Delete current start-mode values; assume default start-mode value of 2 (warm start).
1	Cold start.
2	Use warm start, the default when no other value specified.
4	Prompt for system variables. The system prompts you for startup values: control unit address, dump option, control operator ID, switched-network identifier (XID).
5	System prompts for optional modules. Specify the optional module IDs you will need to execute your commands and application programs.
8	Perform cold start; do not activate host link.
9	Perform warm start; do not activate host link.

This command requires optional module P5E.

065 – Display Stations That Have Opened a Data Set

Command	Operands
065	d did

d = a 1 for the primary diskette drive, or a 2 for the secondary diskette drive. For a disk data set, enter an A for disk drive 1, or a B for disk drive 2.

did = the data set identification (in decimal) of the data set in question. If you do not know the data set ID, ask your programming personnel.

The output format is:

did xx xx xx xx xx xx xx xx

The *did* is the data set ID you specified, each *xx* is the station ID of a station that has the data set open.

066 – Clean Printer Print Wheels

Command	Operands
066	

There are no operands.

067 – Reset Extended Statistical Counters

Command	Operands
067	ctid

ctid = the ID of the extended statistical counter to be reset, or enter the value FFFF to reset all extended statistical counters.

068 – Reset Standard Statistical Counters

Command	Operands
068	lssdd

lssdd = the address of the component associated with the counter that you want to reset, or with FFFF to reset all counters.

For loop or DCA devices, replace *lssdd* with the device address, consisting of the loop number (l), the terminal or port address (ss), and the component address (dd). Always enter the leading 0 for the *ss* part of the operand. The values for *dd* are:

Fixed Addresses:

1	4704/3604/3278/3279 keyboard
2	4704/3604/3278/3279 display
3	Magnetic stripe encoder
4	3610/3612 document printer; 3262/3287/ 5210 printer
5	3611/3612 passbook printer
6	3606/3608 keyboard, display, magnetic stripe reader
7	3608 printer
8	3614 or 3624 terminal

Variable Addresses;

1-15 4710/3615 printer (must be same value as the subaddress switch settings on the printer).
2-14 Even number only, 4720, 3616 journal printer station (must be set same as subaddress switch settings).
3-15 Odd numbers only, for 3616 document print station.

For other devices, code *lssdd* as follows:

9001 Host link
9002 Diskette
9003 Diskette
9006 Encryption facility
9007 Disk drive 1
9008 Disk drive 2
x000 Loop control; x = loop number
9100 DCA adapter
A0p0 DCA port; p = port number

This command displays:

lsdm tt ss xxx xxx xxx . . . xxx

Where:

l Loop number (1-4, A for DCA)

s Terminal or port, in hexadecimal

d Component address, in hexadecimal

m Modulus value of terminal, speed of loop, link type, or 0.

For other devices, *lsdm* is displayed as:

901x host link (x = link module loaded)

0 no link module loaded
2 SDLC SDLC-SNA link loaded
5 BSC
7 X.21 version A (feature 5656)
8 X.21 version B (feature 5656)

9020 Primary diskette drive
9030 Secondary diskette drive
9060 Encryption facility
9A00 DCA adapter counters
Ap00 DCA port counters (p=port number)
x00z Loop control

x = loop number
z = loop speed:
1 = 4800 bps
2 = 2400 bps
4 = 1200 bps

tt = Component types
- 01 = Host link
- 02 = Diskette
- 03 = ALA line
- 04 = Disk
- 06 = Encryption
- 80 = Loop control
- 81 = 4704/3604/3278/3279 keyboard
- 82 = 4704/3604/3278/3279 display
- 83 = 3610, 3611, or 3612 printer
- 84 = 3262/3287/5210 printer
- 86 = Magnetic stripe encoder (3604 and 4704-1)
- 87 = 3614 or 3624 terminal
- 88 = 3606 or 3608 keyboard, display, and magnetic stripe reader
- 89 = 3608 printer
- 8A = 3615 printer
- 92 = 3616 printer
- 95 = DCA adapter
- 9A = 4710 printer
- AB = Magnetic stripe encoder (4704-2/3)
- B0 = 4720 printer

ss Work station ID (does not apply to host, diskette, or loop)

xxx Counter value, in decimal; counters are numbered from left to right, beginning with counter 1.

When a counter reaches 256, it begins again at 128 (if STATS=WRAP is specified on the STARTGEN macro). Thus, values over 128 might not be definitive.

The counts represent the number of events, not the number of retries. One event can increase more than one counter.

This command requires optional module P25.

069 – Control Terminal Indicator Lights

Command	Operands
069	xx

xx = a hexadecimal value indicating the lights to switch on or off. Enter 069 FF to switch on the keyboard system indicator light; then, press Reset to extinguish that light. Or, enter the hexadecimal value of the following bit settings:

Bit	Explanation
0	Action: 1 = switch on; 0 = switch off
1	Reserved all devices
2	Reserved all devices
3	3616 journal forms insert; ignored for other devices.
4	3616/4710/4720 document insert; other devices, check indicatior.
5	Indicator light 3 (ignored 4710/4720)
6	Indicator light 2
7	Indicator light 1

070 – Diagnostic Event Recording (Trace)

Command	Operands
070	xx xx 7yslot xx

xx = the codes that indicate which types of events you want to record in the trace area. Each code you enter starts a trace on a subsystem component. The system components and their codes are:

Code	Event
0	Host link function request and post
1	Terminal component
2	Host link input and output
3	Diskette input and output
4	Reserved
5	Diskette incidents
6	Reserved
7	Loops (see 7yslot operand, below)
8-9	Reserved
10	Disk incidents
11	DCA function request and post
12	DCA interruption processing
13-23	Reserved

7yslot = only the 7 to trace all loops. Or, you can replace the *y* with the number of a specific loop you want to trace. To trace specific slots on a loop, replace *y* with the loop number, and replace *slot* with a mask indicating the slots to trace.

071 – Stop Recording (Tracing) Diagnostic Events

Command	Operands
071	xx xx xx

xx = the codes that indicate which traces you want to stop. Each code you enter stops a trace on a subsystem component. The system components and their codes are:

Code	Event
0	Host link function request and post
1	Terminal component
2	Host link input and output
3	Diskette input and output
4	Reserved
5	Diskette incidents
6	Reserved
7	All loops.
8-10	Reserved
11	DCA function request and post
12	DCA interruption processing
13-23	Reserved

072 – Display or Print Extended Statistical Counters

Command	Operands
072	code ctrid

code = one of these function codes:

Code	Function
1	Displays the counter identified by *ctrid* or, if the second operand is omitted, displays data from the first extended statistical counter. You can then press the Enter key to display the next counter (host operator, use the ATTN command). The 90036 message indicates the end of data.
2	Prints all extended counters at the output printer assigned by the 005 or 006 command. (the *ctrid* operand is not used.)
3	Displays all device IDs assigned to the extended counter identified in the *ctrid* operand.
4	Displays the extended counter ID associated with the device ID contained in the *ctrid* operand. The device ID is entered in the form *lsdx*, where: *l* = loop number; *s* = terminal address; *d* = component ID; and *x* = subaddress if present, or 0.

ctrid = if *code* = 1 or 3, enter the ID of an extended statistical counter. If *code* = 4, enter the device address in the form *lsdx*, where: *l* = loop number; *s* = terminal address; *d* = component ID; and *x* = subaddress if present, or 0.

The output format for codes 1 and 2 follows:

xxxx-nnnnnnnnnnnn eeeeee ddd pp . . . p

Where:

xxxx = Extended statistical counter.

n . . . nn = Number of input bytes received (in decimal).

ee . . . ee = Number of input bytes received in error (in decimal).

ddd = Number of devices assigned to this counter (in decimal).

pp . . . p = Percentage of error bytes received (bytes in error per bytes received).

The output format for codes 3 and 4 follows:

eeee-lsdx lsdx . . . .

Where:

eeee = Extended statistical counter ID (in hexadecimal).

lsdx = Device ID (in hexadecimal) as previously described. For code 3, all device IDs associated with counter ID *eeee* are displayed.

xxxx = Extended statistical counter ID in hexadecimal (counter IDs are specified at system configuration) or device ID in hexadecimal.

This command requires optional module P25.

074 – Start or Stop DCA

Command Operands

074 code port dtype

code = one of these function codes:

Code	Function
0	Start the adapter.
1	Stop the adapter.
2	Enable port.
3	Disable port.
FF	Change port for specified device. (This is available only on the diagnostic diskette.)

port = the port number (0-7) for codes 2, 3, and FF.

dtype = the DCA device type for code FF only, 0 for a keyboard display, or 1 for a printer.

075 – Display the Status of the System

Command Operand

075

There are no operands.

This command display the format:

lite msgs

which might look like this:

0100 - -++

The *lite* field displays the four operator panel lights. Starting on the left, the lights are:

l The Alert light
i The Ready light
t The Check light
e The Test/IPL light

A 1 indicates that the light is on; a 0 indicates that it is off.

The *msgs* shows the LED message indicators that display the system's operational status or an error message. See – Heading id 'msgs' unknown – for descriptions of these messages.

076 – Display System Variables

Command Operands

076

There are no operands.

The command displays:

IR=yyy EC=eeeeee PAT=pp CPG=ver
GENID=name SERIAL NO=serno CUA=cu

Field	Meaning
yyy	Release level of your Controller Data
eeeeee	EC level
pp	Patch level
ver	Required level of 4700 Host Support
name	CPGEN name (from ID operand of STARTGEN)
serno	Controller serial number
cu	Control unit address of controller

123 – Enter Debugging Mode

Command Operands

123 id

id = the work station ID of the work station to be debugged.

133 – Print Application Program Dump

Command Operands

133 [skip]

skip = the number of application program dumps to skip before printing the dump you want. If your dump is the third dump in the dump data set, enter 133 2, skipping two dumps.

You can enter the value D to print a list of all dump IDs with their associated station IDs.

This command presents the dump data in this format:

dumpid dtatyp st apname
disp pfp spf fli

data data data data data data data data
data data data data data data data data
data data data data data data data data

Where:

dumpid = Specified dump ID
dtatyp = Type of data (T1, T2, T3, T4, L PERM, or SEG xx)
st = Station
apname = APNAME
disp = Displacement into the data
pfp = Primary field pointer
spf = Secondary field pointer
fli = Field-length indicator
data = Data

135 – Display Application Program Dump

Command	Operands
135	[skip]

skip = the number of application program dumps to skip before displaying the dump you want.

You can enter the value D to display a list of the dump IDs and their associated station IDs.

This command present the dump data in this format:

dumpidid dtatyp st apname
disp pfp spf fli

data data data data data data data data
data data data data data data data data
data data data data data data data data

Where:

dumpidid = Specified dump ID
dtatyp = Type of data (T1, T2, T3, T4, L PERM, or SEG xx)
st = Station
apname = APNAME
disp = Displacement into the data
pfp = Primary field pointer
spf = Secondary field pointer
fli = Field-length indicator
data = Data

137 – Compress Diskette on Secondary Drive

Command	Operands
137	[dsname func alin sects]

dsname = the name of a data set to be altered. Omit this operand, and all other operands, to compress the entire diskette.

func = indicates what you want to do to the data set named in the first operand. Do not specify this operand without naming a specific data set. The allowable function codes are:

Code	Function
0	No change.
1	Add to the data set.
2	Subtract from the data set.
3	Delete the entire data set.
4	Extend the end of the data set to the end of the extent + 1.
5	Reduce the data set by moving the end of extent to end of data + 1.

alin = a 0 if you do not want to specify alignment, or a 1 if you want the data set aligned on a track boundary. If you specify a 1, the data set must already begin and end on track boundaries before you issue the command.

The system ignores requests to compress sequential data sets. Track alignment cannot be guaranteed for compressed sequential data sets.

sects = the number of sectors to be added or subtracted when using function codes 1 and 2. You can add or subtract from 0 to 65535 sectors.

138 – Compress Application Program Data Set

Command	Operands
138	

There are no operands.

139 – Format a Diskette on Secondary Drive

Command	Operands
139	[lgth type volid crc seq]

lgth = a code, from the following table, indicating the record length:

Code	Length
0	128 bytes per record
1	256 bytes per record
F	Use existing diskette record length:

type = a code from the following table indicating the diskette type.

Code	Diskette Type
1	Diskette 1
2	Diskette 2
3	Diskette 2D
F	Assume the type already inserted in the drive (default value)

volid = a one- to six-character volume identification ONLY if you want to change the existing volume ID. If you omit this operand, the system retains the existing volume ID.

Notes:

1. You can change only the volume ID by omitting the preceding operands. However, you can do this *ONLY* if the volume ID is longer than one character. If the volume ID is only one character, enter all preceding operands.
2. The following two operands, *crc* and *seq*, are optional operands. If you enter them, you must first enter a volume ID.

crc = the code that indicates the type of record to be written to the diskette.

0 Format with delete control records. This is the default.
1 Format with data records.

Specify 1 (data records) for multi-block I/O on the diskette.

seq = to modify the physical sequence numbering of records on a type 2D diskette. You can modify the sequence numbering to get more diskette accesses per revolution of the diskette. Replace *seq* with a number in the range 0 to 13.

166 – Change Volume ID on Diskette

Command	Operands
166	d [X] volid

d = 1 to use the primary drive, or 2 to use the secondary drive.

volid = the volume ID. Either enter X followed by the volume ID in hexadecimal, or just enter the volume ID in alphameric characters. The volume ID is a 1- to 6-character alphameric value, or 2 to 12 hexadecimal characters entered in pairs.

188 – Copy Diskette

Command	Operands
188	

There are no operands.

202 – Display Secondary Drive Status

Command Operands

202

There are no operands.

231 – Read and Display Secondary Diskette Record

Command Operands

231 [X] ttrr [s]

X = the diskette address in hexadecimal; omit the X to enter the address in decimal.

ttrr = the track (tt) and record (rr) numbers in either decimal or hexadecimal.

s = a 0 to read from side 0, or a 1 to read from side 1. If you omit this operand, the system uses side 0.

232 – Change a Diskette Record on Secondary Drive

Command Operands

232 [X] ttrr [X] dsp data

X = X to specify the operand in hexadecimal; omit the X to enter the operand in decimal.

ttrr = the track (tt) and record (rr) numbers in either decimal or hexadecimal.

dsp = the displacement into the record. You can begin the change up to 255 bytes into the record.

To enter the displacement in hexadecimal, enter the character X followed by the hexadecimal displacement, up to XFF.

data = up to 8 hexadecimal bytes (16 input characters) of data to replace the data in the record. If you enter an odd number of characters, the system sets the leftmost 4 bits to 0000.

233 – Print Dump/Storage Map from Secondary Drive

Command Operands

233 [X] [address]

X = the address in hexadecimal; omit the X to enter the address in decimal.

address = the starting address in decimal or hexadecimal. The command prints the stand-alone dump, beginning at the address you specify.

If you omit this address, the command prints the MEF storage map.

234 – Perform Seek Test on Secondary Diskette Drive

Command Operands

234 [X] ttrr [X] ttrr [s]

X = the address in hexadecimal; omit the X to enter the address in decimal.

ttrr = the track (tt) and record (rr) numbers in either decimal or hexadecimal.

s = a 0 to read from side 0, or a 1 to read from side 1. If you omit this operand, the system uses side 0.

235 – Display Dump or Storage Map from Secondary Diskette

Command Operands

235 [X] address

X = the address in hexadecimal; omit the X to enter the address in decimal.

address = Replace *address* with the starting address in decimal or hexadecimal. The command displays the stand-alone dump, beginning at the address you specify.

If you omit this address, the command displays the MEF storage map.

236 – Write Record to Secondary Diskette

Command Operands

236 [X] ttrr xx [s]

X = the diskette address in hexadecimal; omit the X to enter the address in decimal.

ttrr = the track (tt) and record (rr) numbers in either decimal or hexadecimal.

xx = one or two hexadecimal characters to be written as data in the record. If you omit this operand, 0's are written.

If you enter the *s* operand, you must also enter the *xx* operand.

s = a 0 to read from side 0, or a 1 to read from side 1. If you enter the *s* operand, you must also enter the *xx* operand. If you omit this operand, the system uses side 0.

237 – Reconstruct Temporary File or Index Record on Secondary Drive

Command Operands

237 [X] ttrr [s]

x = the diskette address in hexadecimal; omit the X to enter the address in decimal.

ttrr = the track (tt) and record (rr) numbers in either decimal or hexadecimal.

s = a 0 to read from side 0, or a 1 to read from side 1. If you omit this operand, the system uses side 0.

238 – Print Operating Information from Secondary Drive

Command Operands

238

There are no operands.

243 – Change Test Component Characteristics

Command Operands

243 [ddddmmmmmmccuuoo]

The parameters you specify on the 243 command remain in effect until you change them with another 243 command or until you restart the system.

dddd = the 4-digit device type.

mmmmmm = the model number of the device, in character format, right adjusted and padded with blanks.

cc = the ID that indicates the general type of component. The component IDs are defined as follows:

01	Keyboard
02	Display
03	Magnetic Stripe encoder
04	Printer (first or only print component)
05	Printer (second print components)
06	Financial Service Terminal
07	Document Processor
08	Consumer Transaction Facility

uu = 1 byte of user data, as defined during CPGEN. This can be any value from hex 00 to hex FF.

oo = how the device is owned:

80	In the device pool associated with this station.
40	Shared with another station (non-concurrent).
20	Shared with another station (concurrent).

Device Codes

The following, listed by device, are the possible combinations of devices, models, and components.

Loop Attached Devices

Device	Model	Component	Description
3604	1	01	Keyboard (optional MSR and PIN)
		02	Display
		03	Magnetic Stripe (MSE)
	2	01	Keyboard (optional MSR and PIN)
		02	Display
		03	Magnetic Stripe (MSE)
	3	01	Keyboard (optional MSR and PIN)
		02	Display
		03	Magnetic Stripe (MSE)
	4	01	Keyboard (optional MSR and PIN)
		02	Display
		03	Magnetic Stripe (MSE)
	5	01	Keyboard (optional MSR and PIN)
		02	Display
	6	01	Keyboard (optional MSR and PIN)
		02	Display
	7	01	Keyboard (optional MSR and PIN)
		02	Display
3606		06	Financial Service Terminal (Keyboard, Display, MSR)
3608		04	Printer
		06	Financial Service Terminal (Keyboard, Display, MSR)
3609	1	01	Keyboard (optional MSR and PIN)
		02	Display
		03	Magnetic Stripe (MSE)
	2	01	Keyboard (optional MSR and PIN)
		02	Display
		03	Magnetic Stripe (MSE)
	3	01	Keyboard (optional MSR and PIN)
		02	Display
		03	Magnetic Stripe (MSE)

Device	Model	Component	Description
3610	1	04	Document Printer
	2	04	Document Printer
	3	04	Document Printer
	4	04	Document Printer
	5	04	Document Printer
	12	04	Document Printer
	13	04	Document Printer
3611	1	04	Passbook Printer
	2	04	Passbook Printer
3612	1	04	Document Printer
		05	Passbook Printer
	2	04	Document Printer
		05	Passbook Printer
	3	04	Document Printer
		05	Passbook Printer
	12	04	Document Printer
		05	Passbook Printer (CPGEN MODEL=12P)
	13	04	Document Printer
		05	Passbook Printer (CPGEN MODEL=13P)
3613		04	Journal Printer
		05	Passbook Printer w/MSR (optional MSE and OLF)
3614		08	Consumer Transaction Facility
3615	1	04	Administrative Printer
	2	04	Administrative Printer
3616		04	Journal Print Station
		05	Document Print Station
3619	A01	04	Administrative Printer
	B01	04	Administrative Printer
	B02	04	Administrative Printer
	B03	04	Administrative Printer
3620		03	Magnetic Stripe (MSRE)
		04	Journal Print Station
		05	Document Print Station
3621	B01	04	Statement Printer
	B02	03	Magnetic Stripe (optional Keyboard and PIN)
		04	Statement Printer
	B03	03	Magnetic Stripe (optional Keyboard and PIN)
		04	Statement Printer
3624		08	Consumer Transaction Facility

Device	Model	Component	Description
4704	1	01	Keyboard (optional MSR and PIN; CPGEN MODEL=11 or 12)
		02	Display (CPGEN MODEL=11 or 12)
		03	Magnetic Stripe (MSE CPGEN MODEL=11 or 12)
4710		04	Receipt/Validation Printer
4713		04	Verification Printer
		05	See DCA-Attached Devices for this Component
4720	1	04	Cutform Printer
	2	04	Cutform Printer with Journal
	3	04	Cutform/Passbook Printer
	4	04	Cutform/Passbook Printer with Journal
4723		07	Document Processor
5906		06	Financial Services Terminal (Keyboard, Display, MSR)
5922		07	Document Processor

DCA Attached Devices

Device	Model	Component	Description
3262		04	Line Printer
3268		04	Line Printer
3278	2	01	Keyboard (optional MSR; CPGEN MODEL=8)
		02	Display (CPGEN MODEL=8)
	52	01	Keyboard (optional MSR; CPGEN MODEL=E)
		02	Display (CPGEN MODEL=E)
3279	2A	01	Keyboard (optional MSR; CPGEN MODEL=8A)
		02	Display (CPGEN MODEL=8A)
	2B	01	Keyboard (optional MSR; CPGEN MODEL=8B)
		02	Display (CPGEN MODEL=8B)

Device	Model	Component	Description
3283		04	Printer
3287		04	Printer
4704	2/3	01	Keyboard (optional MSR and PIN) (CPGEN MODEL=21 or 22)
		02	Display (CPGEN MODEL=21 or 22)
		03	Magnetic Stripe (MSE; CPGEN MODEL=21 or 22)
4713		04	See Loop Attached Devices for this Component
		05	Passbook Printer w/MSR (optional MSE and OLF)
5210		04	Printer
7436		04	Printer
7486	2	01	Keyboard (CPGEN MODEL=8)
		02	Display (CPGEN MODEL=8)

301 – Display Message Type

Command	Operands
301	type

type = the 3-digit message-type code.

302 – Display Message Types in the System Log

Command	Operands
302	

There are no operands.

310 – Retrieve Work Station Status

Command	Operands
310	id lssdd

id = the work station ID associated with the program to be debugged.

lssdd = the address of the terminal using the program, consisting of the loop number (1) or A for a DCA device, the terminal or port address (ss), and the component address (dd). Always enter the leading 0 for the *ss* part of the operand. Enter component address as:

dd	Component
Fixed Addresses:	
1	4704/3604/3278/3279 keyboard
2	4704/3604/3278/3279 display
3	Magnetic stripe encoder
4	3610/3612 document printer; 3262/3287/5210 printer
5	3611/3612 passbook printer
6	3606/3608 keyboard, display, magnetic stripe reader
7	3608 printer
8	3614/3624 terminal
Variable Addresses:	
1-15	4710/3615 printer (must be same value as the subaddress switch settings on the printer).
2-14	Even number only, 4720, 3616 journal printer station (must be set same as subaddress switch settings).
3-15	Odd numbers only; 3616 document print station.

320 – Test Cryptographic Facilities

Command	Operands
320	

There are no operands.

330 – Load, Verify, Erase Cryptographic Keys

CAUTION
Work carefully. If you turn the key clockwise from its initial, vertical position, you might erase all of the keys.

Command	Operands
330	func key

func = one of these codes to indicate what you want to do.

Code	Function
1	Load encrypted keys. You cannot use this option with a master key.
2	Load keys in two parts from two operators. A send cross-domain key must be encrypted under the first variant of the master key. A receive cross-domain key must be encrypted under the second variant of the master key.
3	Verify keys.
4	Erase keys.

key = one of these codes to indicate which key you want to load, verify, or erase.

Code	Key
1	Master key (KM) You cannot use this value with a *func* value of 1.
3	Send cross-domain key (KCD1)
4	Receive cross-domain key (KCD2)

711 – Send Message to Remote Operator

Command	Operands
711	opid text

opid = the NCCF network identifier of the remote operator.

text = the message text to send to the remote operator.

712 – Control Access By CNM/CS

Command	Operands
712	[func secs]

func = one of these control codes:

Code	Function
0	Disable CNM/CS processing; prevent access from host.
1	Enable CNM/CS access; DO NOT allow access to alert processing.
2	Enable CNM/CS access to alert processing only.
3	Enable all CNM/CS access.

secs = a value in the range 10 to 65535 indicating the maximum time (in seconds) between host queries for loop or alert status.

Enter the 712 command with no operands to see the values already assigned to these codes.

777 – Immediate Controller Dump

Command	Operands
777	password

password = the system monitor password.

778 – Add Trap-After-Store Table Entry

Command	Operands
778	password eq sp [X] address mask value

password = the system monitor password.

eq = a 1 to trap on an equal comparison, or a 0 to trap on an unequal comparison.

sp = a 1 to check for register space, or a 0 to check storage space.

address = an even (real or mapped) address in either decimal or hexadecimal. This is the address at which the trap takes place.

mask = the 4 hexadecimal digits to be ANDed with the halfword at the location specified by the *address* operand.

value = the 4 hexadecimal digits to be compared with the result of the AND operation.

779 – Activate or Deactivate Trap-After-Store

Command	Operands
779	password xx

password = the system monitor password.

xx = the 2 hexadecimal character representing bits (binary indicators) set in the following pattern.

Bit	Meaning
bit 0	Set to 1 to activate, 0 to deactivate the trap function when each executor returns to the function interpreter.
bits 1-7	Set to 1 to activate, 0 to deactivate, the trap before and after each routine gets control by the CLIH for level 1 to 7, respectively.

Set *xx* = 00 to deactivate the function and clear the trap-after-store table.

888 – Transmit Diskette

Command	Operands
888	[drive]

drive = a 1 for the primary drive, or a 2 for the secondary drive. If you omit this operand, 1 is used.

929 – Test Host Link

Command	Operands
929	rpt stop test address

rpt = the number of times to repeat the test

stop = a 0 to stop the test when an error occurs, or a 1 to continue the test when status 0203 (wrap failed) is received.

test = a 01 to perform the adapter wrap test, or 02 to test the modem.

address = the adapter address, 15 for the host link.

936 – Write Record to Primary Diskette

Command	Operands
936	[X] ttrr xx [s]

X = the diskette address in hexadecimal; omit the X to enter the address in decimal.

ttrr = the track (tt) and record (rr) numbers in either decimal or hexadecimal.

xx = the one or two hexadecimal characters to be written as data in the record. If you omit this operand, 0's are written.

If you enter the *s* operand, you must also enter the *xx* operand.

s = a 0 to read from side 0, or a 1 to read from side 1.

If you enter the *s* operand, you must also enter the *xx* operand. If you omit this operand, the system uses side 0.

937 – Rebuild Temporary File/Index Record on Primary Drive

Command	Operands
937	[X] ttrr [s]

X = the diskette address in hexadecimal; omit the X to enter the address in decimal.

ttrr = the track (tt) and record (rr) numbers in either decimal or hexadecimal.

s = a 0 to read from side 0, or a 1 to read from side 1. If you omit this operand, the system uses side 0.

955 – Copy Daya Set

Command Operands

955 [x]

x = 1 to copy from the primary to the secondary diskette; 2 to copy from the secondary to the primary diskette. If you do not enter the operand, the system uses 1.

973 – Activate or Deactivate Loop Device for Testing

Command Operands

973 [lssdd]

lssdd = the device address, consisting of the loop number (l), the terminal address (*ss* can be 02 for modulus 4, or 04 for modulus 8). Always enter the leading 0 for the *ss* part of the operand. The device type can be:

Code	Device Type
91	3604
92	3610, 3611, 3612
93	3614, 3624
94	4720
95	3608, 3606
96	3615
97	3616
98	4704
99	4710

Omit the operand to deactivate all active devices. Enter a terminal address to activate the new terminal and deactivate the previous test terminal.

980 – Create an Exec File

Command Operands

980

There are no operands.

981 – Repeat Previous Commands

Command	Operands
981	rpt

rpt = a number, from 1 to 3 digits, indicating the number of times the system monitor is to repeat the previous commands in this EXEC file, when the EXEC file is invoked.

982 – Invoke an Exec File

Command	Operands
982	execid

execid = the 4-digit EXEC message ID that was displayed when you created the EXEC with the 980 command.

990 – Display Diskette Data Set Names

Command	Operands
990	[drive]

drive = 0 for the primary drive, 1 for the secondary drive. If you omit this operand, the system uses 0.

991 – Display Application Program Names and Addresses

Command	Operands
991	

There are no operands.

998 – Prepare for Unattended Creation

Command	Operands
998	

There are no operands.

999 – Create Operating Diskette

Command	Operands
999	[init drv volidcuacopidtransid]

init = a 0 to have the host initiate the session, or a 1 to have the controller initiate the session. If you enter 1, the Host Transmit Facility, at the host computer, must be active.

drv = 0 to use the diskette on the primary drive; enter a 1 if you want to insert a new diskette on the primary drive; enter 2 to use the secondary drive. If you omit this operand, the system uses the diskette on the primary drive (0).

volidcuacopidtransid This compound operand consists of the:

- volid Volume ID to be written on the diskette. If you omit this, 6 blanks are written.
- cua Control unit address to be written on diskette. If you omit this, X'40' is written.
- copid Control operator ID, consisting of 1 to 16 characters, which can be 0-9, A-F, X, and blank. If you omit this, 16 blanks are written. If you enter the *transid* operand, you must enter this *copid* operand as a 16-character ID.
- transid Transmission ID (XID) for the switched network. If you omit this, X'00000' is written. If you enter this operand, you must enter the previous *copid* operand as a 16-character ID.

Debugging Commands

This section describes the 2-digit debugging commands that you can use after you enter the 123 system monitor command to begin debugging a logical work station.

00 Debugging Command: Leave Debugging Mode

Command	Operands
00	

There are no operands.

01 Debugging Command: Display 8 Bytes of Data

Command	Operands
01	seg [X] disp [id]

seg = the number of the segment (0-15) that contains the data to be displayed. Enter 14 to display the constants section of a split application program. Enter 140 to display the instruction section.

disp = the displacement in decimal, or an X followed by the displacement in hexadecimal.

id = 1 or 2 digits that represent the segment space ID used during the processing of an APCALL instruction in the application program. If you omit this operand, the system uses the current segment space ID.

02 Debugging Command: Write 8 Bytes of Data

Command	Operands
02	seg [X] disp data [id]

seg = the number of the segment (0-15) to which data is to be written. Enter 14 to write in the constants section of a split application program. Enter 140 to write in the instruction section.

disp = the displacement in decimal, or an X followed by the displacement in hexadecimal.

data = up to 8 bytes of data. If you enter an odd number of bytes, the system adds 4 bits of zeros to the left.

id = 1 or 2 digits that represent the segment space ID used during the processing of an APCALL instruction in the application program.

03 Debugging Command: Place Stop In Application Program

Command	Operands
03	[X] disp [progname]

disp = the decimal address, or an X followed by the hexadecimal address where you want to place a stop. When the program reaches this address, execution stops.

progname = the application program name as coded on the BEGIN instruction. If you omit this name, the system puts the stop in the application program currently associated with the work station.

You can enter this name in either EBCDIC or hexadecimal. If the name is 8 characters or less, the system assumes EBCDIC. If the name is 16 characters, the system assumes hexadecimal, and converts it to EBCDIC.

04 Debugging Command: Remove Stop

Command	Operands
04	[X] disp [progname]

disp = the decimal address, or an X followed by the hexadecimal address where you want to remove a stop. If you enter 0, the system removes all stops.

progname = the application program name as coded on the BEGIN instruction. If you omit this name, the system removes the stop in the application program currently associated with the work station.

You can enter this name in either EBCDIC or hexadecimal. If the name is 8 characters or less, the system assumes EBCDIC. If the name is 16 characters, the system assumes hexadecimal, and converts it to EBCDIC.

05 Debugging Command: Start a Stopped Work Station

Command	Operands
05	[[X]disp]

disp = the instruction address in the stopped application program where you want operation to continue. Enter the displacement in decimal, or enter an X followed by the displacement in hexadecimal.

If you omit this operand, the program resumes with the stopped instruction.

06 Debugging Command: Identify Attached Terminal Components

Command	Operands
06	id

id = the work station ID.

07 Debugging Command: Display Header Segment

Command	Operands
07	seg [id]

seg = the segment number (00-15).

id = 1 or 2 digits representing the segment space ID used during the processing of an APCALL instruction in the application program. If you omit this operand, the system uses the current segment space ID.

08 Debugging Command: Stop Work Station

Command	Operands
08	

There are no operands.

09 Debugging Command: Specify Operator A or B

Command	Operands
09	op

op = 0 for operator A, or 1 for operator B.

10 Debugging Command: Change Segment Header Data

Command	Operands
10	seg [id] code [X] data

seg = the 2-digit segment number (00-15). For segment 00, the system assumes that you want operator A data unless you use the 09 command to request operator B data.

id = the 2-digit segment space ID used in APCALL processing.

code = a code to indicate:
- 1 Change primary field pointer.
- 2 Change field length.
- 3 Change secondary field pointer.

data = the value in decimal, or an X followed by the value in hexadecimal, that is to replace the current setting.

11 Debugging Command: Display Data At Byte Address

Command	Operands
11	[X] address

address = the byte address in decimal, or enter an X followed by the address in hexadecimal.

12 Debugging Command: Change Data At Byte Address

Command	Operands
12	[X] address data

address = the byte address in decimal, or enter an X followed by the byte address in hexadecimal.

data = the 8 bytes of data that you want written at the specified byte address.

13 Debugging Command: Begin Single-Cycle/Hard-Copy Trace

Command	Operands
13	[x]

x = any character to begin the hard-copy trace.

14 Debugging Command: Stop Single-Cycle/Hard-Copy Trace

Command	Operands
14	

There are no operands.

15 Debugging Command: Test Indexing Status

Command	Operands
15	[seg]

seg = the number of the segment (0-15) you want to test. If you omit this operand, only the station itself is tested.

16 Debugging Command: Find and Display Data Pattern

Command	Operands
16	data [[X]address1 [Xaddress2]

data = up to 16 characters. The system searches the work station for this pattern.

address1 = the address in decimal or hexadecimal where you want the scan to begin. If you omit this operand entirely, the scan begins at 0.

address2 = the address in decimal or hexadecimal where you want the scan to stop. If you omit the operand, the scan stops at the end of storage.

17 Debugging Command: Display Log Message Text

Command	Operands
17	[X] [msid]

X = X to display the message text in hexadecimal; omit the X to see the text in EBCDIC.

msid = the ID of the message you want displayed. The command displays that message, and several preceding messages.

If you omit this operand, the system begins with the current message.

18 Debugging Command: Display Log Message

Command	Operands
18	[X] msid

X = X to display up to 504 bytes of the message text in hexadecimal. Omit this operand to display up to 252 characters in EBCDIC. If the message is larger than the screen size, the system overlays the message number and beginning of the text with the end of the text.

msid = the 4-digit message ID of the log message to be displayed.

19 Debugging Command: Activate Stops In Program

Command	Operands
19	code

code = a 1 to activate all stops for all stations; a 0 to deactivate all stops for all stations.

20 Debugging Command: Display Stops In Application Program

Command	Operands
20	

There are no operands.

21 Debugging Command: Display Application Program Stack

Command	Operands
21	[level]

level = the level number of the entry you want to see. If you omit this operand, the system displays the current entry.

Chapter 7. Messages Displayed on the Controller Operator Panel

A detailed description is in the *IBM 4700 Finance Communication System; Subsystem Operating Procedures.*

MESSAGE	*MEANING*
A-OK	If you are using the diagnostic diskette, the controller adapter test completed without error. Open and close the primary diskette drive handle.
C00x	You pressed the Interrupt key; the *x* indicates the interrupt level. If a dump is not required, press the Reset switch on the controller to reload the system. If a dump is required, remove the diskette from the primary diskette drive. Press and release the Interrupt key to display the D1xx message. Insert a 256-byte formatted diskette into the primary diskette drive. The dump begins when the diskette becomes ready. When the dump is complete, the system displays the D2xx message.
C1xx	A controller error occurred; the *xx* is the trap code. Request a dump. Press and release the Interrupt key to display the D1xx message. **Note:** Write the D1xx message number on the diskette label because this diskette can no longer be used to load the system.
C2xx	Storage parity failed; the *xx* indicates the 64K sections of storage that have bad parity. Record the message and press Reset to restart; no dump is required. Notify your service personnel.
C4xx	An invalid sequence was detected between the processor and an adapter. The *xx* is the adapter address. Notify your service personnel. A dump is not required.
C8xx	An input parity error occurred; the *xx* is the adapter address. A dump is not required.

D001 The controller diskette drive is not ready, has the wrong type of diskette (2-sided diskette on a 1-sided drive), or no diskette. If you want to IPL from the primary diskette drive, insert the diskette into the primary diskette drive. If you want to IPL from the secondary diskette drive, press the Interrupt key.

D002 The auxiliary diskette drive is not ready, does not exist, has the wrong type of diskette, or no diskette.

D003 Adapter detected diskette error. Try another diskette. If the message continues to appear, the controller requires service.

D004 The diskette is not an operating diskette. Verify that the diskette you are using is the correct diskette. If it is the correct diskette, create a new operating diskette. If you get this message again, notify your service personnel.

D005 A diskette hash checksum error occurred, indicating a bad diskette or the wrong diskette. Verify that the diskette you are using is the correct diskette. If it is the correct diskette, create a new operating diskette. If you receive this message again, consult your service personnel.

D006 Diagnostic and startup microcode is being checked. If the D006 message is blinking, the test failed. Verify that the diskette you are using is the correct diskette. If it is the correct diskette, create a new operating diskette. If you receive this message again, consult your service personnel.

D1xx The controller is ready to dump; *xx* is the next segment to be dumped. If a dump is not required, press the Reset switch on the controller.

If a dump is required, use a 256-byte formatted diskette for the dump.

D200 Controller dump is complete.

D201 The controller dump is not complete, but dumping to the current diskette is complete.

D202 An output error occurred when dumping to the diskette. Remove the diskette, and continue the dump process as described by the message displayed when the diskette is removed.

D203 The loaded diskette is not a 256-byte formatted diskette.

D600 The diskette adapter test is ready to test the secondary diskette drive. If you used the secondary drive to load the system, the D600 indicates the primary diskette drive. Move the diagnostic diskette from the primary drive to the secondary drive.

D601 Testing of the secondary diskette adapter is complete. Move the diagnostic diskette from the secondary diskette drive to the primary diskette drive.

E000 The system is testing the processor and the first 64K bytes of storage. Record the message and press the Reset switch. If the message continues to occur, refer to the *IBM 4700 System Problem Determination Guide.*

E001 The system is testing the second 64K bytes of storage. Record the message and press the Reset switch. If the message continues to occur, refer to the *IBM 4700 System Problem Determination Guide.*

E002 The system is testing the third 64K bytes of storage. Record the message and press the Reset switch. If the message continues to occur, refer to the *IBM 4700 System Problem Determination Guide.*

E003 The system is testing the fourth 64K bytes of storage. Record the message and press the Reset switch. If the message continues to occur, refer to the *IBM 4700 System Problem Determination Guide.*

E004 The system is testing the fifth 64K byte of storage. Record the message and press the Reset switch. If the message continues to occur, refer to the *IBM 4700 System Problem Determination Guide.*

E005 The system is testing the sixth 64K bytes of storage. Record the message and press the Reset switch. If the message continues to occur, refer to the *IBM 4700 System Problem Determination Guide.*

E006 The system is testing the seventh 64K bytes of storage. Record the message and press the Reset switch. If the message continues to occur, refer to the *IBM 4700 System Problem Determination Guide.*

E007 The system is testing the eighth 64K bytes of storage. Record the message and press the Reset switch. If the message continues to occur, refer to the *IBM 4700 System Problem Determination Guide.*

E008 The system is testing the ROS (Read Only Storage) check sum and parity. Record the message and press the Reset switch. If the message continues to occur, refer to the *IBM 4700 System Problem Determination Guide.*

E009 Many adapter failures occurred.

Action: Record the message and press the Reset switch. If the message continues to occur, refer to the *IBM 4700 System Problem Determination Guide.*

E00A The system is testing the cryptographic storage. Make sure the key is not in the encryption key lock. If it is not in the encryption key lock, record the message and press the Reset switch. If the message continues to occur, refer to the *IBM 4700 System Problem Determination Guide.*

E020 The system is testing the system control card — operator panel. Record the message and press the Reset switch. If the message continues to occur, refer to the *IBM 4700 System Problem Determination Guide.*

E021-E0B4 The system is testing the controller configuration. Failure to proceed past these messages indicates a conflict between the configuration description and the installed adapters. Report any problem to your service representative.

E101 The DCA adapter test detected a controller problem. Press the Reset switch to retry the test. If the message continues to appear, the controller requires service.

E155 One or more of the diagnostic test switches appear to be set incorrectly. Set all switches off and press Reset. If the message continues to appear, the controller requires service.

E201 The host-adapter test detected a controller problem during testing with the diagnostic diskette. Press reset to retry the test. If the message continues to appear, the controller requires service.

E202 The host-adapter wrap test detected a controller problem during testing with the diagnostic diskette. Press Reset to retry the test. If the message continues to appear, the controller requires service.

E203 The host-adapter cable wrap test detected a controller problem during testing with the diagnostic diskette. Check to see that:

- The wrap switch is not in test position (EIA).
- The wrap plug is not in place (X.21 switched or Multi-Use loop).
- A powered-on wrappable modem is not properly connected (X.21 non-switched).
- There is a controller cable connector failure.

You can:

1. Retry the test by opening and closing the primary diskette drive handle.
2. Ensure that the required wrap-test is in place.
3. Press Reset to retry.

If the message continues to appear, the controller requires service.

E204 A host link adapter setup error occurred while you were using the diagnostic diskette to test the host link. The controller requires service.

E209 A parameter error occurred while you were using the diagnostic diskette to test the host link. The diagnostic diskette is damaged or a diskette read error occurred, or the required link module was not loaded. Press Reset to retry the test.

E411 The loop-1 adapter test detected a problem while you were testing the adapter with the diagnostic diskette. The controller requires service.

E412 The loop-1 adapter test detected a problem while you were using the diagnostic diskette to test the adapter. Unplug the loop-1 connectors at the rear of the controller. Press Reset to rerun the test. If the message continues to appear, the controller requires service. Refer to the *IBM 4700 System Problem Determination Guide* for loop and terminal problems.

E413 The loop-1 signal propagation delay is too long. Switch off one device at a time to determine the maximum number of devices that you can have on a loop. If the error continues, the controller needs service.

E421 The loop-2 adapter test detected a problem while you were using the diagnostic diskette to test the adapter. The controller requires service.

E422 The loop-2 adapter test detected a problem while you were using the diagnostic diskette to test the adapter. Unplug the loop-2 connectors at the rear of the controller. Press Reset to rerun the test. If the message continues to appear, the controller requires service.

E423 The loop-2 signal propagation delay is too long. Too many devices are attached to the loop. Switch off one device at a time to determine the maximum number of devices that you can have on a loop. If the error continues, the controller needs service.

E431 The loop-3 adapter test detected a problem while you were using the diagnostic diskette to test the adapter. The controller requires service.

E432 The loop-3 adapter test detected a problem while you were using the diagnostic diskette to test the adapter. Unplug the loop-3 connectors at the rear of the controller. Press Reset to rerun the test. If the message continues to appear, the controller requires service.

E433 The loop-3 signal propagation delay is too long. Too many devices are attached to the loop. Switch off one device at a time to determine the maximum number of devices that you can have on a loop. If the error continues, the controller needs service.

E441 The loop-4 adapter test detected a problem while you were using the diagnostic diskette to test the adapter. The controller requires service.

E442 The loop-4 adapter test detected a problem while you were using the diagnostic diskette to test the adapter. There is a problem with loop 4, with a terminal attached to loop 4, or with the controller. Unplug the loop-4 connectors at the rear of the controller. Press Reset to rerun the test. If the message continues to appear, the controller requires service.

E443 Loop-4 signal propagation delay is too long. Too many devices are attached to the loop. Switch off one device at a time to determine the maximum number of devices that you can have on a loop. If the error continues, the controller needs service.

E500-E5A8 An error occurred on the disk. Press Reset to retry the test. If this message continues, refer to the *IBM 4700 System Problem Determination Guide,* and notify your service personnel. To bypass the test, open and close the primary diskette drive handle; the test sequence continues with the next test.

E601 A write error occurred on the primary diskette side 0, track 73. Press Reset to retry the test. If the message continues to occur, try another diskette.

E602 A read error occurred on the primary diskette, side 0, track 73. Press Reset to retry. If the message continues to occur, try another diskette.

E603 The read/write data on side 0, track 73 did not compare. Press Reset to retry the test. If the message continues to occur, try another diskette.

E604 A read error occurred after the D601 message. Press Reset to retry the test. If the message continues to occur, try another diskette.

E611 A write error occurred on the primary diskette side 1, track 73. Press Reset to retry the test. If the message continues to occur, try another diskette.

E612 A read error occurred on the primary diskette, side 1, track 73. Press Reset to retry. If the message continues to occur, try another diskette.

E613 The read/write data on side 1, track 73 did not compare. Press Reset to retry the test. If the message continues to occur, try another diskette.

E620 The auxiliary diskette drive was not ready within 1 minute after the D600 message signaled the request to transfer the diagnostic diskette to the auxiliary drive.

E621 A write error occurred on the auxiliary diskette side 0, track 73. Press Reset to retry the test. If the message continues to occur, the controller or auxiliary diskette drive requires service.

E622 A read error occurred on the auxiliary diskette side 0, track 73. Press Reset to retry the test. If the message continues to occur, the controller or auxiliary diskette drive requires service.

E623 The read/write data on side 0, track 73, did not compare. Press Reset to retry the test. If the message continues to occur, the controller or auxiliary diskette drive requires service.

E624 A read error occurred while you were using the diagnostic diskette. Press Reset to retry the test. If the message continues to occur, try another diskette. If this does not correct the problem, the controller or auxiliary diskette drive requires service. You can bypass the test by opening and closing the primary diskette drive handle.

E631 A write error occurred on the auxiliary diskette side 1, track 73. Press Reset to retry the test. If the message continues to occur, the controller or auxiliary diskette drive requires service.

E632 A read error occurred on the auxiliary diskette side 1, track 73. Press Reset to retry the test. If the message continues to occur, the controller or auxiliary diskette drive requires service.

E633 The read/write data on side 1, track 73, did not compare. Press Reset to retry the test. If the message continues to occur, the controller or auxiliary diskette drive requires service.

E901 You made an invalid request to the diagnostic control console (address 1, loop 1). Reenter using the correct input.

E902 A read error occurred on the diagnostic diskette while an overlay was being loaded. Enter the command again. If you get the message again, the controller or auxiliary diskette drive requires service.

E903 An incorrect read from the diagnostic control console (address 1, loop 1) occurred. Reenter command, if the error continues to occur, refer to the *IBM 4700 System Problem Determination Guide* for correction of loop or terminal problems.

E905 An error occurred while you were writing to the diagnostic control console. Check that the display at loop 1, address 1 is ready.

E909 An invalid address was entered from the control console. Reenter the command using the correct input.

E916 You requested the wrong test routine from the diagnostic diskette. Enter the correctly formatted input request.

E932 Station read or write error occurred during system monitor programmable input mode while you were using the diagnostic diskette. Retry. If the error continues, request a dump.

E933 System monitor already used at another terminal. Log off the other terminal and retry the logon.

E960 A timer interrupt occurred while you were using the diagnostic diskette. Retry. If the error continues to occur, request a dump.

E961 A program check occurred while you were using the diagnostic diskette.

Action: Load optional module 6A or suppress the host-link test.

I00v The system is going through startup, *v* is the controller storage size in 64K-byte units. This is an informational message.

I100 Loading base microcode. This is an informational message.

I200 The system is loading CPGEN. This is an informational message.

I3nn The system is processing or modifying CPGEN data; *nn* is the module identifier. This is an informational message.

I4nn The system is loading system application programs; *nn* is the sequence number of the system application program being loaded. This is an informational message.

I5nn The system is loading application programs; *nn* is the sequence number of the application program being loaded. This is an informational message.

I601 The system is allocating the initialized segment headers requested by the application program. This is an informational message.

I602 This system is allocating MAXSTOR and segment space to the stations, and connecting the segment space to the segment headers. This is an informational message.

I603 The system is initializing the segment as specified by the CPGEN or the application program. This is an informational message.

I698 The system is getting ready to run the system monitor. This is an informational message.

I699 The system is prompting for startup information at a terminal. This is an informational message. You might be required to respond to the 00001 message. Or, you can either press Enter or wait 2 minutes for the system to run the standard startup procedure.

I7nn The system is loading optional module *nn*. This is an informational message.

I8nn The system is loading and running startup overlay module *nn*. This is an informational message.

I9nn The system is loading communications module *nn*. This is an informational message.

T000 This is the first message displayed by the system monitor on the diagnostic diskette.

T100 This is the first message from the DCA adapter test when you IPL using the diagnostic diskette.

T199 The DCA adapter test is complete.

T200 This is the first message from the host adapter test when you IPL using the diagnostic diskette. This is an informational message.

T299 The host adapter test is complete. This is an informational message.

T400 This is the first message from the loop adapter test when you IPL using the diagnostic diskette. This is an informational message.

T410 The system is testing loop 1. This is an informational message.

T420 The system is testing loop 2. This is an informational message.

T430 The system is testing loop 3. This is an informational message.

T440 The system is testing loop 4. This is an informational message.

T499 The loop adapter test is complete. This is an informational message.

T500 The disk test is beginning. This is an informational message.

T599 The disk test is complete. This is an informational message.

T600 The diskette adapter test started. This is an informational message.

T699 Diskette adapter testing is complete. This is an informational message.

T998 Diagnostic testing is complete. When this appears on the control console (display on loop 1, address 1), you can log onto the system monitor or invoke another test. Press the Reset three times to log on the system monitor, or enter a manual test request.

T999 Adapter testing has completed with retry or error bypass. Open and close the primary diskette drive handle to change the controller display to the operational state.

XCxx A machine check or microcode check occurred during early phase of startup. The *xx* identifies the specific cause of the malfunction. Record the message and press the Reset switch. If you get the message again, load the diagnostics diskette and press the Reset switch.

X011 The SYSBAS diskette data set cannot be found. The startup process cannot proceed. Verify that the diskette you are using is the correct diskette. If it is the correct diskette, create a new operating diskette. If you get this message again, inform your service personnel.

X012 The SYSOPT diskette data set cannot be found. The startup process cannot proceed. Verify that the diskette you are using is the correct diskette. If it is the correct diskette, create a new operating diskette. If you get this message again, inform your service personnel.

X013 The CPGEN data is not compatible with the microcode on the diskette. The startup ends. Correct the host libraries, and re-create the operating diskette.

X014 The SYSCPG diskette data set cannot be found. The startup process cannot proceed. Verify that the diskette you are using is the correct diskette. If it is the correct diskette, create a new operating diskette. If you get this message again, inform your service personnel.

X015 The SYSST1 diskette data set cannot be found. The startup process cannot proceed. Verify that the diskette you are using is the correct diskette. If it is the correct diskette, create a new operating diskette. If you get this message again, inform your service personnel.

X017 The SYSSM diskette data set cannot be found. Verify that the diskette you are using is the correct diskette. If it is the correct diskette, create a new operating diskette. If you get this message again, inform your service personnel.

X018 The temporary or permanent diskette data set cannot be found or has an invalid data set label. The startup process cannot proceed. Re-create the operating diskette with VTAM traces active.

X01B The microcode level and controller EC level are not compatible. Try another operating diskette. If this fails, report the problem to your service personnel.

X01D The diskette adapter control code is missing. Report the problem to your service personnel.

X020 The control operator failed to respond to message 00001 within the predefined time-out, and the system attempted an automatic startup. If you do not want a dump, press the Reset switch on the controller.

If you want a dump, open the cover and press the Interrupt key. Insert a 256-byte formatted diskette into the controller. Do not press the Interrupt key again. The dump begins when the diskette becomes ready.

When the dump is complete, the D1xx message is replaced with the D200 message.

Note: Write the D1xx message number on the diskette label because this diskette can no longer be used to load the system.

X021 A diskette read error occurred while you were accessing the SYSCTL data set. The diskette is physically or magnetically damaged. Try another operating diskette.

X022 The system is unable, or for security reasons not permitted, to write to the SYSCTL data set. The diskette was unloaded or the diskette was physically or magnetically damaged. If the diskette was not ready, make it ready and press Reset. If the diskette was ready, retry using another operating diskette.

X030 There is not enough controller storage to execute startup process. Insert the installation diskette. If you still get an error, there is a control storage problem. If the installation diskette loads without error, create a new operating diskette.

X031 There is not enough storage to load control program. Insert the installation diskette. If you still get an error, there is a control storage problem. If the installation diskette loads without error, create a new operating diskette.

X032 There is insufficient storage to build an application program table. The startup process cannot proceed. Decrease the size of CPGEN or add storage to the controller.

X033 There is not enough storage to load optional or link microcode modules. The startup process cannot proceed. Decrease the size of CPGEN or the application program, selectively load optional modules, or add storage to the controller.

X034 There is not enough storage to load CPGEN. The startup process cannot proceed. Decrease CPGEN size or add storage to the controller.

X035 There is not enough storage to load the application program constant section. The startup process cannot proceed. Decrease CPGEN size or add storage to the controller.

X036 There is not enough storage to load the application program instruction section. The startup process cannot proceed. Decrease CPGEN size or add storage to the controller.

X037 There is not enough storage to load the dummy application program. The startup process cannot proceed. In the CPGEN, some stations are not assigned to any of the application programs contained on the operating diskette. Modify the CPGEN to assign stations to the available application programs. Otherwise, decrease the size of the CPGEN or the application programs, or add storage to the controller.

X038 There is not enough storage to define the diskette index counters. The startup process cannot proceed. Decrease the size of CPGEN or add storage to the controller.

X03A There is not enough storage to load data stream mapping (DATSM) image buffers. Decrease CPGEN size or add storage to the controller.

X03B There is not enough storage to load scratch pad area (SPA) buffers. Decrease CPGEN size or add storage to the controller.

X03C There is not enough storage for interval-timer buffers. Decrease CPGEN size or add storage to the controller.

X041 A checksum error was detected while you were loading the controller microcode. Try another operating diskette.

X042 A checksum error was detected while loading the optional module, startup overlay, or communication link module. The startup process cannot proceed. The previous I*nnn* message identifies the failing microcode module. Try another operating diskette.

X046 A checksum error was detected while loading the system application programs. Try another operating diskette.

X050 The system is unable to initialize the diskette drive with the current diskette (if there is one in the diskette drive) or unable to read either the data set label or the control data set. Startup cannot continue. If the diskette was not ready, make it ready and press Reset. If the diskette was ready, retry with another operating diskette.

X051 A control program diskette read error occurred. The startup process cannot proceed. If the diskette was not ready, make it ready and press Reset. If the diskette was ready, retry with another operating diskette.

X053 A diskette error occurred while the system was loading an optional module, startup overlay, or communication link module. If the diskette was not ready, make it ready and press Reset. If the diskette was ready, retry with another operating diskette.

X054 A CPGEN diskette read error occurred. The startup process cannot proceed. If the diskette was not ready, make it ready and press Reset. If the diskette was ready, retry with another operating diskette.

X056 An application program diskette read error occurred. The startup process cannot proceed. If the diskette was not ready, make it ready and press Reset. If the diskette was ready, retry with another operating diskette.

X057 A diskette read error occurred in the error map phase of startup. The startup process cannot proceed. If the diskette was not ready, make it ready and press Reset. If the diskette was ready, retry with another operating diskette.

X058 A diskette read error occurred in the advanced phase of startup. The startup process cannot proceed. If the diskette was not ready, make it ready and press Reset. If the diskette was ready, retry with another operating diskette.

X059 A diskette read error occurred while the system was loading a startup overlay used in modifying CPGEN data. If the diskette was not ready, make it ready and press Reset. If the diskette was ready, retry with another operating diskette.

X060 The SYSLCF diskette data set cannot be found. The startup process cannot proceed. Reinitialize the operating diskette with the installation diskette using the LCF create diskette option. Load the system from the diskette.

X061 There is not enough storage for LCF startup to construct the defined configuration. The startup process cannot proceed. Decrease size of configuration, or add storage to the controller.

X062 A checksum error was detected while loading the SYSLCF data set. The startup process cannot proceed. Try with another LCF pre-operating diskette.

X063 A diskette read error occurred while the system was loading startup control data used by LCF in building CPGEN data. The startup process cannot proceed. If the diskette was not ready, make it ready and press Reset to retry the procedure. If you want to use another diskette, retry with another pre-operating diskette.

X064 The LCF data required for the LCF startup process does not agree with the data that was expected. The startup process cannot proceed. Re-create the diskette using another pre-operating diskette.

X065 The data contained in SYSLCF is inconsistent and cannot be processed. The startup process cannot proceed. Re-create the configuration using the LCF option on the installation diskette.

X066 LCF startup processing has detected a procedure being performed out of sequence. The startup process cannot proceed. Re-create the diskette using the installation diskette LCF procedure. If the error still occurs, open the cover and press the Interrupt key. Follow D1xx procedure.

X070 Application program section length is odd or zero. Startup process cannot proceed. Reassemble the application program, and create a new diskette. If the error occurs with the new diskette, refer to the *IBM 4700 Host Support Problem Diagnosis and Logic* for further action.

X071 The application-program length is less than the sum of its parts. Reassemble the application program, and re-create the diskette. If the error occurs with the new diskette, refer to the *IBM 4700 Host Support Problem Diagnosis and Logic* for further action.

X072 Total application program length is greater than the sum of its parts. Reassemble the application program, and re-create the diskette. If the error occurs with the new diskette, refer to the *IBM 4700 Host Support Problem Diagnosis and Logic* for further action.

X073 End of the extent, but not the application program. Reassemble the application program, and re-create the diskette. If the error occurs with the new diskette, refer to the *IBM 4700 Host Support Problem Diagnosis and Logic* for further action.

X074 The application program data set is invalid. Verify that the diskette you are using is the correct diskette. If it is the correct diskette, create a new operating diskette. If you get this message again, inform your service personnel.

X075 There is an overlay directory length error. Reassemble the application program, and re-create the diskette. If the error occurs with the new diskette, refer to the *IBM 4700 Host Support Problem Diagnosis and Logic* for further action.

X076 There is an overlay directory format error. Reassemble the application program, and re-create the diskette. If the error occurs with the new diskette, refer to the *IBM 4700 Host Support Problem Diagnosis and Logic* for further action.

X077 Application program name in the index does not match the name in the data set. Verify that the diskette you are using is the correct diskette. If it is the correct diskette, create a new operating diskette. If you get this message again, inform your service personnel.

X078 An optional module contains invalid information. Verify that the correct operating diskette has been inserted. If you are using the correct diskette, create a new operating diskette. If the error still occurs, open the cover and press the Interrupt key to cause a dump. Follow the action in the Dlxx message.

X07A The application-program level is below the current release of the controller microcode. The sequence number of the failing application program in your application program data set is identified by the I5nn message.

Reassemble the application program at the correct level and create a new diskette. If the problem still occurs, refer to the *IBM 4700 Host Support Problem Diagnosis and Logic.*

X07B The AP header specified that this AP is written with the SPLIT = Y option set and the CPGEN designates it as transient. This combination is invalid. The sequence number of failing AP in user AP data set is identified by the I5nn message displayed just before this message. Reassemble the AP with SPLIT = N, or change the CPGEN to designate the AP as resident.

X081 The CPGEN storage definition is invalid. The startup process cannot proceed. Re-create the operating diskette with VTAM traces active. Load the diskette. If it fails to load, refer to the *IBM 4700 Host Support Problem Diagnosis and Logic* for further action.

X082 The diskette might not be loaded or it has been modified. Verify that the correct operating diskette was inserted. If it is the correct diskette, re-create the operating diskette. If the error occurs again, open the cover and press the Interrupt key to cause a dump. Follow the D1xx procedure.

X083 to X087 Errors occurred in the configuration or in the microcode. Inform your service representative.

X089 There is insufficient controller storage for the minimum logging area. Decrease the size of CPGEN or the application programs, selectively load optional modules, or add storage to the controller.

X08A The configuration being loaded has requirements that exceed maximum controller resources. Use prompt mode for optional modules (reply 5 to message 00001) to reduce the set of optional modules. If this is not possible, rerun the CPGEN to request fewer optional modules; re-create the diskette.

X093 The system cannot allocate an application program table. This is not necessarily due to a shortage of storage. Decrease some combination of the following: number of stations, number of segment headers for a station, size of SMS extensions, or size of APCALL stack areas.

X0A0 There is not enough storage for the allocation of the segments. Change the CPGEN or the application program, and re-create the diskette or add more storage to the controller.

X0A1 A work station has been allocated too few segment headers. Re-specify the number of segments or the number of segment headers required and re-create the diskette.

X0A2 The segment 13 definitions specified by the application and CPGEN do not agree. Re-specify the shared segment 13 with the CPGEN or the application program, and re-create the diskette.

X0A3 The stations specified by SEGINIT/INITSEG cannot be found. Re-specify the SEGINIT CPGEN macro or the INITSEG instruction with the proper station ID, or specify a station for the segment. Reassemble the CPGEN and the application program, then re-create the operating diskette.

X0A4 The station specified by INITSEG does not have the specified segment. Re-specify the SEGINIT CPGEN macro or the INITSEG instruction with the proper segment number, or specify a segment for the station. Reassemble the CPGEN and the application program, then re-create the operating diskette.

X0A5 SEGINIT/INITSEG segment offset plus the data length exceeds the size of the segment. Re-specify the SEGINIT CPGEN macro or the INITSEG instruction with the proper offset into the segment, or decrease the data length. Reassemble the CPGEN and the application program, then re-create the operating diskette.

X0A6 A diskette read error occurred on SEGINIT. Re-create the diskette and retry the operation.

X0A7 The CPGEN extent ended prematurely. The SEGINIT or INITSEG data indicates there are more initialization entries, but the CPGEN extent has ended. Re-create the diskette and retry the operation. If this fails to correct the problem, refer to the *IBM 4700 Host Support Diagnosis and Logic* for further action.

X0A8 A diskette read error occurred on INITSEG. Re-create the diskette and retry the operation.

Chapter 8. Messages Displayed on Your Terminal

The system monitor displays these messages at your terminal.

MESSAGE MEANING

00001 Enter one of these codes to specify the type of startup. If you do not respond in the time-out period, the controller performs an automatic startup.

1 Cold start (system log and temporary files discarded) with host-communication link activated.

2 Warm start (system log and temporary files retained) with host-communication link activated.

4 Prompt mode for system variables.

5 Prompt mode for optional modules.

6 xx Change the model number associated with the control operator display station.

8 Cold start with no host link (no start link command).

9 Warm start with no host link (no start link command).

B yyyy mm dd hh mn ss Set date and time-of-day clock. Standard default separators are provided.

C x ttttt Type of CNM/CS processing; available only on the expanded and CNM/CS system monitor

Note: If you do not press a key within the 2-minute time-out (20 seconds on the diagnostic diskette) the system performs an automatic startup. When entering your response, use the universal translation table keyboard locations.

00002 Enter a 1- or 2-character control unit address (CUA) in hexadecimal. This CUA writes this CUA on the diskette.

Note: If the CUA switches are all off, the diskette address is used. Otherwise, the CUA switches override the diskette address.

Press the Reset key twice to leave prompt mode and to return to the 00001 message. Press the Enter key to continue without changing the CUA.

00003 Enter a 1 to request dump option; enter 0 to refuse the dump option.

Note: The dump is made only when there is a system failure. If the dump option is in effect, you must reload the system after an error.

Press the Reset key twice to leave prompt mode and to return to the 00001 message. Press the Enter key to continue without changing this value in the system.

00004 Enter a 1- to 16-character control-operator identification. Construct your ID from the characters: 0-9, A-F, X, and blank. After you enter the new control-operator ID, the system displays the 90000 message. Enter the current control-operator ID.

Press the Reset key twice to leave prompt mode and to return to the 00001 message. Press the Enter key if you don't have an entry for the item requested.

00005 Load optional modules. (This is not required with the installation and diagnostic diskettes, where all optional modules on the diskette are loaded.) Enter up to 50 module identification code (IDs); press the Enter key after each ID. The system loads only the optional modules that you specify here. The system then displays all the module IDs you have entered. After you enter the last module ID, press the Enter key again to complete the sequence.

Enter 00 to load *no* optional modules.

Enter FF to load all optional modules that were included in the configuration.

The optional module IDs are:

01 Change loops.

04 3279 color convergence, enter 95, 98, A8. (See Note 1.)

06 Multiblock diskette I/O. (See Note 2.)

0B PIN pad.

0C 3278/3279 MSR, enter 98, 95, A8. (See Notes 1 and 2.)

11 HPCA link diagnostics, enter 67 too. (See Notes 1 and 2.)

13 Change DCA port, enter 95 also — only on diagnostic diskette. (See Note 1.)

15 UDP TGU address processor. (See Note 1.)

18 LUASSIGN optional function. (See Notes 1 and 2.)

1B EDAM (Extended Diskette Access Method).

1D EDAM Allocate/Deallocate, enter 1B also.

1E EDAM Temporary File Processor, enter 1B too. (See Note 1.)

21 Translate instruction, LTRT. (See Notes 1 and 2.)

24 Binary table lookup instruction, LSEEKP. (See Notes 1 and 2.)

25 Extended statistical counter instruction, STATS. (See Note 1.)

26 Decompress/Decompact instructions, DECOMP and DECOMPTB. (See Notes 1 and 2.)

27 Compress/Compact instructions, COMP and COMPTB. (See Notes 1 and 2.)

28 Security instructions, ENCIPHER, DECIPHER, and KEYGEN. (See Note 1.)

2A Scratch pad instruction (SCRPAD). (See Notes 1 and 2.)

2C Interval timer instruction, INTMR. (See Notes 1 and 2.)

30 LDIAG. (See Note 1.)

31 Zoned decimal instructions. (See Notes 1 and 2.)

32 Read data and time of day in variable format, LTIMEV. (See Note 2.)

34 Binary character conversion instruction, LCONVERT. (See Notes 1 and 2.)

40 Format diskette instruction, FORMDKT. (See Note 2.)

41 Compress diskette instruction, COMPDKT. (See Note 2.)

57 DES data encryption instructions, ENCODE/DECODE.

5C Data sequencing instructions, LSORT/LMERGE. (See Notes 1 and 2.)

5E SETDSKT instruction.

68 Enhanced instruction set, SCALE, SETX, SETXREC, TESTX. (See Note 2.)

6A Diagnostic diskette link wrap tests. (See Note 1.)

70 DATSM instructions, PUTFLD, GETFLD, PUTDMS, GETDMS. (See Note 1.)

83 3610, 3611, or 3612 printer. (See Note 1.)

86 4704-1, 3604 magnetic stripe encoder. (See Note 1.)

87 3614/3624 terminal. (See Note 1.)

88 3606 or 3608 keyboard, display, and magnetic stripe reader. (See Note 1.)

89 3608 printer. (See Note 1.)

8A 3615 Printer. (See Note 1.)

93 4710/4720/3616/3262/3287 printer. (See Note 1.)

95 DCA device. (See Note 1.)

98 3278/3279 terminal, enter 95 and A8 also. (See Note 1.)

99 3262/3287/5210 printer, enter 93 and 95 also. (See Note 1.)

A7 3279 (7 color), 3278/3279 terminal, enter 95, 98, and A8.

A8 3278/3279 display translation, enter 95 and 98 also. (See Note 1.)

AB 4704-2 local key tracking, PIN, MSR, MSRE; enter 95, 98, and A8 also. (See Note 1.)

Notes:

1. These optional modules are not available on the installation diskette.

2. These optional modules are not available on the diagnostic diskette.

Press the Reset key twice to leave prompt mode and return to the 00001 message. Press the Enter key if you don't have an entry for the item requested.

00006 Enter a 5-character node ID (in hexadecimal) or press the Enter key to accept the currently assigned XID. This XID replaces the XID on the diskette.

Press the Reset key twice to leave prompt mode and to return to the 00001 message. Press the Enter key if you don't have an entry for the item requested.

00010 You must respond with a one-digit code indicating whether the host or controller begins a session to create a diskette, or if the current diskette creation should be stopped and normal operation resumed.

If the controller initiates the session, you can create a complete diskette only. The host program must initiate partial replacements. Key and enter:

0 Host initiates session.
1 Controller initiates session (with the Host Transmit Facility only).
2 Cancel diskette creation; return to normal monitor mode.

If you enter 1, Host Transmission Facility (HTF) must be active in the host.

00011 Enter 0 for the primary drive, or 1 for the secondary drive.

00012 xxxxxx s

Remove the current diskette and insert a formatted diskette. The *xxxxxx* is displayed only when the Host Transmit Facility creates the diskette. The *s* is displayed when the diskette is created by the Host Diskette Image Create program. Remove the diskette from the drive that you specified on the previous 011 command. Insert the requested diskette. If *xxxxxx* is displayed, insert the diskette with VOLID = *xxxxxx*. If *s* is 1, insert a one-sided diskette. If *s* is 2, insert a two-sided diskette. No operator keyboard response is required. About 10 seconds after you insert the diskette, the diskette drive should move to track 0 to read the volume ID. If there is no diskette drive activity after 15 seconds, open and close the diskette drive door.

00013 xxxxxx

This message means either:

- The diskette inserted in the diskette drive is not appropriate for the request in message 00012, or
- The diskette satisfies the request to load ANY formatted diskette. The VOLID of the inserted diskette is *xxxxxx*.

Do one of the following:

- Enter 0 to indicate that the create diskette function should continue, whether or not the VOLID is appropriate for message 00012.

- Enter 1 to tell the system to read the diskette VOLID again. Then load a diskette that is appropriate for message 00012 or load any diskette you want to use.

00014 The create-diskette operation is complete.

00015 Insert a diskette for a partial replacement on a diskette. Remove the diskette and insert the requested diskette in the drive that you selected.

Because only portions of the diskette are replaced, it is important that you insert the diskette with the VOLID requested.

00016 Insert the diskette from which you loaded the system in the primary drive. If you have already mounted the diskette, open and then close the diskette drive door.

00018 data set name
The named data set was copied successfully by the 955 command.

00017 data set name (**)
The named data set has been added or replaced on the diskette. If ** follows the data set name, the data add or replace was not successful.

00019 data set name
The named data set was successfully copied from a diskette to a disk during the loading of the system.

00020 The control operator pressed the Reset key twice. Key and enter one of the following:

0 To continue at the point interrupted.
1 To tell the host to end the session. When the host ends the session, the 00010 message is displayed.
2 To end the current session, leave diskette create mode and return to normal monitor mode.

00024 Diskette creation completed with no data from the host. Check the host console logs. The controller load image to be transmitted was not on the VSAM data base, or the data base was not available.

00030 xxxxxx

The *xxxxxx* is the volume ID currently on the diskette. If there is no *xxxxxx*, the VOLID is blank.

Do either of the following:

- Enter a 0 to retain the *xxxxxx* as the VOLID to be rewritten on the diskette.
- Enter 1, followed by a space and a 1- to 6-character volume ID to be written on the diskette.

00032 xx The *xx* is the control unit address (CUA) currently on the diskette.

Do either of the following:

- Enter 0 to retain the control unit address, *xx*, currently on the diskette.
- Enter 1 followed by a space and one or two hexadecimal characters to change the CUA on the diskette.

00034 You can retain the current operator ID, or change the operator ID on this diskette.

Do either of the following:

- Enter 0 to retain the current operator ID.
- Enter 1 followed by a space and a new control operator ID to be written on the diskette. The ID consists of 1 to 16 characters. Valid characters are 0-9, A-F, X, and blank. After you enter the new ID, the 90000 message is displayed. Enter the ID just transmitted from the host. If you enter the wrong ID, the new code does not take effect; the system displays the 00034 message again. If an ID code was not transmitted from the host, use blanks, and press the Enter key in response to the 90000 message.

00038 xxxxxx

Verify the XID associated with this diskette; *xxxxxx* is the current XID.

Do either of the following:

- To accept the currently assigned XID, enter 0.
- To change the XID, enter 1 followed by a space and the new XID.

00050 This message is a response to an attention from the keyboard, or to a reply other than the Enter key after a 00053 message. It also acts as a prompt after the volume IDs (VOLIDs) are displayed by a 166 command or after a diskette read error during a 188 command.

Reply with either 0 to continue, or a 1 to cancel the request.

00053 This is a request for you to insert the diskette during a copy operation.

Insert the diskette being copied in the primary drive, and insert the diskette being written in the secondary drive. Press the Enter key after inserting both diskettes. Any other reply causes a 00050 message.

00054 This message requests you to specify the data to be copied during a diskette-copy function (188 command).

To copy the entire diskette from track 0 to track 74, enter 0 or just press the Enter key.

To copy a selected portion of the diskette, enter the beginning address *(ttrr)* and the ending address *(ttrr)* in this format:

(X)ttrr (X)ttrr

Use the hexadecimal form (precede the *ttrr* with the character X) to select side 1.

00055 This message requests you to enter the starting track and record on receiving diskette during a copy (188 command) operation.

To start the copy on the same track and record as the start of the source information on the original diskette, press the Enter key. To start the copy at a specific track *(tt)* and record *(rr)*, specify them in the following format:

(X)ttrr

Use the hexadecimal form (X) if side 1 is required.

00056 This message requests you to insert a diskette in the drive specified in the 166 command. Insert the requested diskette. If the diskette is already inserted, open and close the secondary diskette door.

00059 This message indicates that the diskette copy function completed successfully (188 command), or that you ended the command request with a 1 response to a 00050 message. Note that this message can appear when you change the volume ID (166 command) or when you copy the data set (955 command).

00060 Issued during the 955 command, this message requests the names or numbers of the data sets to be copied.

00064 Requests you to insert a diskette in the diskette drive.

Insert the diskette with the temporary file block to be reconstructed according to the command used:

- On the primary drive for a 937 command.
- On the secondary drive for a 237 command.

00070 The diskette to be formatted has data on it. If you want the formatting to continue, press the Enter key. If you do not want to format this diskette, press any data key and the Enter key.

00071 Formatting is complete. Replace the operating diskette if it was removed. If the operating diskette was formatted, re-IPL with the correct operating diskette.

00073 Compression is complete. Replace the operating diskette if it was removed. If the operating diskette was compressed, re-IPL with the correct operating diskette.

00090 You pressed Reset twice during an 888 command request. Enter 0 to continue transmission or 1 to end transmission.

00091 This message requests you to insert the diskette to be transmitted.

00092 This message requests you to enter the identifiers for the data to be transmitted.

- Press Enter to transmit a controller dump. (You mounted a dump diskette in the previous step.)
- Enter 0 to transmit the entire diskette.
- Enter the characters CF followed by a data set name to transmit a data set from the Local Configuration Facility library.
- Enter AD to transmit an application program dump.
- Enter the track and record addresses that begin and end the part of the diskette you want to transmit.

00093 This message requests you to indicate whether the host or the controller will start the transit session.

- Enter 1 for the controller to initiate the session.
- Enter 0 for the host user program to initiate the session.

00099 Transmission is complete. All other system commands are now valid.

00100 x Debugging mode has transferred to work station *x*.

00112 xxxxxx
This message is used only during automatic backup, and asks you to remove the current diskette and insert a formatted diskette in the secondary drive. If *xxxxxx* is displayed, insert the diskette with VOLID = *xxxxxx*. No operator keyboard response is required. About 10 seconds after you insert the diskette, the diskette drive should move to track 0 to read the volume ID. If there is no diskette drive activity after 15 seconds, open and close the diskette drive door.

00153 This is a request for you to insert the diskette during a copy operation while automatic backup is in effect.

Insert the diskette being written in the primary drive. This diskette must be the same type (1, 2, 2D) as the diskette in the secondary drive.

10075 Enter either an encrypted key, or the first part of a two-part key. Enter the key as a single string of 16 hexadecimal alphameric characters.

10076 You are being prompted to enter the second part of a two-part key.

10077 xxxx The *xxxx* is the verification code that is returned by the 330 command with the 1, 2, or 3 operand.

The verification operation is complete. Compare the verification code with the expected verification code.

3262C 3262 DCA printer test complete.

3262S You started the 3262 DCA printer test.

3278C 3278 DCA display test complete.

3278S You started the 3278 DCA display test.

3287C 3287 DCA printer test complete.

3287S You started the 3287 DCA printer test.

3604C 3604 display test is complete.

3604S You started the 3604 display test.

4704C 4704 display test is complete.

4704S You started the 4704 display test.

4710C 4710 printer test is complete.

4710S You started the 4710 printer test.

4720C 4720 printer test is complete.

4720S You started the 4720 printer test.

90000 This message requests you to enter the identification code during the logon procedure. It also appears during startup or create-diskette prompting if you attempt to change the identification code.

If you enter the ID correctly, the system displays the 91111 message. If you make a keying error, the system displays the 91005 message. If the system does not accept the ID, the system displays the 00004 or 00034 message.

90001 You entered an invalid command (not numeric, more or less than three characters, or not in the command code table), or you entered the 032 command to write to the operating diskette. Enter the command correctly.

90002 xxxx An error occurred while the system was reading from the diskette. The status code appears immediately after the error message. The *xxxx* is the status. For some read operations, the system also displays *ttrr* address and diskette drive in error following the status; 1 indicates the error was on the primary drive, 2 indicates the error was on the secondary drive.

a. If the diskette stopped, start the diskette (042 0).

b. Insert the correct diskette, and close the diskette drive door.

c. If an invalid record was requested (status 0480), enter the command again with the correct track and record number.

d. If the read operation was for a control record (status 0204), the requested record is not there.

e. If the temporary file was reset with the set diskette option, no action is required.

90003 An error was detected while the system was reading from the keyboard. If the error continues for two more keyboard entries, the control operator is automatically logged off. Also, this message appears when you press the Reset key twice; status is 0800. Reenter the command.

90005 The system detected an error while the system was writing to the display.

Action: If this is a recurring error, log off the display station via the 000 command. If this does not work, press the Reset key six times. When the logoff is complete, log on using another display station, and test the display causing the error.

90006 You entered invalid data when keying the log message number in a 002 or 046 command, or when keying the track and record number of a read diskette command. Enter the command correctly.

90007 A command has been given to print on the output printer and one of the following conditions was detected: (1) no output printer has been assigned, (2) an error occurred while printing, or (3) the optional module was not loaded for the assigned terminal.

a. Assign an output printer with the 005 or 006 command if none was assigned.

b. Determine the meaning of the status bytes.

c. Try the operation again; if the error still occurs, reassign the printer with the 005 or 006 command and then assign it as the test component with the 007 command. Test the printer; if errors still occur, inform your service representative.

d. The wrong diskette was inserted for this controller, or the optional module for this terminal was not included at startup. Insert the correct diskette for this controller or request that the proper optional modules be loaded at startup.

90008 You issued a command to a test component, and one of the following conditions was detected: (1) no test component has been assigned, (2) an error was detected while using the test component, or (3) the optional module was not loaded for this component.

a. Assign a test component if none was assigned with the 007 command.

b. Determine the meaning of the status bytes connected with the test component. If the errors continue, inform your service representative.

c. The wrong diskette was inserted for this controller, or the optional module for this terminal was not included at startup. Insert the correct diskette for this controller or request that the proper optional module be loaded at startup.

90009 An operator keyboard error occurred; an invalid component address, station ID, or logical device or LU address was keyed in, or the component is not included in the configuration.

Enter the command correctly or verify that the component is defined in the configuration.

90010 You entered an invalid field for test repetition. The valid repeated values range from 0 to 999.

90011 Either you entered the 069 command incorrectly, or did not assign a test component.

90012 xxxx An error occurred while the system was writing to the diskette. For some write operations the address and diskette drive are also displayed following the status. The *xxxx* is the status; 1 indicates the error was on the primary drive, 2 indicates the error was on the secondary drive.

Inform your system programmer.

90013 You have entered an invalid work-station identification (ID) with a 123 command, or the work station was not included in the configuration procedure.

90014 An error was detected while the system was attempting to load overlays within the system monitor.

Do one of the following:

a. Ensure that the diskette drive door is closed.

b. If the diskette is logically stopped, ensure that the correct diskette is inserted, and then enter the start diskette command (042 0).

c. Again enter the command that caused the error.

d. If the error occurs again, try a new operating diskette or try an installation diskette.

90015 The component requested as the output printer or test component (command 005, 006, 007, or 008) is busy and cannot be assigned, or the device cluster adapter (DCA) port should execute a 074 request first before using one of the commands, or the LUASSIGN function is invalid.

90016 You entered an invalid command field.

90017 You entered an invalid line length or a line length greater than that permitted for the test.

90018 No component has been assigned for the requested function. Assign a terminal component capable of performing the function requested.

90019

- You attempted to assign the presently assigned test component as the output printer, or
- You attempted to assign the presently assigned output printer as the test component, or
- You attempted to assign an LU address to, or components to or from, your control-operator terminal with the 008 command, or
- You attempted to assign your terminal as test component.

Use another component for the function. If the component is correct, reassign it first with the 005 or 006 and 007 commands, and then assign it again with the 007 and 005 or 006 commands or the 008 command.

90020 You entered 051 or 052, but did not assign a 3624 as the test component. Assign the appropriate component and reenter the command.

90021 An error was detected when reading from the test component. Refer to the status bytes to identify the problem.

90022 The echo message from the automated teller machine did not match the output test message.

The format of the 90022 error message is:

```
90022 xxx s. . . .se. . .e
xxx = counter
s. . . .s = data sent
e. . . .e = echo data
```

Compare the displayed data sent with the displayed echo data. Try the test again. If the error persists, call the service representative.

90023 You changed the diskette and entered 042 0 to start the diskette; or, if the diskette was inserted in response to the 00016 message, the diskette was not the diskette used to load the system.

Either reload the original diskette and retry the operation, leave the second diskette inserted and start up again (reset), or insert the IPL diskette.

90024 Unexpected data was received from the 3624.

The format of the 90024 message is:

90024 - d. . . .d
d. . . .d = unexpected data received from the 3624

Refer to the *IBM 3624 Operator's Guide* for the meaning of the data received.

90025 Unable to reassign test component or output printer to the original owner.

Start up again (reset). If the error recurs, notify the service representative.

90026 Unable to assign an LU address or a component because the logical device address (LDA) requested already has a component or LU address assigned.

If you have issued 005, or 006 and 007, start the system again. If the error recurs, notify the service representative. If the error resulted from a 008 command, remove the component or LU address that is assigned to the LDA before assigning the new component or LU address.

90030 Either:

- A command was issued that required an optional module that was not loaded or was not available on the loaded diskette
- The command requires disk drive 2 or the auxiliary diskette drive, which is not present.

Insert the correct disk or diskette or start up the system again (press the Reset switch on the controller) and enter a 5 in response to the 00001 message to specify the required optional module ID.

90032 A read or write error occurred during programmable input mode, or during operation of a 029 command. Inform your programming personnel.

90033 Another work station is logged on in programmable input mode.

90034 Link adapter wrap test failed or the host link adapter was not stopped (status = 8000). Ensure that the host link adapter is stopped and try the command again. If the error still occurs, inform the service representative.

90035 One of the following occurred:

- you entered an invalid 040 command.
- A 973 command failed because of invalid loop number or device type.
- A 074 device cluster adapter (DCA) or port control command failed.

Take action according to the status returned. If no status was returned, enter the command correctly.

90036 Invalid extended statistical counter or device ID entered with the 067, 068, or 072 command; or end of list reached when displaying counters with the 072 command (code 1),

90047 The diskettes used during a copy operation are not the same type or format, or an invalid diskette was inserted for a copy operation.

90048 The HEADER01 record could not be found, the record after HEADER01 is not a Host Diskette Image Create (HDIC) header. Notify your host programming personnel.

90050 The response of data entered from the keyboard during startup is not valid. Enter the correct response to the 00001 message that appears on the next line.

90051 Either you did not load the correct diskette, or requested the wrong link module.

90052 If the status bytes are zero, an invalid data set ID was entered in a 065 command; a nonzero status indicates the LDKT instruction failed. Verify the data set ID and reissue the 065 command.

90053 A valid diskette was not inserted in the auxiliary drive, one or both diskettes were not 256-byte format, or the starting or ending track on the primary drive diskette was zero. Correct the diskette problem, and retry the operation.

90061 aaaa bbbb cc ddddzzz...z

Invalid or unexpected response or input was received from the host. Transmission ends.

aaaa = Device status bytes

bbbb = Host read control field

cc = Communication link status byte

dddd = Host write control field

zzz = First 20 bytes of data in input/output segment

All characters are hexadecimal.

90062 aaaa bbbb cc dddd zzz...z

An error was detected during a read operation to the host. Transmission ends.

aaaa = Device status bytes

bbbb = Host read control field

cc = Communication link status byte

dddd = Host write control field

zzz = First 20 bytes of data in input/output segment

All characters are hexadecimal.

90063 aaaa bbbb cc dddd zzz...z

Contact with the host was lost during a read or write operation. Transmission ends.

aaaa = Device status bytes

bb = Host read control field

cc = Communication link status byte

dddd = Host write control field

zzz = First 20 bytes of data in input/output segment

All characters are hexadecimal.

90064 aaaa bbbb cc dddd zzz...z

An error was detected during a write operation to the host. Transmission ends.

aaaa = Device status bytes

bbbb = Host read control field

cc = Communication link status byte

dddd = Host write control field

zzz = First 20 bytes of data in input/output segment

All characters are hexadecimal.

90066 aaaa bbbb cc dddd zzz...z

The controller could not establish a session with the host. Transmission ends.

aaaa = Device status bytes

bbbb = Host read control field

cc = Communication link status byte

dddd = Host write control field

zzz = First 20 bytes of data in input/output segment

All characters are hexadecimal.

90070 aaaa bbbb cc dddd zzz...z

This message appears only when the host is transmitting a diskette image processed by the Host Diskette Image Create (HDIC) service program. The message indicates one of the following:

a. The header record (produced by HDIC) preceding the diskette image is less than 7 bytes long.

b. All other records (except 0588) are not 256 bytes long.

c. The starting *ttrr*, specified in the header record, is invalid.

aaaa = Device status bytes

bbbb = Host read control field

cc = Communication link status byte

dddd = Host write control field

zzz = First 20 bytes of data in input/ output segment

All characters are in hexadecimal.

Diskette creation ends.

90071 aaaa bbbb ccc dddd zzz...z

Invalid or unexpected response or input received from the host. Diskette creation ends.

aaaa = Device status bytes

bbbb = Host read control field

cc = Communication link status byte

dddd = Host write control field

zzz = First 20 bytes of data in input/ output segment

All characters are hexadecimal.

90072 aaaa bbbb cc dddd zzz...z

An error was detected during a read operation to the host. Diskette creation ends.

aaaa = Device status bytes

bbbb = Host read control field

cc = Communication link status byte

dddd = Host write control field

zzz = First 20 bytes of data in input/ output segment

All characters are in hexadecimal.

90073 aaaa bbbb cc dddd zzz...z

Contact with the host was lost during a read or write operation. Diskette creation ends.

aaaa = Device status bytes

bbbb = Host read control field

cc = Communication link status byte

dddd = Host write control field

zzz = First 20 bytes of data in input/ output segment

All characters are in hexadecimal.

90074 aaaa bbbb cc dddd zzz...z

An error was detected during a write operation to the host. Diskette creation ends.

aaaa = Device status bytes

bbbb = Host read control field

cc = Communication link status byte

dddd = Host write control field

zzz = First 20 bytes of data in input/ output segment

All characters are in hexadecimal.

90076 aaaa bbbb cc dddd zzz...z

The controller was unable to establish a session with the host. Diskette creation ends.

aaaa = Device status bytes

bbbb = Host read control field

cc = Communication link status byte

dddd = Host write control field

zzz = First 20 bytes of data in input/ output segment

All characters are in hexadecimal.

90077 aaaa bbbb cc dddd zzz...z

This message means either:

a. The diskette you inserted in the diskette drive is not formatted correctly, or

b. The diskette is an invalid operating diskette on which you are attempting to add or replace an application program.

This message is followed by a 00012 or 00015 message.

To display the next 64 bytes (40 to 7F), press the Enter key. Location 75 (4B) should contain an F1 which indicates this is a properly formatted diskette. Anything other than an F1 causes a 90077 error message, and the diskette cannot be used for the create diskette function.

aaaa = Device status bytes
bbbb = Host read control field
cc = Communication link status byte
dddd = Host write control field
zzz...z = First 20 bytes of data in input/output segment

All characters are in hexadecimal.

90078 aaaa bbbb cc dddd zzz...z
The controller could not load a required portion of the create diskette function. Diskette creation ends.

aaaa = Device status bytes
bbbb = Host read control field
cc = Communication link status byte
dddd = Host write control field
zzz...z = First 20 bytes of data in input/output segment

All characters are in hexadecimal.

90079 aaaa bbbb cc dddd zzz...z
You inserted the wrong type diskette in response to a 00012 message.

aaaa = Device status bytes
bbbb = Host read control field
cc = Communication link status byte
dddd = Host write control field
zzz...z = First 20 bytes of data in input/output segment

All characters are in hexadecimal

90080 An invalid debugging command was entered or one of the input fields was not valid.

90081 An error was detected while the system was writing to the display during debugging mode. Enter the command again. If the error persists, log off and try another display station.

90082 One of the following:

- The operator keyed an invalid segment number or an invalid address of the first byte to be displayed or changed.

- The end of the segment was reached during a segment display.

90084 One of the following:

- The stop table is full when you issue the 03 command.
- You entered an invalid address on a 04 command.
- You entered an invalid direct address on an 11 or 12 command

Enter the command again with the correct information. If there is no error on the 03 command, the stop table is full. You must remove a stop with a 04 command before entering a new stop with the 03 command.

90086 You tried to change the area of storage containing application programs and the configuration data on an operating diskette.

Make these changes at the host, and create a new operating diskette; or use a non-operating diskette.

90088 One of the following:

- You entered an invalid work station identification on a 06 command.
- There is no A/B operator on the work station being debugged.

90089 Scan pattern on debug request not found.

90090 Mount a formatted diskette for the unattended creation of an operating diskette.

90091 The unattended creation of an operating diskette is in progress. DO NOT enter any system monitor commands.

90092 You issued the 320 command and the controller found a problem with the 3600-level of encryption (ENCODE/DECODE). Follow the problem reporting procedures described in *4701 Controller Operating Instructions.*

90093 You issued the 320 command and the controller found a problem with the 4700-level of encryption (ENCIPHER/DECIPHER).

First ensure that a master key has been loaded into the controller.

If the encryption keylock is activated and the controller contains a master key, the 90093 message means that the controller is malfunctioning. Follow the problem reporting procedures described in *4701 Controller Operating Instructions.*

90094 The controller's cryptographic facilities are not working properly because of an internal machine failure. Follow the problem reporting procedures described in *4701 Controller Operating Instructions.*

9009[illegible] You requested a utilization report but the timer was not started or has wrapped.

90096 The address you requested when printing or displaying the dump diskette is not available on the diskette you are using.

90097 The next screen of data was requested, but there is no data to display.

90098 No available space on the diskette to perform the copy.

90099 xx yyyy nnnnnnnn
An error was detected in the system monitor or the CNM/CS function of the controller. When this message appears you might be logged off the system monitor.

The *xx,yyyy,* and *nnnnnnnn* are diagnostic data for service personnel.

91001 A 711 command was issued, but CNM/CS processing is disabled. Enter a 712 command from the CNM/CS workstation.

91002 No operator is logged on at the remote controller.

91003 The remote host operator is attempting to log on the system monitor through CNM/CS. Notify the remote operator of system monitor availability with a 711 command, or follow your local procedure.

91004 The remote controller has an operator logged on the system monitor. The controller operator receives a message of your logon attempt, and should follow the procedure established at the controller site.

91005 The operator attempted to log on the system monitor with an invalid control operator ID.

91006 The controller accepted the logon from the remote operator.

91007 This message acknowledges a command from the remote operator to the system monitor.

91010 The accompanying message is from the remote operator.

91013 You entered an interval time (014) command containing an invalid function code.

91014 You requested interval timer data for an invalid work station or timer.

91077 A problem occurred that prevents the controller from loading, verifying, or erasing the cryptographic key.

If this message appeared after you received a 10075 or 10076 prompting message, you may have entered a key in the wrong format.

If this message appeared after you issued the 330 command, the problem could be one of the following:

- You forgot to activate the encryption keylock. Insert the encryption key in its keylock, turn the key *counterclockwise,* and reissue the 330 command.
- Your institution did not include the necessary module of controller data in the operating diskette. The P28 module is required to support the cryptographic facilities, and must be included during the configuration procedure (OPTMOD configuration instruction), or during the startup procedure (enter P28 in response to the 00005 startup message).
- The controller is malfunctioning. Follow the problem reporting procedures described in *4701 Controller Operating Instructions.*

91111 x yy The control operator identification is correct. Logon is complete.

loop number of 0 for DCA devices.
yy = the address or port number for DCA devices.

92010 A diskette read error occurred during startup; the system is unable to continue processing.

Do the following:

a. Try startup again (press Reset).

b. Try another operating diskette.

c. Try an installation diskette.

If the error still occurs, inform the service representative.

92076 xxyy xxyy...

The Module IDs displayed are:

- IDs that are invalid.
- IDs of modules not available on the loaded diskette.
- IDs that were entered more than once in response to prompt message 00005.

invalid, not available, or duplicate ID.
yy = 00 Module not found on diskette.
01 Module xx requires an optional module that was not loaded.
03 Controller data error.
04 Insufficient base control storage space.
05 System capacity for optional modules exceeded.
07 Configuration assembly error.
08 Configuration assembly error.
09 Configuration assembly error.
0A Configuration assembly error.

Press the Enter key to continue the startup sequence if these modules are not required, or insert the correct diskette, press the Reset switch on the controller, and enter prompt mode during the startup to specify the correct optional module IDs.

Chapter 9. Controller Log Messages

The controller places the following types of messages in the log:

- **1y hhmm 002 aa bbbbbb cccccccc dddddd ee ff gggg.**

y
0 for no alert light; 1 lights the alert light.

hhmm
Time stamp.

a
The control code supplemental version ID.

b
Diskette volume identification (VOLID).

c
Configuration generation identification (GENID).

d
The EC level of the controller data.

e
Control unit address (CUA).

f
Relocation count.

g
Session identification (ID), the number of cold starts since the diskette was created.

- **11 hhmm 003 nnnnnnnn ss pp aaaa cccc iiii**

nnnnnnnn
Application program name.

ss
Work-station identification in hexadecimal.

pp
Program check code in hexadecimal.

aaaa
Program check address in hexadecimal.

cccc
Loop threshold count in hexadecimal.

iiii

First 2 bytes of the instruction. If *pp* = 0B (user's instruction counter invalid), *iiii* is not valid data.

The above information is related to your institution's application program.

- **11 hhmm 004 DSIx dddd dddd ... dddd**

DSI is design support information. The service representative might request this information when correcting a problem.

x

Record containing data bytes.

1 First 64 bytes of data.
2 Last 64 bytes of data.

dddd dddd ... dddd

64 bytes of data.

- **11 hhmm 005 LOOP x ERROR, CODE - y - z**

x

Loop number.

y

Loop status:

0	The controller loop adapter card and the modem (if a remote loop) passed the wrap test.
1	Modem failed wrap test.
2	Controller loop adapter card failed wrap test.
4	Adapter check.
6	Combination of 2 and 4.
8	User requested stop loop.
9	A loop defined in the CPGEN is not started because it is not present, not included in the controller.

z

Modem status (always equal to 3):

- **10 hhmm 006 axxdd**

a

indicates the reason for the log message.

xx

retry or completion status.

dd

signal status.

For more information on this message, refer to the *IBM 4700 Subsystem Operating Procedures,* GC31-2032.

- **11 hhmm 006**

The host link error message has five formats:

- **11 hhmm 006 a**

a = 0
An error condition caused the controller to run wrap tests, and they were successful.

a = 2
A stop link command was issued by the system monitor or your institution's application program.

- **11 hhmm 006 a xxbbxxxxxxxxxxxx**

a = 1
The controller ran wrap tests, but they were not run successfully. Inform the appropriate service representative.

xx
This is the adapter address, 15.

bb
Identifies the test that failed:

09	Adapter wrap test
0D	Modem wrap test, X.21 switched network DCE, or cable wrap switch test
11	Interface select test
21	Adapter reset test
27	X.21 translator card test

xxx...x
This is diagnostic information for the service representative.

- **11 hhmm 006 tttttrrrrrrcc a sssssssstttxxttttttrrrrrrcc**

If *a* = 3, a message was received in error. If *assss* = 38004, the station *(xx)* does not exist. All other occurrences of this message suggest a network error.

ssssssss
Sense returned to the host, if possible, on a bad message.

ttttt
Transmission header of the failing message.

rrrrrr
Request header of the failing message.

cc

First byte of the request unit.

See "Communication Sense Codes," in the *IBM 4700 Controller Programming Library.*

- **11 hhmm 006 a xxyy**

The X.21 host link requires immediate attention. The type of status depends on the settings of the message information.

- **11 hhmm 006 a xx ss...ss**

This message for X.21 switched links defines intermediate or error completion status, depending on the value of *a* as follows:

a = 4

OPEN error completion status, selection sequence error.

a = 5

OPEN intermediate completion status, line identification error.

- **11 hhmm 010 x**

The *x* is the error return code:

X'1'	Link adapter type unknown.
X'2'	Link module requested cannot be found.
X'3'	Default taken, but no link module matches adapter.

- **1y hhmm 014 t dp ccc....ccc**

y

0 for no check light or 1 for check light.

t

Reason code:

1 = DCA shutdown
2 = Port shutdown
3 = Machine check
4 = Adapter timeout
5 = Invalid port ID
6 = No extended status
7 = Stray Op complete
8 = Feature error
9 = Device check
10 = Device/CPGEN mismatch

dp

For port errors, *d* = A, and *p* = port number.

dp

For adapter errors, d = 9, and p = A.

ccc....ccc

Port statistical error counter for port errors, adapter statistical error counters for adapter errors.

- **11 hhmm 017 lsdm TYPE=tt fc s1 s2**

l

Loop number or A if DCA.

s

Terminal address in hexadecimal, or port number if DCA.

d

Component address in hexadecimal.

m

Modulus value for the terminal.

tt

Component type:

01	host link
02	diskette
80	loop control
81	4704/3604/3278/3279 keyboard
82	4704/3604/3278/3279 display
83	3610, 3611, or 3612 printer
84	3262/3287/5210 printer
86	magnetic stripe encoder
87	3614/3624 terminal
88	3606 or 3608 keyboard, display, and magnetic stripe reader
89	3608 printer
8A	3615 printer
92	3616 printer
9A	4710 printer
AB	4704-2/3 magnetic stripe encoder
B0	4720 printer

fc

Failure code:

01	communication failure, S1S2 = I/O status code
02	operational failure, S1S2 = I/O status code
03	propagation delay, S1S2 = I/O status code
08	power-on failure, S1 for 4704 = device code (See Figure 9-1 and Figure 9-2)

s1

When FC = 08: for the 4710 and 4720, the terminal's failing subcomponent; for the 4704, the power-on device code.

S2

For FC = 08: the ROS/EC number.

Terminal Status 4704-1 Power-On Device Codes

Keyboard Test Error

11	Invalid keyboard on primary or secondary port.
12	Keyboard internal test of primary port.
13	Keyboard internal test of secondary port.
14	No keyboard attached to port.
15	No response from keyboard adapter.
16	PIN pad offline test error.
17	Keyboard adapter port check.
18	Keyboard adapter microprocessor check.

Keyboard Echo Test Error.

25	Transmit check.
27	No response from magnetics device.
38	Magnetics adapter self test.

CRT Adapter Test Error.

47	CRT controller check.
48	Refresh RAM check.

IPL Check Error.

81	Sense response time-out.
82	Microcode patch load error.
86	UDC font load or microcode patch failure.
87	Typematic load failure.
88	Sense failure.

Figure 9-1. 4704-1 Device Codes for Message Type 017

Terminal Status 4704-2 Power-On Device Codes	
80	Keyboard test error
40	Magnetic device error
20	Patch load check

Figure 9-2. 4704-2 Device Codes for Message Type 017

- **10 hhmm 018 a yyyyrmmrddrhhrmmrss**

a
Timer set by:

0 Date and time of day set by operator.
1 Date increased by system.
2 Time of day adjusted by operator.

r
Separator

yyyy
Year

mm
Month

dd
Day

hh
Hour

mm
Minute

ss
Second

- **10 hhmm 020 xxx ... xxx**

This log message records a command entered from the control operator's terminal. The *xxx ... xxx* is the command.

- **11 hhmm 021 NVM ERROR ENCIPHER STATUS xxxx**

The *xxxx* is the status returned from an encipher instruction, as contained in SMSDST.

- **1y hhmm 022 dllccffraaaaiiiisseeppppppppp**

y

0 for no check light; 1 lights the check light.

d

Drive number

0	First disk drive
1	Second disk drive

ll

Log flags

Bit	**Meaning**
0	partial log; no address or data CPR
1	Partial log; no adapter internal address
2,3	Partial log; device sense validity flags: 00 All sense bytes read with no errors 10 Current sense mode read with no errors 01 All sense bytes read; errors occurred 11 No sense information
4	Partial log; no SCA director
5	Reserved
6	Partial log; adapter status
7	Partial log; no operational statistics

cc

Retry count

ff

First status flags

Bit	**Meaning**
0-4	Reserved
5	First ARC is an error ARC
6	First ARC is an exception ARC
7	Reserved

rr

First return code (ARC)

aaaa

Adapter status bytes

iiii

Adapter internal status byte

ss

SCA director

ee

Error syndrome

ppppppppp

Failing physical block number (PBN)

- **10 hhmm 023 dhhhhhhhhddd...ddd**

d
Drive

1 First disk drive
2 Second disk drive

hhhhhhhh
Failing cylinder head and sector

ddd...ddd
Device sense bytes

- **10 hhmm 024 dccc...ccc**

d
Drive

1 First disk drive
2 Second disk drive

ccc...ccc
Active device commands

- **10 hhmm 025 dssssssswwwwzzzznnnnmmmmaaaa**

d
Drive

1 First disk drive
2 Second disk drive

ssssssss
Number of sectors read or verified

wwww
Number of sectors written

zzzz
Number of zero-length seek operations

nnnn
Number of non-zero-length seek operations

mmmm
Number of missed revolutions

aaaa
Number of off-track alternates accessed

- **10 hhmm 030 data**

This message is written when you issue the 030 system monitor command to place a message in the log.

data
The text you wrote with the 030 command.

- **01 hhmm 101 token data**

This is a user-written log message for communicating error information to the host-resident NPDA program. Your application program writes this message into the log using an LWRITE instruction. The system passes this message for Alert processing by the host program.

token
A 1- to 8-character error identification (an NPDA token).

data
Up to 240 bytes of user-written error data.

Chapter 10. Statistical Counters

The controller records errors in statistical counters and extended statistical counters. The statistical counters record errors for each device and each component of the system, including all attached devices, as well as the:

- Host communication link
- Diskette drives
- Loop control
- DCA control
- DCA ports

Host Communication Link Statistical Counters

These counters report the status of the host communication link.

Counter 1: First and Ending Sequence Counter (Information)
Counter 2: Test Message Received Counter (Information)
Counter 3: Write Retry Error Counter (Soft Error)
Counter 4: Time-out Error Counter (Hard Error)
Counter 5: Overrun Error Counter (Soft Error)
Counter 6: Underrun Error Counter (Soft Error)
Counter 7: Connection Problem Error Counter (Hard Error)
Counter 8: Invalid Adapter Status Error Counter (Hard Error)
Counter 9: Frame Check Sequence Error Counter (Soft Error)
Counter 10: Abnormal End Error Counter (Soft Error)
Counter 11: Data Communication Equipment Error Counter (Hard Error)
Counter 12: Busy Counter (Information)
Counter 13: Command Reject Error Counter (Protocol Error)
Counter 14: Machine Check Error Counter (Hard Error)
Counter 15: Invalid Data Field (Protocol Error)
Counter 16: Nonsupported Command Received (Soft Error)
Counter 17: Receive line Dropped (Multiuse Loop and World Trade Only)
Counter 18: CTS Transition During Write (Multi-Use Loop and World Trade Only)
Counters 19-24: Reserved

BSC Host Communication Link Statistical Counters

Counter 1: Poll Counter (Information)
Counter 2: Test Request Counter (Information)
Counter 3: Write Retry Counter (Soft Error)
Counter 4: Time-out Error Counter (Hard Error)
Counter 5: Overrun Error Counter (Soft Error)
Counter 7: Connection Problem Error Counter (Hard Error)
Counter 8: Invalid Adapter Status Counter (Hard Error)
Counter 9: Block Check Error Counter (Soft Error)
Counter 10: Primary Abort Counter (Information)
Counter 11: Data Communication Equipment Error Counter (Hard Error)
Counter 12: Secondary Busy Counter (Information)
Counter 13: Sequence Error (Soft Error)
Counter 14: Adapter Check Counter (Hard Error)
Counter 15: Select Counter (Information)

Device Cluster Adapter (DCA) Statistical Counters

DCA Adapter Counter Descriptions

Counter 1: Over-63 Error
Counter 2: Read Time-out
Counter 3: Transmission Turnaround Error or Read Line Parity Error
Counter 4: Read Data Parity Error
Counter 5: Stop Poll Set in Status
Counter 6: Reserved
Counter 7: Entry in Error Queue
Counters 8-11: Reserved
Counter 12: Invalid Port ID in Queue Entry
Counter 13: No Extended Status Bits
Counter 14: DCA Machine Check Status

DCA Port Counter Descriptions

Counter 1: 32nd Poll Retry Occurred
Counter 2: Terminal Shutdown
Counter 3: Poll Time-out
Counter 4: Poll Error
Counter 5: Stray Operation Complete Interrupt
Counter 6: Feature Error
Counter 7: Device Check
Counter 8: Device/CPGEN Mismatch

Diskette Statistical Counters

Counter 1: Intervention required counter
Counter 2: Command reject
Counter 3: No record found
Counter 4: CRC - Cyclic Redundancy Check
Counter 5: Bad diskette format
Counter 6: Diskette hardware malfunction.
Counter 7: Seek failure
Counter 8: Overrun

Loop Control Counters

Counter 1: Syncronization (Loop Outage)
Counter 2: Noise (Slot Damage)
Counter 3: Loop Adapter Check
Counter 4: Adapter Wrap Failure
Counter 5: Reserved
Counter 6: Propagation Delay

Terminal Component Counters

4704-1 Keyboard

Counter 1: Loop Error (Status 0200)
Counter 2: Terminal Loop Adapter Error
(Status = 0201)
Counter 3: Device Errors
Counter 4: Translation Error
Counter 5: Segment Overflow

4704-1 Display

Counter 1: Loop Error (Status 0200)
Counter 2: Terminal Loop Adapter Error
(Status = 0201)

4704-1 Magnetic Stripe Encoder

Counter 1: Loop Error (Status 0200)
Counter 2: Terminal Loop Adapter Error
(Status = 0201)
Counter 3: Device Error

4704-2/3 Keyboard

Counter 1: DCA Error (Status 0200)
Counter 2: Port/Cable/Terminal Error (Status 0201)
Counter 3: Reserved
Counter 4: Translation Check (Status 2000)
Counter 5: Segment Overrun (Status 0101)
Counters 6-15: Reserved

4704-2/3 Display

Counter 1: DCA Error (Status 0200)
Counter 2: Port/Cable/Terminal Error (Status 0201)
Counter 3-4: Reserved
Counter 5: Keyboard POR Test Failure
Counter 6: Magnetic Device POR Failure
Counter 7: Device Patch Error
Counters 8-12: Reserved

4704-2/3 Magnetic Stripe Encoder

Counter 1: DCA Error (Status 0200)
Counter 2: Port/Cable/Terminal Error (Status 0201)
Counter 3: Device Error

3604 Keyboard

Counter 1: Loop Error (Status 0200)
Counter 2: Terminal Loop Adapter Error
(Status = 0201)
Counter 3: Device Errors
Counter 4: Translation Error
Counter 5: Segment Overflow

3604 Display

Counter 1: Loop Error (Status 0200)
Counter 2: Terminal Loop Adapter Error
(Status = 0201)

3604 Magnetic Stripe Encoder

Counter 1: Loop Error (Status 0200)
Counter 2: Terminal Loop Adapter Error
(Status = 0201)
Counter 3: Device Error

3614/3624 Consumer Transaction Facility

Counter 1: Loop Error (Status 0200)
Counter 2: Terminal Loop Adapter Error
(Status = 0201)
Counter 3: Time Out
Counters 4-16: Not Used

4710/4720 Printer

Counter 1: Loop Error (Status 0200)
Counter 2: Terminal Loop Adapter Error
(Status = 0201)
Counter 3: Timeout (Status 0202)
Counter 4: Terminal Protocol Violation (Status 0204)
Counters 5-8: Unused
Counter 9: Power On (Status 0808)
Counter 10: Unused
Counter 11: A Operator Active
Counter 12: B Operator Active
Counter 13: Controller Protocol Violation (Status 0204)
Counter 14: DEVPARM Request Rejected (Status 0480)
Counter 15: Incorrect Message Length (Status 0208)
Counter 16: Inhibit Print Key (Status 4001)
Counter 17: Line Length Exceeded (Status 0101)
Counter 18: Warning Line (Status 4080)
Counter 19: End of Forms (Status 8010)
Counter 20: Print Emitter Check (Status 0202)
Counter 21: Unused
Counter 22: Translation Check (Status 2000)
Counter 23: End of Page (Status 4020)
Counter 24: Unused

3606 and 3608 Terminals

Counter 1: Loop Error (Status 0200)
Counter 2: Terminal Loop Adapter Error
(Status = 0201)
Counter 3: Read Time Out (Status 0202)
Counters 4-9: Unused
Counters 10,11: Total Transactions
Counters 12,13: Transaction Queued
Counters 14,15: Total Transactions Received When Not
Idle
Counter 16: Transactions Queued

3608 Printer

Counter 1: Loop Error (Status 0200)
Counter 2: Terminal Loop Adapter Error
(Status = 0201)
Counter 3: Print Emitter Check (Status 0202)
Counter 4: Incorrect Message Length (Status 0101)
Counter 5: Intervention Required (Status 8000)
Counter 6: Timeout (Status 0202)

3610/3611/3612 Printers

Counter 1: Loop Error (Status 0200)
Counter 2: Terminal Loop Adapter Error
(Status = 0201)
Counter 3: Intervention Required (Status 8000)
Counter 4: Print Emitter Check (Status 0202)
Counter 5: End of Forms (Status 8010)
Counter 6: Platen Open (Status 0202/8202)
Counter 7: Timeout (Status 0202)
Counter 8: Missing Left Margin

3615 Printer

Counter 1: Loop Error (Status 0200)
Counter 2: Terminal Loop Adapter Error
(Status = 0201)
Counter 3: Print Check (Status 0202)
Counter 4: End of Forms (Status 8010)
Counter 5: Printer Not Ready (Status 0202)
Counter 6: Timeout (Status 0202)
Counter 7: Halt on Check (Status 0202)
Counter 8: Forms Emitter Check (Status 0202)

3616 Printer

Counter 1: Loop Error (Status 0200)
Counter 2: Terminal Loop Adapter Error
(Status = 0201)
Counter 3: Timeout (Status 0202)
Counter 4: Terminal Protocol Violation (Status 0204)
Counters 5-8: Unused
Counter 9: Power On (Status 0808)
Counter 10: Unused
Counter 11: A Operator Active
Counter 12: B Operator Active
Counter 13: Controller Protocol Violation (Status 0204)
Counter 14: DEVPARM Request Rejected (Status 0480)
Counter 15: Incorrect Message Length (Status 0208)
Counter 16: Inhibit Print Key (Status 4001)
Counter 17: Line Length Exceeded (Status 0101)
Counter 18: Warning Line (Status 4080)
Counter 19: End of Forms (Status 8010)
Counter 20: Print Emitter Check (Status 0202)
Counter 21: Print Wire Check (Status 0202)
Counter 22: Translation Check (Status 2000)
Counter 23: End of Page (Status 4020)
Counter 24: Unused

3278/3279 Keyboard

Counter 1: DCA Error (Status 0200)
Counter 2: Port/Cable/Terminal Error (Status 0201)
Counter 3: Keyboard Overrun (Status 0204)
Counter 4: Translation Check (Status 2000)
Counter 5: Segment Overrun (Status 0101)
Counters 6-15: Reserved

3278/3279 Display

Counter 1: DCA Error (Status 0200)
Counter 2: Port/Cable/Terminal Error (Status 0201)
Counters 3-12: Reserved

3262/3287/5210 Printers

Counters 1,2: Reserved
Counter 3: Timeout (Status 0202)
Counters 4-8: Reserved
Counter 9: Power On (Status 0808)
Counter 10: Printer Ready (Status 0840)
Counter 11: Operator Active (Status 0000)
Counters 12,13: Unused
Counter 14: DEVPARM Request Rejected (Status 0480)
Counter 15: DCA Not Active (Status 0200)
Counter 16: Operator Intervention Required (Status 4001)
Counter 17: Power Off (Status 0804)
Counter 18: Data Check — Parity (Status 2008)
Counter 19: No Device on Port (Status 0201)
Counter 20: Equipment Check (Status 0210)
Counter 21: Device I/O Error (Status 0208)
Counter 22: Invalid SCS Control Parameter (Status 2002)
Counter 23: Invalid SCS Control Code (Status 2001)
Counter 24: Cancel Key (Status 0880)
Counter 25: PA1 Key (Status 0801)
Counter 26: PA2 Key (Status 0802)
Counter 27: 8 LPI Key (Status 4008)
Counter 28: 6 LPI Key (Status 4004)
Counter 29: Single Space Key (Status 4010)
Counter 30: Double Space Key (Status 4040)
Counter 31: Mono Case Key (Status 4020)
Counter 32: Dual Case Key (Status 4002)

Statistical Counters for Encryption

Counter 1: Machine Check
Counter 2: Intervention Required
Counter 3: Invalid Key Checksum
Counter 4: Unsuccessful Write
Counters 5-8: Reserved
Counters 9-11: Attempted PIN Validation
Counters 12-14: Unsuccessful PIN Validation
Counters 15-17: Successful PIN Translation
Counters 18-32: Reserved

Disk Statistical Counters

Counter 1: CRC - Cyclic Redundancy Check
Counter 2: Not Ready
Counter 3: No Alternate Sectors Available
Counter 4: Disk Hardware Malfunction
Counter 5: Data Unsafe
Counter 6: Alternate Sector Assignment Failed
Counter 7: Seek Failure
Counter 8: Equipment Check
Counter 9: No Record Found
Counter 10: Successful Engineering Change Correction
Counter 11: Alternate Sector Assigned

Chapter 11. SMSDST Status

Status messages noted with a "P" are possible programming errors; these are explained in the *Controller Programming Library*. Status messages noted with an "H" are possible hardware errors. Status messages noted with an "O" report operational conditions that the application program might need to resolve. Status messages with a "W" are warning messages.

Status Applies to:	*Status*	*Action*	*Meaning*
Work Station	8000	P	Message pending.
	4000	P	Message not found.
	0800	O	Operator attention.
	0402	P	Invalid station ID.
	0401	P	Missing entry point (EP) in current application program.
	0101	P	Message too long.
Indicator Lights Signal*	0480	P	Invalid request.
	0440	O	A test component has not been assigned.
	0200	H	Common loop error.
	0201	P,H	Terminal address error.

*Status related to operation of indicator lights always contain an active bit 1 in first character of byte 1. Thus, the status is the previous write operation plus any residual status.

Diskette/Disk	8200	O	Intervention required, unit check.
	8101	O	Intervention required, wrong length sectors.
	8002	O	Intervention required, diskette not stopped.
	8001	P	Intervention required.
	8000	O,P,H	Diskette not ready.
	4095	O,P	Unit exception, invalid record length.
	4094	O,P	Unit exception, invalid sector count.
	4093	O,P	Unit exception, wrong diskette type.
	4092	O,P	Unit exception, invalid diskette type.
	4090	O,P	Unit exception, too many defaults.
	1000	P,H	Previous read or write in error.
	0A20	H,P	Attention, unit check, invalid volume label.

	0A08	H,P	Attention, unit check, invalid extent.
	0A00	H	Attention, bad diskette or disk.
	0440	O,P	Diskette is logically stopped.
	0404	P	Command reject, drive not supported.
	0403	P	Command reject, volume access inhibited.
	0402	O,H	Command reject, wrong diskette.
	0210	H	Unit check, read error count exceeded.
	0208	H	Unit check, unreadable sectors were logged.
	0203	H	Unit check, bad diskette.
	0201	P	Unit check, wrong length, temporary file record.
Diskette/Disk	0020	P	Duplicate keys.
	0100	P	Record too short.
	0101	P	Record too long; user data truncated.
	0102	P	Diskette sectors too large.
	0142	P	Secondary index exception.
	0200	H	Hardware error.
	0202	P	Unit check; session ID error.
	0204	P	Unit check; control record read.
	0240	P	Unit check, wrong temporary file record ID.
	0401	P	Command reject, buffer alignment or overlay length error.
	0408	P	Invalid extent parameter.
	0410	P	Data set not open.
	0420	P	Data set write protected or unexpired.
	0441	P	Keyed record too short to contain record key.
	0442	P	Duplicate keys not allowed in primary index.
	0444	P	Insufficient buffer space.
	0480	P	Command reject; invalid request.

0482	P	Command reject; data set size conflict.
0481	P	Command reject; optional module missing.
0484	P	Command reject; open conflict.
0485	P	Command reject; invalid temporary file request.
0488	P	Invalid data set specification; invalid data set type; invalid user request flag.
0489	P	Invalid keyed access specification.
04C2	P	Command reject; invalid address.
0800	P	Attention.
0887	P	Attention; data set allocation not complete.
0888	P	Attention; data set definition does not exist.
0889	P	Attention; invalid associated data set specification.
0A40	P	Attention. Unit check; unrecoverable disk error.
0A80	P	Attention; unit check; extent overlap.
0A86	P	Attention; unit check; data set defined, not allocated.
0A88	P	Attention; invalid data set.
0A89	P	Attention, unit check, invalid associated data set specification.
1800	P	Prior operation; attention.
2000	P,H	Data check; unable to read/write a record block.
2001	P	Data exception; incorrect record type.
2204	H,O	Data check, control record read.
4000	P	Unit exception; end-of-file, invalid RSN request, data set full, no unkeyed data set to contain record for keyed LWRITE.

	4001	P	Unit exception, temporary file too large.
	4002	P,O	Unit exception; exclusive use conflict.
	4004	P,O	Unit exception; data set name unknown.
	4008	P,O	Unit exception; incompatible disk or diskette.
	4010	P	Unit exception; too many open requests.
	4011	P	Unit exception; multiple temporary file opens.
	4020	P	Unit exception; data set name not unique.
	4040	P	Unit exception; no unused extent.
	4080	P,O	Unit exception; space unavailable.
	4100	P,O	Unit exception; invalid record length.
3262/3287/5210	8000	O,P	Intervention required.
	4040	O	Line-spacing switch changed from single to double.
	4002	O	Case switch changed from mono to dual case.
	4020	O	Case switch changed from dual to mono case.
	4010	O	Line-spacing switch changed from double to single.
	4008	O	The LPI switch changed from 6 to 8 LPI.
	4004	O	The LPI switch changed from 8 to 6 LPI.
	4001	O	Operator intervention, cover or platen open, end of forms, paper jam. Hold print left on for 10 minutes during printing.
	2008	H	Parity error in print buffer.
	2002	P	Invalid SCS control parameter.
	2001	P	Invalid SCS control code.
	1000	O	Prior operation or asynchronous status.

	0880	O	Cancel switch pressed while printing.
	0840	O	Printer became ready.
	0808	O	Printer completed power-on sequence.
	0804	O	Power off.
	0802	O	PA2 switch activated.
	0801	O	PA1 switch activated.
	0800	O	Wait stopped by attention.
	0480	O,P	Command reject (optional module not loaded or invalid operation).
	0440	O	Printer not assigned as test component.
	0210	H	Equipment check, thermal check, hammer-fire check, belt-synchronous check, or repeatable parity error.
	0208	H	Repeated attempts to write to printer have failed.
	0201	O,H	3262/3287 power off, port does not have 3262/3287 attached, port stopped, or hardware error.
	0200	O,H	DCA stopped or DCA adapter error.
DCA Adapter	0485	O,P,H	Command reject, no DCA devices on any port.
	0484	O,P	Device assigned to another station.
	0483	O	Device type not 00-01.
	0482	O	Port number not 00-07.
	0481	O	Port not 00-07 on enable/ disable port.
	0480	O,P	Command reject (optional module not loaded, or invalid operation).
	0201	H	Unit check DCA port error.
	0200	H	DCA adapter error.

4704/3604/ 3278/3279 Keyboard/Display	4000	H	Tracking error.
	2000	P,H	Translate check.
	0800	O	Read ended by attention (pressing Reset key twice).
	0480	P	Invalid request.
	0440	O,P	Device has not been assigned.
	0204	O,P	PIN data error.
	0202	P	Unit check.
	0201	P,H	Terminal address error in 4704-1 or 3604; port error in 3278, 3279, or 4704-2.
	0200	H	Common loop or DCA adapter error.
	0101	P	Segment overrun.
4704/3604 Magnetic Stripe Encoder	2000	P,H	Translate error.
	0800	O	Wait ended by attention.
	0480	P	Invalid request.
	0440	O,P	Magnetic stripe encoder has not been assigned.
	0202	H	A failure in readying or writing to the encoder.
	0201	P,H	Terminal address error in 4704/3604.
	0200	H	Common loop error.
	0101	P	Record too long.
3606 Keyboard/ Display and 3608 Keyboard/ Display	0101	P	Incorrect length.
	0200	H	Common loop error.
	0201	P,H	Terminal address error.
	0202	P,H	Time-out.
	0404	P	Invalid request.
	0440	P	Invalid request.
	0480	P	Invalid request.
	1000	O	Prior operation.
	2000	P	Data check.
	4001	P	Read issued with no message pending.
3608 Printer	0101	P	Incorrect length.
	0200	H	Common loop error.
	0201	P,H	Terminal address error.
	0202	H	Printer error.
	0404	P	Invalid request.
	0440	P	Invalid request.
	0480	P	Invalid request.
	1000	O	Prior operation.
	2000	P	Data check.
	8000	O	Intervention required.

4710, 3610, 3611, 3612, 3615, 3616, 4720	8202	O,P	Stop switch pressed or tried to print off right edge of passbook (3611-2).
	8020	O,P	(4720) Printer overrun.
	8010	O,P	End of forms.
	8000	O,P	Printer not ready; intervention required.
	4080	O,P	Warning line.
	4020	O,P	End of page.
	4001	O	Printer deactivated.
	2000	P	Translation error; invalid character in data stream.
	1000	O	Prior operation.
	0808	O	Power on (4710/3616/4720).
	0800	O	Wait ended by attention (pressing 4704/3604/3278/3279 Reset key twice.)
	0480	O,P	Optional module not loaded or invalid request.
	0440	O,P	No component assigned at logical address.
	0208	P	Incorrect message length (4720, 4710, 3616).
	0204	H	Protocol or hardware error (4710/3616/4720).
	0202	H	Printer error.
	0201	P,H	Terminal address error in printer.
	0200	H	Common loop error.
	0101	P	Incorrect length (4710/3616/4720).
3614/3624	0801	P	Write ended, data pending.
	0800	O,P	Read ended by attention (pressing 4704/3604 Reset key twice).
	0480	P	Invalid request.
	0440	O	3624 not assigned as test component.
	0201	P,H	Terminal address error in 3614.
	0200	H	Common loop error.
	0101	P	Record too long.
	0100	P	Incorrect length.

Chapter 12. 4700 Copy Files

- **DEFAPB**

```
APBRLD   DEFxx    0,2          APB ROOT LENGTH OR ROOT CONSTANT LEN
APBXIT   DEFxx    ,2           LEXIT INSTRUCTION
APBPID   DEFxx    ,8           APB NAME
APBGID   DEFxx    ,6           APB GENERATION DATE
APBVER   DEFxx    ,1           APB VERSION NUMBER
APBLVL   DEFxx    ,1           APB LEVEL NUMBER
         DEFxx    ,4           RESERVED
APBLTH   DEFxx    ,2           LENGTH OF SEGMENT 14
APBROD   DEFxx    ,2           POINTER TO RESIDENT OVERLAY DIRECTORY

APBFLG   DEFxx    ,1           FLAG BYTE:
APBMVO   DEFxx    APBFLG,1       OPTION FLAG
APBMVOM  EQUATE    X'01'             OPTION FLAG MASK

APBSPR   DEFxx    ,1           RESERVED
APBSUE   DEFxx    ,2           STARTUP        ENTRY POINT
APBAPC   DEFxx    ,2           APCALL         ENTRY POINT
APBPCE   DEFxx    ,2           PROGRAM CHECK ENTRY POINT
         DEFxx    ,2           RESERVED
APBAPI   DEFxx    ,2           ASYNCHRONOUS   ENTRY POINT - POST
APBATE   DEFxx    ,2           ASYNCHRONOUS   ENTRY POINT - TERMINAL
APBACE   DEFxx    ,2           ASYNCHRONOUS   ENTRY POINT - CPU
APBASE   DEFxx    ,2           ASYNCHRONOUS   ENTRY POINT - STATION
APBTME   DEFxx    ,2           ASYNCHRONOUS   ENTRY POINT - TIMER
APBALA   DEFxx    ,2           ASYNCHRONOUS   ENTRY POINT - ALA
APBPCL   DEFxx    ,2           RESERVED
APBIVN   DEFxx    ,2           1ST HWD OF INSTR SECTION ADR
APBIAD   DEFxx    ,2           2ND HWD OF INSTR SECTION ADR
APBSDT   DEFxx    ,2           SEGMENT DEFINITION TABLE ADDR
APBDEL   DEFxx    ,2           DELIMITER TABLE ADDRESS
         DEFxx    ,10          RESERVED

APBFUI   DEFxx    ,0           FIRST USER CONSTANT OR INSTRUCTION
```

- **DEFCDK**

```
* * *    'COMPDKT' INSTRUCTION DATA SET MODIFICATION ENTRY

CDKENT   DEFxx    s,22         COMPRESS DISKETTE REQUEST ENTRY
CDKDSN   DEFxx     CDKENT,17   DATA SET NAME
CDKFLG   DEFxx    s,1          FLAG BYTE:
CDKFL0M  EQUATE     X'80'             TRUNCATE DATA SET
CDKFL1M  EQUATE     X'40'             SET EOD EQUAL TO EOE
CDKFL2M  EQUATE     X'20'             MAINTAIN TRACK ALIGNMENT
CDKFL3M  EQUATE     X'10'             DELETE DATA SET
CDKNUM   DEFxx    s,4          BINARY SECTOR COUNT
```

- **DEFCPL**

```
CPLPAR   DEFxx    s,19              COMPRESS/COMPACT PARMETER LIST
CPLINS   DEFxx     CPLPAR,2         SEGMENT CONTAINING INPUT AREA
CPLIND   DEFxx    s,2              DISPLACEMENT TO INPUT AREA
CPLINL   DEFxx    s,2              LENGTH OF INPUT AREA
CPLOUS   DEFxx    s,2              SEGMENT CONTAINING OUTPUT ARE
CPLOUD   DEFxx    s,2              DISPLACEMENT TO OUTPUT AREA
CPLOUL   DEFxx    s,2              LENGTH OF OUTPUT AREA
CPLPRI   DEFxx    s,1              PRIME COMPRESSION CHARACTER
CPLFLG   DEFxx    s,1              INPUT FLAG BYTE
CPLFCT   EQUATE  X'80'                    PERFORM COMPACTION
CPLFCR   EQUATE  X'40'                    PERFORM COMPRESSION
CPLTBS   DEFxx    s,2              SEG CONTAINS COMPACT TABLE IN
CPLTBD   DEFxx    s,2              DISPLACEMENT TO TABLE AREA
CPLTST   DEFxx    s,1              TERMINATION STATUS
CPLTOV   EQUATE  X'80'                    OUTPUT OVERFLOW
CPLTIL   EQUATE  X'40'                    INITIAL INPUT LENGTH IS ZERO
```

- **DEFDCP**

```
DCPPAR   DEFxx    s,19             DECOMPRESS/DECOMPACT PARAMETER LIST
DCPINS   DEFxx     DCPPAR,2        SEGMENT CONTAINING INPUT AREA
DCPIND   DEFxx    s,2              DISPLACEMENT TO INPUT AREA
DCPINL   DEFxx    s,2              LENGTH OF INPUT AREA
DCPOUS   DEFxx    s,2              SEGMENT CONTAINING OUTPUT AREA
DCPOUD   DEFxx    s,2              DISPLACEMENT TO OUTPUT AREA
DCPOUL   DEFxx    s,2              LENGTH OF OUTPUT AREA
DCPPRI   DEFxx    s,1              PRIME COMPRESSION CHARACTER
DCPFLG   DEFxx    s,1              INPUT FLAG BYTE
DCPFCT   EQUATE  X'80'                    INPUT DATA IN COMPACT CODE
DCPTBS   DEFxx    s,2              SEGMENT CONTAINING DEC TABLE
DCPTBD   DEFxx    s,2              DISP TO DECOMPACTION TABLE
DCPTST   DEFxx    s,1              TERMINATION STATUS
DCPTOV   EQUATE  X'80'                    OUTPUT OVERFLOW
DCPTIV   EQUATE  X'40'                    INPUT OVERFLOW
DCPTCE   EQUATE  X'20'                    COMPACT CODE INPUT
DCPTSL   EQUATE  X'10'                    SCB COUNT FIELD IS ZERO
DCPTIL   EQUATE  X'08'                    INITIAL INPUT LENGTH IS ZERO
```

- **DEFDKT**

```
DKTRCD   DEFxx    s,1              REQUEST CODE
DKTRADM  EQUATE     X'01'                 ALLOCATE DATA SET
DKTRUDM  EQUATE     X'02'                 UPDATE HEADER LABEL
DKTRODM  EQUATE     X'03'                 OPEN DATA SET
DKTRDDM  EQUATE     X'04'                 DEALLOCATE DATA SET
DKTRCDM  EQUATE     X'05'                 CLOSE DATA SET
DKTRIVM  EQUATE     X'06'                 INHIBIT ACCESS TO VOLUME
DKTRPAM  EQUATE     X'07'                 PERMIT ACCESS TO VOLUME
DKTRBQM  EQUATE     X'08'                 BUFFER INQUIRY
DKTRBRM  EQUATE     X'09'                 BUFFER RELEASE
DKTRVQM  EQUATE     X'0A'                 VOLUME ID INQUIRY
DKTROQM  EQUATE     X'0B'                 OPEN STATUS INQUIRY
DKTRRSM  EQUATE     X'0C'                 RESET DATA SET INPUT POINTER
DKTRQSM  EQUATE     X'0D'                 UNALLOCATED SPACE INQUIRY
DKTRRGM  EQUATE     X'0E'                 REORGANIZE ASDS OR KSAP
DKTRQLM  EQUATE     X'0F'                 QUERY EXTENDED LABEL INFO
DKTRRNM  EQUATE     X'10'                 RENAME DATA SET
DKTLST   DEFxx    s,0              BEGINNING OF PARAMETER LIST
```

```
* ALLOCATE DATA SET ( REQUEST CODE: DKTRADM )

DKT1F1   DEFxx    DKTLST,1     FLAG BYTE
DKT1FBM  EQUATE    X'80'                ALLOCATE ON TRACK BOUNDARY
DKT1FTM  EQUATE    X'40'                EXTENT SIZE GIVEN IN TRACKS
DKT1FKM  EQUATE    X'20'                EXTENT SIZE GIVEN IN K BYTES
DKT1ES   DEFxx   s,2            EXTENT SIZE (TRKS, SCTS, KBYTS)
DKT1NA   DEFxx   s,2            NUMBER OF SECTORS TO ASSIGN
DKT1NI   DEFxx   s,2            NUMBER OF SECTORS TO INITIALIZE
DKT1IV   DEFxx   s,1            INITIALIZATION VALUE
DKT1F2   DEFxx   s,1            SECONDARY FLAG BYTE
DKT1FTPM EQUATE    X'10'                TEMP TYPE OF DATA SET
DKT1FESM EQUATE    X'20'                ESDS TYPE OF DATA SET
DKT1FEDM EQUATE    X'40'                EDDS TYPE OF DATA SET
DKT1FASM EQUATE    X'50'                ASDS TYPE OF DATA SET
DKT1FRKM EQUATE    X'60'                RKAP TYPE OF DATA SET
DKT1FKSM EQUATE    X'70'                KSAP TYPE OF DATA SET
DKT1FCM  EQUATE    X'04'                ALLOCATE USING STORED DEF'N
DKT1FDM  EQUATE    X'08'                STORE DEF'N; DO NOT ALLOCATE
DKT1FOM  EQUATE    X'0C'                STORE DEF'N; AND ALLOCATE
DKT1HD   DEFxx   s,128          HEADER LABEL (DEFINED BELOW)
DKT1SP   DEFxx    DKT1HD,80     ALLOCATE DATA SET HEADER

* EXTENDED ALLOCATE DATA SET PARAMETER LIST ( REQUEST CODE: DKTRADM )

DKT1XP   DEFxx   s,9            EXTENDED ALLOCATE PARAMETER LIST
DKT1KS   DEFxx    DKT1XP,2      KEY STARTING OFFSET
DKT1KL   DEFxx   s,1            KEY LENGTH
DKT1DK   DEFxx   s,1            EXTENSION FLAG BYTE
DKT1DKAM EQUATE    X'01'                DUPLICATE KEYS ALLOWED
DKT1NBI  DEFxx   s,1            NUMBER OF BYTES TO INITIALIZE
DKT1XC   DEFxx   s,1            NUMBER OF SECONDARY EXTENTS
DKT1XS   DEFxx   s,2            SEC EXTENT SIZE IN K BYTES
DKT1AC   DEFxx   s,1            NUMBER OF ASSOCIATED DATA SETS
DKT1AN   DEFxx   s,119          ASSOCIATED DATA SET NAMES

* UPDATE HEADER LABEL ( REQUEST CODE: DKTRUDM )

DKT2F1   DEFxx    DKTLST,1     FLAG BYTE
DKT2FBM  EQUATE    X'80'                UPDATE BYPASS INDICATOR
DKT2FSM  EQUATE    X'40'                UPDATE DATA SET SECURITY
DKT2FWM  EQUATE    X'20'                UPDATE WRITE PROTECT INDICATOR
DKT2FVM  EQUATE    X'10'                UPDATE VERIFY/COPY INDICATOR
DKT2FRM  EQUATE    X'01'                RETURN HEADER LABEL
DKT2BI   DEFxx   s,1            NEW BYPASS INDICATOR
DKT2DS   DEFxx   s,1            NEW DATA SET SECURITY VALUE
DKT2WP   DEFxx   s,1            NEW WRITE PROTECT INDICATOR
DKT2VC   DEFxx   s,1            NEW VERIFY/COPY INDICATOR

* OPEN DATA SET ( REQUEST CODE: DKTRODM )

DKT3F1   DEFxx    DKTLST,1     FLAG BYTE
DKT3FTM  EQUATE    X'80'                TEMPORARY FILE DATA SET
DKT3FWM  EQUATE    X'40'                WARM START
DKT3FXM  EQUATE    X'20'                EXCLUSIVE USE
DKT3FRM  EQUATE    X'01'                RETURN HEADER LABEL
DKT3DN   DEFxx   s,17           DATA SET NAME

* DEALLOCATE DATA SET ( REQUEST CODE: DKTRDDM )

DKT4F1   DEFxx    DKTLST,1     FLAG BYTE
DKT4FPM  EQUATE    X'80'                RETAIN DATA SET DEFINITION
DKT4DN   DEFxx   s,17           DATA SET NAME
DKT4TD   DEFxx   s,6            TODAY'S DATE

* CLOSE DATA SET ( REQUEST CODE: DKTRCDM )

DKT5F1   DEFxx    DKTLST,1     FLAG BYTE
DKT5FAM  EQUATE    X'80'                CLOSE FOR ALL STATIONS
DKT5FIM  EQUATE    X'40'                IGNORE BUFFER ERRORS


* VOLUME ID INQUIRY ( REQUEST CODE: DKTRVQM )

DKTAF1   DEFxx    DKTLST,1     FLAG BYTE
DKTAVI   DEFxx ,6               VOLUME ID RETURN AREA

* OPEN STATUS INQUIRY ( REQUEST CODE: DKTROQM )

DKTBF1   DEFxx    DKTLST,1     FLAG BYTE
DKTBOM   DEFxx   s,4            OPEN BIT MAP RETURN AREA
DKTBSM   DEFxx    DKTBOM,8      OPEN BIT MAP, 60 STATIONS
DKTHDR   DEFxx    DKT1HD,128    HEADER LABEL AREA
DKTHID   DEFxx    DKTHDR,4      HEADER LABEL IDENTIFIER
         DEFxx   s,1            RESERVED
DKTHDN   DEFxx   s,17           DATA SET NAME
DKTHBL   DEFxx   s,5            BLOCK LENGTH
DKTHRA   DEFxx   s,1            RECORD ATTRIBUTE
DKTHBE   DEFxx   s,5            BEGINNING OF EXTENT ADDRESS
DKTHPL   DEFxx   s,1            PHYSICAL SECTOR LENGTH
DKTHEE   DEFxx   s,5            END OF EXTENT ADDRESS
DKTHRB   DEFxx   s,1            RECORD/BLOCK FORMAT
DKTHBI   DEFxx   s,1            BYPASS INDICATOR
DKTHDS   DEFxx   s,1            DATA SET SECURITY
DKTHWP   DEFxx   s,1            WRITE PROTECT INDICATOR
DKTHET   DEFxx   s,1            EXCHANGE TYPE INDICATOR
DKTHBTM  EQUATE    X'40'                BASIC EXCHANGE TYPE
DKTHETM  EQUATE    C'E'                 FULL EXCHANGE TYPE
DKTHMV   DEFxx   s,1            MULTI-VOLUME INDICATOR
DKTHVS   DEFxx   s,2            VOLUME SEQUENCE NUMBER
DKTHCD   DEFxx   s,6            CREATION DATE
DKTHRL   DEFxx   s,4            LOGICAL RECORD LENGTH
DKTHON   DEFxx   s,5            OFFSET TO NEXT RECORD SPACE
         DEFxx   s,4            RESERVED
DKTHXD   DEFxx   s,6            EXPIRATION DATE
DKTHVC   DEFxx   s,1            VERIFY/COPY INDICATOR
DKTHDO   DEFxx   s,1            DATA SET ORGANIZATION
DKTHDDM  EQUATE    C'D'                 DIRECT ORGANIZATION
DKTHDSM  EQUATE    C'S'                 SEQUENTIAL ORGANIZATION
DKTHED   DEFxx   s,5            END OF DATA ADDRESS
         DEFxx   s,1            RESERVED
DKTHPD   DEFxx   s,48           PADDING ZEROS OR SPACES

* UNALLOCATED SPACE INQUIRY ( REQUEST CODE: DKTRQSM )
```

```
DKTDF1   DEFxx    DKTLST,1     FLAG BYTE
DKTDUDS  DEFxx    s,4          UNALLOCATED SPACE RETURN AREA
DKTDUD0  DEFxx     DKTDUDS,2   FIRST TWO BYTES OF RETURN AREA
DKTDUD2  DEFxx    s,2          SECOND TWO BYTES OF RETURN AREA
DKTDU1   DEFxx    s,2          # BLOCKS <4K
DKTDU2   DEFxx    s,2          # BLOCKS >=4K AND <16K
DKTDU3   DEFxx    s,2          # BLOCKS >=16K AND <64K
DKTDU4   DEFxx    s,2          # BLOCKS >=64K AND <256K
DKTDU5   DEFxx    s,2          # BLOCKS >=256K AND <1024K
DKTDU6   DEFxx    s,2          # BLOCKS >=1024K

* REORGANIZE DATA SET REQUEST ( REQUEST CODE: DKTRRGM )

DKTGF1   DEFxx     DKTLST,1    FLAG BYTE
DKTGDN   DEFxx    s,17         DATA SET NAME


* HEADER LABEL EXTENSION INQUIRY ( REQUEST CODE: DKTRQLM )

DKTQF1   DEFxx     DKTLST,1    FLAG BYTE
DKTQRPM  EQUATE     X'80'                QUERY BY RELATIVE POSITION
DKTQNPM  EQUATE     X'C0'                QUERY TO NEXT RELATIVE POS'N
DKTQPRL  DEFxx    s,2          RELATIVE POS'N OF LABEL
         DEFxx    s,1          UNUSED POSITION (BLANK)
DKTQDN   DEFxx    s,17         DATA SET NAME
DKTQBL   DEFxx    s,5          BLOCK LENGTH
DKTQRA   DEFxx    s,1          RECORD ATTRIBUTE
DKTQBE   DEFxx    s,5          BEGINNING OF EXTENT ADDRESS
DKTQPL   DEFxx    s,1          PHYSICAL SECTOR LENGTH
DKTQEE   DEFxx    s,5          END OF EXTENT ADDRESS
DKTQRB   DEFxx    s,1          RECORD/BLOCK FORMAT
DKTQBI   DEFxx    s,1          BYPASS INDICATOR
DKTQDS   DEFxx    s,1          DATA SET SECURITY
DKTQWP   DEFxx    s,1          WRITE PROTECT INDICATOR
DKTQET   DEFxx    s,1          EXCHANGE TYPE INDICATOR
DKTQBTM  EQUATE     X'40'                BASIC EXCHANGE TYPE
DKTQETM  EQUATE     C'E'                 FULL EXCHANGE TYPE
DKTQMV   DEFxx    s,1          MULTI-VOLUME INDICATOR
DKTQVS   DEFxx    s,2          VOLUME SEQUENCE NUMBER
DKTQCD   DEFxx    s,6          CREATION DATE
DKTQRL   DEFxx    s,4          LOGICAL RECORD LENGTH
DKTQON   DEFxx    s,5          OFFSET TO NEXT RECORD SPACE
         DEFxx    s,4          RESERVED
DKTQXD   DEFxx    s,6          EXPIRATION DATE
DKTQVC   DEFxx    s,1          VERIFY/COPY INDICATOR
DKTQDO   DEFxx    s,1          DATA SET ORGANIZATION
DKTQDDM  EQUATE     C'D'                 DIRECT ORGANIZATION
DKTQDSM  EQUATE     C'S'                 SEQUENTIAL ORGANIZATION
DKTQED   DEFxx    s,5          END OF DATA ADDRESS
         DEFxx    s,1          RESERVED

DKTQCSZ  DEFxx    s,2          CURRENT DATASET SIZE (K-BYTES)
DKTQFG1  DEFxx    s,1          FLAG BYTE
DKTQFBM  EQUATE     X'80'                ALLOCATE ON TRACK BOUNDARY
DKTQFTM  EQUATE     X'40'                EXTENT SIZE GIVEN IN TRACKS
DKTQFKM  EQUATE     X'20'                EXTENT SIZE GIVEN IN K BYTES
DKTQES   DEFxx    s,2          EXTENT SIZE (TRKS/SCTS/KBYTES)
DKTQNA   DEFxx    s,2          NUMBER OF SECTORS TO ASSIGN
DKTQNI   DEFxx    s,2          NUMBER OF SECTORS TO INITIALIZE
DKTQIV   DEFxx    s,1          INITIALIZATION VALUE
DKTQF2   DEFxx    s,1          SECONDARY FLAG BYTE
DKTQFTPM EQUATE     X'10'                TEMP TYPE OF DATA SET
DKTQFESM EQUATE     X'20'                ESDS TYPE OF DATA SET
DKTQFEDM EQUATE     X'40'                EDDS TYPE OF DATA SET
DKTQFASM EQUATE     X'50'                ASDS TYPE OF DATA SET
DKTQFRKM EQUATE     X'60'                RKAP TYPE OF DATA SET
DKTQFKSM EQUATE     X'70'                KSAP TYPE OF DATA SET
DKTQFCM  EQUATE     X'04'                ALLOCATE USING STORED DEF'N
DKTQFDM  EQUATE     X'08'                STORE DEF'N; DO NOT ALLOCATE
DKTQFDM  EQUATE     X'0C'                STORE DEF'N; AND ALLOCATE
DKTQKS   DEFxx    s,2          KEY STARTING OFFSET
DKTQKL   DEFxx    s,1          KEY LENGTH
DKTQDK   DEFxx    s,1          EXTENSION FLAG BYTE
DKTQDKAM EQUATE     X'01'                DUPLICATE KEYS NOT ALLOWED
DKTQNBI  DEFxx    s,1          NUMBER OF BYTES TO INITIALIZE
DKTQXC   DEFxx    s,1          NUMBER OF SECONDARY EXTENTS
DKTQXS   DEFxx    s,2          SEC EXTENT SIZE IN K BYTES
DKTQXA   DEFxx    s,1          NUMBER OF SEC'Y EXTENTS ALLOC
DKTQAC   DEFxx    s,1          NUMBER OF ASSOCIATED DATA SETS
DKTQAN   DEFxx    s,119        ASSOCIATED DATA SET NAMES

* RENAME DATA SET REQUEST ( REQUEST CODE: DKTRRNM )

DKTRF1   DEFxx     DKTLST,1    FLAG BYTE
DKTRDN   DEFxx    s,17         CURRENT DATA SET NAME
DKTRRN   DEFxx    s,17         NEW DATA SET NAME
DKTEND   DEFxx     DKTRCD,(D:DKTQAN+L:DKTQAN-D:DKTRCD) MAX SIZE
```

- **DEFDMS**

```
DMSBEG      DEFxx    s,0,BDY=HALF BEGINNING OF DMS
DMSFLAG     DEFxx     DMSBEG,2    CONTROL BYTES
DMSFLG1     DEFxx     DMSFLAG,1   FIRST CONTROL BYTE
DMSFLG2     DEFxx    s,1          SECOND CONTROL BYTE
DMSSTAT     DEFxx    s,2          STATUS BYTES
DMSSTS1     DEFxx     DMSSTAT,1   FIRST STATUS BYTE
DMSSTS2     DEFxx    s,1          SECOND STATUS BYTE
DMSERCD     DEFxx    s,1          ERROR CODE BYTE
DMSEID      DEFxx    s,1          EOM/EID VALUE FROM SMSEID
DMSLSEQ     DEFxx    s,1          LINE SEQUENCE NUMBER
DMSDELM     DEFxx    s,1          USER DELIMITER FOR NATIVE CONTROL
DMSIML      DEFxx    s,2          VALUE FROM SMSIML
DMSFSEQ     DEFxx    s,2          FIELD SEQUENCE NUMBER
DMSFLG      DEFxx    s,2          DATSM WORK STATE FLAGS
DMSMOD      DEFxx    s,2          MODE FIELD
DMSHEND     DEFxx    s,0          END OF DMS HEADER FIELDS
DMSHSIZ     EQUATE (D:DMSHEND-D:DMSBEG)  SIZE OF THE HEADER AREA
```

```
*
*         END OF HEADER AREA
*
DMSPPF      DEFxx   s,2          PRESENTATION POSITION FIELD
DMSPPFR     DEFxx    DMSPPF,1    ROW POSITION
DMSPPFC     DEFxx   s,1          COLUMN POSITION
DMSACT      DEFxx   s,2          ACTUAL DEVICE DIMENSIONS
DMSACTR     DEFxx    DMSACT,1
DMSACTC     DEFxx   s,1
DMSACD      DEFxx   s,2          ACTUAL DEVICE COORDINATES
DMSACDR     DEFxx    DMSACD,1
DMSACDC     DEFxx   s,1
DMSBCD      DEFxx   s,2          IMAGE BUFFER COORDINATES
DMSBCDR     DEFxx    DMSBCD,1
DMSBCDC     DEFxx   s,1
DMSWIN      DEFxx   s,2          WINDOW DIMENSIONS (ROW,COL)
DMSWINR     DEFxx    DMSWIN,1
DMSWINC     DEFxx   s,1
DMSORG      DEFxx   s,2          ORIGINAL DEVICE DIMENSIONS
DMSORGR     DEFxx    DMSORG,1
DMSORGC     DEFxx   s,1
DMSRSA      DEFxx   s,1          RESERVED
DMSIBF      DEFxx   s,3          IMAGE BUFFER ADDRESS
DMSRSB      DEFxx   s,1          RESERVED
DMSFCT      DEFxx   s,3          FIELD CONTROL TABLE HEADER ADDR
DMSRS2      DEFxx   s,1          RESERVED
DMSDMS      DEFxx   s,3          DMS SELF LOCATE SECTION
DMSDMSEG    DEFxx    DMSDMS,1     SEGMENT NUMBER
DMSDMSPL    DEFxx   s,2           DISPLACEMENT
DMSLDA      DEFxx   s,1          DEVICE LDA
DMSRS3      DEFxx   s,3          RESERVED
DMSEND      DEFxx   s,0
DMSSIZ      EQUATE (D:DMSEND-D:DMSBEG)
DMSSECT     DEFxx    DMSBEG,DMSSIZ  FULL DMS DEFINED AS ONE FIELD
DMSHEAD     DEFxx    DMSBEG,DMSHSIZ FULL HDR DEFINED AS ONE FIELD

*
** PUTDMS CONTROL FLAG EQUATES
*
DMSPPFF   EQUATE X'8000'  PRESENTATION POSITION
DMSORGF   EQUATE X'4000'  ORIGINAL DEVICE SIZE
DMSACTF   EQUATE X'1000'  ACTUAL DEVICE SIZE
DMSACDF   EQUATE X'0400'  ACTUAL DEVICE COORDINATES
DMSBCDF   EQUATE X'0200'  IMAGE BUFFER COORDINATES FOR DISPLAY USE
DMSWINF   EQUATE X'0100'  WINDOW SIZE FOR DISPLAY USE

*
** GETFLD AND PUTFLD CONTROL FLAG EQUATES
*
DMSTYP    EQUATE    X'8000' DATASTREAM TYPE 0 = SCS, 1 = 3270
DMSIOT    EQUATE    X'4000' DATASTREAM BOUND  0 = DISP, 1 = PROCESSOR
DMSTIO    EQUATE    X'C000' DATASTREAM = 3270 PROCESSOR BOUND
*       00XX XXXX = SCS DISPLAY BOUND FLOW
*       01XX XXXX = SCS PROCESSOR BOUND FLOW
*       10XX XXXX = 3270 DISPLAY BOUND FLOW (INVALID FOR GETFLD)
*       11XX XXXX = 3270 PROCESSOR BOUND FLOW (INVALID FOR PUTFLD)
*
DMSSEG    EQUATE    X'0001' GETFLD SEGMENTS FIELDS LINE BY LINE
*
** GETFLD CONTROL FLAG EQUATES
*
DMSRMA    EQUATE    X'2000' READ MODIFIED ALL REQUEST
DMSDFO    EQUATE    X'1000' GET DATA ONLY FROM FIELD
DMSMVF    EQUATE    X'0800' GET ALL FIELDS OF REQUESTED CLASS
DMSMCF    EQUATE    X'0000' XXXX X000 = GET CURRENT FIELD
DMSMAD    EQUATE    X'0900' XXXX 1001 = GET ALL DISPLAY BOUND FIELDS
DMSMND    EQUATE    X'0100' XXXX 0001 = GET ONE DISPLAY BOUND FIELD
DMSMAP    EQUATE    X'0A00' XXXX 1010 = GET ALL PROCESSOR BOUND FIELDS
DMSMNP    EQUATE    X'0200' XXXX 0010 = GET ONE PROCESSOR BOUND FIELD
DMSMAE    EQUATE    X'0B00' XXXX 1011 = GET ALL USER FLAGGED FIELDS
DMSMNE    EQUATE    X'0300' XXXX 0011 = GET ONE USER FLAGGED FIELD
DMSMAU    EQUATE    X'0C00' XXXX 1100 = GET ALL UNPROTECTED FIELDS
DMSMNU    EQUATE    X'0400' XXXX 0100 = GET ONE UNPROTECTED FIELD
DMSMSQ    EQUATE    X'0500' XXXX X101 = GET FIELD OF SEQUENCE NUMBER
DMSMAL    EQUATE    X'0F00' XXXX 1111 = GET ALL FIELDS IN THE BUFFER
DMSMAA    EQUATE    X'0700' XXXX 0111 = GET ONE FIELD ANY TYPE
DMSGFB    EQUATE    X'0080' READ FULL BUFFER REQUIRED
DMSREI    EQUATE    X'0040' REISSUE GETFLD
DMSATT    EQUATE    X'0020' PROVIDE ATTRIBUTES FOR FIELD
DMSPRT    EQUATE    X'0010' OUTPUT IS FOR A PRINTER
DMSDEL    EQUATE    X'0008' INSERT USER DELIMITERS
DMSCAD    EQUATE    X'0004' GENERATE CURSOR ADDRESS FOR 3604
DMSPPC    EQUATE    X'0002' SET PRESENTATION POSITION TO THIS FIELD

*
** PUTFLD CONTROL FLAG EQUATES
*
DMSMDT    EQUATE    X'1000' TURN OFF MODIFIED DATA TAGS
DMSUSF    EQUATE    X'0800' TURN ON USER EXIT FLAG IN FLAGS BYTE
DMSSEQ    EQUATE    X'0400' USE SEQUENCE NUMBER TO FIND FIELD
DMSUDL    EQUATE    X'0200' DATASTREAM HAS USER DELIMITERS
DMSREP    EQUATE    X'0040' THIS IS A PUTFLD REISSUE
DMSSFP    EQUATE    X'0100' SET FIELD TO PROTECT
DMSFWT    EQUATE    X'2000' FORCE WRITE

*
** COMMON STATUS FLAG EQUATES
*
DMSMCR    EQUATE    X'0800' CURSOR POSITION OUT OF WINDOW
DMSERR    EQUATE    X'0200' AN EXCEPTION EXISTS.  CHECK THE ERROR CODE
DMSMPL    EQUATE    X'0100' THE OPERATION IS COMPLETED SUCCESSFULLY
DMSBOF    EQUATE    X'0010' BUFFER OVERFLOW
DMSFNF    EQUATE    X'0008' FIELD NOT FOUND

*
** GETFLD STATUS FLAG EQUATES
*
DMSNDS    EQUATE    X'8000' THE REQUESTED FIELD WAS NON-DISPLAYABLE
DMSUFL    EQUATE    X'0400' USER FLAG WAS ON FOR REQUESTED FIELD
DMSPRO    EQUATE    X'0002' THE REQUESTED FIELD WAS PROTECTED
DMSKJF    EQUATE    X'0001' THE REQUESTED FIELD WAS KANJI FIELD
```

```
*
** PUTFLD STATUS FLAG EQUATES
*
DMSRNT   EQUATE   X'8000' ISSUE LREAD KEYBOARD NOTRACK
DMSRTK   EQUATE   X'4000' ISSUE LREAD KEYBOARD TRACK
DMSRII   EQUATE   X'2000' REISSUE PUTFLD
DMSEOM   EQUATE   X'1000' EID RECD FROM KBD - PROCESSING REQUIRED
DMSHST   EQUATE   X'0800' ISSUE LWRITE TO HOST
DMSTRM   EQUATE   X'0400' ISSUE LWRITE TO DISPLAY
DMSRFB   EQUATE   X'0080' ISSUE GETFLD FOR FULL BUFFER READ
DMSUSP   EQUATE   X'0040' THIS FIELD HAD A USER FLAG TURNED ON
DMSRDM   EQUATE   X'0020' ISSUE GETFLD FOR PROCESSOR BOUND MODIFIED
DMSRML   EQUATE   X'0004' READ MODIFIED REQUEST RECEIVED FROM HOST

*
**  DATSM WORK STATE FLAGS
*
DMSFOR   EQUATE   X'4000' B'1' SCREEN FORMATTED, B'0' NOT FORMATTED
DMSUNP   EQUATE   X'2000' B'1' UNPROTECTED AREA ON SCREEN, B'0' NONE
DMSIRT   EQUATE   X'0200' B'1' INSERT MODE, B'0' NORMAL MODE

*
**  ERROR CODE EQUATES
*
DMSEC1   EQUATE   1       INVALID GETFLD REQUEST
DMSEC4   EQUATE   4       INVALID COMMAND OR ORDER IN DATA STREAM
DMSEC6   EQUATE   6       INVALID PUTFLD REISSUE
DMSEC7   EQUATE   7       NO PROTECTED FIELDS DEFINED IN IMAGE BUFFER
DMSEC8   EQUATE   8       DEVICE ERROR ON BUFFERED I/O
DMSEC10  EQUATE   10      STATION NOT DEFINED FOR DATSM USE
DMSEC11  EQUATE   11      RESERVED
DMSEC12  EQUATE   12      PUTDMS PARMETER EXCEEDS PERMITTED RANGE
DMSEC14  EQUATE   14      FIELD CONTROL TABLE OVERFLOW
DMSEC15  EQUATE   15      INVALID ADDRESS IN DATA STREAM

*
**  MODE FLAGS
*
DMSKJM   EQUATE   X'8000' KANJI (3270 EXTENDED) MODE
DMSEAB   EQUATE   X'4000' APL/EXTENDED ATTRIBUTE MODE
DMSBFM   EQUATE   X'2000' BUFFERED MODE
DMSNFF   EQUATE   X'1000' NO FORCED FIELD GEN FOR SCS
```

- **DEFELP**

```
ELPARM   DEFxx    s,14          ENTIRE PARAMETER LIST
ELPOVN   DEFxx     ELPARM,8     OVERLAY NAME
ELPFLG   DEFxx    s,1           SECTION LOAD FLAGS
ELPCSF   EQUATE     X'80'               LOAD CONSTANT SECTION
ELPISF   EQUATE     X'40'               LOAD INSTRUCTION SECTION
ELPSEG   DEFxx    s,1           CONSTANT SECTION SEGMENT NUMBER
ELPCLA   DEFxx    s,2           CONSTANT SECTION LOAD ADDRESS
ELPILA   DEFxx    s,2           INSTRUCTION SECTION LOAD ADDRESS
```

- **DEFENC**

```
ENCPAR   DEFxx    s,,24          INSTRUCTION PARAMETER LIST
ENCFLG   DEFxx    s,ENCPAR,1     FLAG BYTE:
ENCPADM  EQUATE     X'80'                PAD OPTION
ENCPADC  DEFxx    s,,1           PAD CHARACTER VALUE
ENCKEY   DEFxx    s,,8           KEY (ENCRYPTED UNDER MASTER KEY)
ENCMAC   DEFxx    s,,4           MSG AUTHENTICATION CODE (MAC)
ENCIV    DEFxx    s,ENCMAC,8     INITIALIZATION VECTOR
         DEFxx    s,,1           RESERVED
ENCSEG   DEFxx    s,,1           SEGMENT NUMBER OF DATA
ENCDISP  DEFxx    s,,2           DISPLACEMENT TO DATA
ENCLEN   DEFxx    s,,2           LENGTH OF DATA
```

- **DEFESP**

```
ESPREQ   DEFxx    ,1            REQUEST CODE
ESPFLG   DEFxx    ,1            OPTION FLAGS
ESPRST   EQUATE     X'80'               RESET COUNTER AFTER READOUT
ESPNXT   EQUATE     X'40'               REQUEST NEXT COUNTERS DATA
ESPEID   DEFxx    ,2            EXTENDED COUNTER ID
ESPDID   DEFxx    ,2            DEVICE ID
ESPTOT   DEFxx    ,6            TOTAL BYTES
ESPERR   DEFxx    ,4            ERROR BYTES
ESPNDV   DEFxx    ,1            NUMBER OF DEVICES ASSIGNED
ESPSID   DEFxx    ESPEID,2      DEVICE STAT CTR ID
ESPTYP   DEFxx    ,1            DEVICE TYPE CODE
ESPFEA   DEFxx    ,1            FEATURE FLAGS
ESPSTA   DEFxx    ,1            ASSIGNED STATION ID
ESPNCT   DEFxx    ,1            NUMBER OF COUNTERS
ESPSCTR  DEFxx    ,32           BYTE COUNTERS

*
* * * *  DEFINITION OF FEATURE FLAGS IN ESPFEA
*
* LOOP (DEVICE TYPE X'80'
*
ESPLPCLK EQUATE    X'80'         CLOCKING LOOP
ESPLP06  EQUATE    X'08'         600 BPS SPEED
ESPLP12  EQUATE    X'04'         1200 BPS SPEED
ESPLP24  EQUATE    X'02'         2400 BPS SPEED
ESPLP48  EQUATE    X'01'         4800 BPS SPEED

*
* COMMON FOR ALL LOOP ATTACHED DEVICES
*
ESPFFRA  EQUATE    X'08'         READ OPERATION DEVICE
ESPFFWA  EQUATE    X'04'         WRITE OPERATION DEVICE
ESPFFSA  EQUATE    X'02'         SHARED ADDRESS SLOT DEVICE
ESPFFSO  EQUATE    X'01'         SHARED OPERATOR DEVICE

*
* FLAGS THAT ARE DEVICE DEPENDENT
*
* 3604 DISPLAY
ESPDSCUR EQUATE    X'80'         CURSOR SET ON AT DEVICE INITIALIZATION

*
* 3604 KEYBOARD
ESPKBCUR EQUATE    X'80'         CURSOR IS NOT SET ON/OFF AT READ TIME
ESPKBERT EQUATE    X'40'         ERTSL OPTION
ESPKBPIN EQUATE    X'10'         PIN PAD AVAILABLE AND USED
```

```
*
* 3610/3612 DOCUMENT PRINTER
ESPPRPFT EQUATE    X'20'          PIN FEED TRACTOR
ESPPRACT EQUATE    X'10'          AUTOMATIC CUT FORM MODE

*
* 3618 ADMINISTRATIVE PRINTER
ESPADEPL EQUATE    X'80'          EXPANDED PRINT LINE
ESPADDFF EQUATE    X'20'          DUAL FORMS FEED
*
```

- **DEFFAP**

```
FAPARM   DEFxx    s,16           ENTIRE PARAMETER LIST
FAPAPN   DEFxx     FAPARM,8      AP NAME
FAPFLG   DEFxx    s,1            REQUEST/RESPONSE FLAGS
FAPCLR   EQUATE    X'80'                 INPUT: READ=0 / READ,RESET=1
FAPTRN   EQUATE    X'40'                 OUTPUT: RESIDENT=0 / TRANSIENT=
         DEFxx    s,1            RESERVED
FAPCAL   DEFxx    s,2            NUMBER OF CALLS MADE
FAPLOD   DEFxx    s,2            NUMBER OF LOADS FROM DISKETTE
FAPWAT   DEFxx    s,2            NUMBER OF WAITS FOR BUFFER SPACE
```

- **DEFFDK**

```
* * *     'FORMDKT' INSTRUCTION PARAMETER LIST DEFINITION

FDKPAR   DEFxx    s,9            'FORMDKT' PARAMETER LIST

FDKLNG   DEFxx     FDKPAR,1      DISKETTE RECORD LENGTH :
FDKLN0   EQUATE    X'00'                 128 BYTE RECORD LENGTH VALUE
FDKLN1   EQUATE    X'01'                 256 BYTE RECORD LENGTH VALUE
FDKLNF   EQUATE    X'FF'                 DEFAULT TO MOUNTED DSK REC LEN

FDKTYP   DEFxx    s,1            DISKETTE TYPE :
FDKTY1   EQUATE    X'01'                 DISKETTE 1 VALUE
FDKTY2   EQUATE    X'02'                 DISKETTE 2 VALUE
FDKTY2D  EQUATE    X'03'                 RESERVED
FDKTYF   EQUATE    X'FF'                 DEFAULT TO MOUNTED DSK TYPE

FDKRES   DEFxx    s,1            RESERVED

FDKVID   DEFxx    s,6            DISKETTE VOLUME IDENTIFIER
```

- **DEFGMS**

```
GMSTMR   DEFLD     15,0,4          GLOBAL SEGMENT 15 TIMER
GMSHTM   DEFLD     15,0,2          HIGH ORDER 2 BYTES OF TIMER
GMSLTM   DEFLD     15,,2           LOW ORDER 2 BYTES OF TIMER
GMSWTM   DEFLD     15,,4           CONTROLLER WAIT TIME (IN SECONDS)
GMSHRT   DEFLD     15,,2           HIGH RESOLUTION TIMER

GMSMFS   DEFLD     15,,2           MACHINE FEATURE SWITCHES:
GMSMTDM  EQUATE    X'2000'           TWO DISKETTE ADAPTERS
GMSMDCM  EQUATE    X'1000'           DCA ADAPTER
GMSMBLM  EQUATE    X'0700'           MASK TO EXTRACT NUM OF B-LOOPS
GMSMHLM  EQUATE    X'00F0'           MASK TO EXTRACT HOST LINK TYPE:
GMSMH0M  EQUATE    X'0000'             NO HOST LINK
GMSMH1M  EQUATE    X'0010'             HPCA - X.21 SWITCHED
GMSMH2M  EQUATE    X'0020'             HPCA - EIA
GMSMH3M  EQUATE    X'0030'             HPCA - MULTI USE LOOP
GMSMH5M  EQUATE    X'0050'             CCA  - EIA
GMSMH6M  EQUATE    X'0060'             HPCA - X.25 EIA
GMSMH7M  EQUATE    X'0070'             HPCA - X.21 LEASED
GMSMH8M  EQUATE    X'0080'             HPCA - X.25 X.21

GMSDSM   DEFLD     15,,1           RESERVED FOR SYSTEM USE
GMSMSZ   DEFLD     15,,1           NUMBER OF 64K SECTIONS OF MEMORY

GMSFTR   DEFLD     15,,2           OPTIONAL MODULES LOADED FLAGS:
GMSFEDM  EQUATE    X'8000'           ENCODE/DECODE (AET)
GMSFSMM  EQUATE    X'2000'           LSORT/LMERGE
GMSFSDM  EQUATE    X'1000'           SETDSKT
GMSF6AM  EQUATE    X'0800'           STARTER ADAPTER DIAGS MASK
GMSFSLM  EQUATE    X'0400'           ALTERNATE LINE MASK
GMSF68M  EQUATE    X'0200'           SCALE/SETX.../TESTX
GMSFDEM  EQUATE    X'0100'           ENCODE/DECODE (DES)
GMSFMTM  EQUATE    X'0080'           DISKETTE MULTI-BLOCK I/O MASK
GMSF20M  EQUATE    X'0040'           THIS FLAG IS ALWAYS SET, AS
                                     LCHAP IS IN THE BASE

GMSF25M  EQUATE    X'0020'           STATS
GMSF21M  EQUATE    X'0010'           LTRT
GMSF6DM  EQUATE    X'0040'           LEFT FOR IR COMPATIBILITY
GMSF26M  EQUATE    X'0008'           DECOMP/DECOMPTB
GMSF70M  EQUATE    X'0004'           DATA STREAM MANAGEMENT
GMSF24M  EQUATE    X'0002'           LSEEKP
GMSF27M  EQUATE    X'0001'           COMP/COMPTB

GMSFT2   DEFLD     15,,2           OPTIONAL MODULES LOADED FLAGS:
GMSF2AM  EQUATE    X'8000'           SCRPAD
GMSF2CM  EQUATE    X'2000'           INITMR
GMSF2DM  EQUATE    X'1000'           THIS FLAG IS ALWAYS SET, AS
*                                    LLOAD WITH PARM=EXP IS IN BASE
GMSF28M  EQUATE    X'0800'           SECURITY INSTRUCTIONS
GMSF31M  EQUATE    X'0400'           ZONED DECIMAL INSTRUCTIONS
GMSF32M  EQUATE    X'0200'           LTIMEV INSTRUCTION
GMSF40M  EQUATE    X'0100'           FORMDKT
GMSF41M  EQUATE    X'0080'           COMPDKT
GMSF29M  EQUATE    X'0040'           TRANSIENT APCALL/APRETURN
GMSF34M  EQUATE    X'0020'           LCONVERT/CRETN
GMSF36M  EQUATE    X'0010'           DTACCESS/SEGALLOC
GMSF42M  EQUATE    X'0008'           DPOOL
GMSF2FM  EQUATE    X'0004'           FORMDSK
GMSFB7M  EQUATE    X'0002'           DISK MULTI-BLOCK I/O MASK
         DEFLD     15,,2           RESERVED

GMSCUA   DEFLD     15,,1           CONTROL UNIT ADDRESS
GMSLID   DEFLD     15,,1           COMMUNICATION LINK ID:
GMSL48M  EQUATE    X'01'             4800 BPS
GMSL96M  EQUATE    X'02'             9600 BPS
GMSL02M  EQUATE    X'02'             4502 - SNA/SDLC
GMSL05M  EQUATE    X'05'             1422 - BSC
GMSL07M  EQUATE    X'07'             5656 - X.21
GMSL08M  EQUATE    X'08'             4850 - MULTI-USE
GMSL09M  EQUATE    X'09'             8V0134 - X.25
```

```
GMSRES1   DEFLD     15,,2          RESERVED
GMSBLN    DEFLD     15,,2          XID BLOCK NUMBER
GMSMOD    DEFLD     15,,1          INPUT MSG ROUTING CONTROL
GMSDEF    DEFLD     15,,1          DEFAULT STB ID IF DEFAULT USE
GMSSDI    DEFLD     15,GMSDEF,1    CLEAR STB ID DISP IN MSG
GMSTAB    DEFLD     15,,2          POINTER TO ROUTING TABLE
GMSSPR    DEFLD     15,,1          RESERVED
GMSLOP    DEFLD     15,,1          SUMMARY LOOP OUTAGE MAP

GMSIND    DEFLD     15,,1          GLOBAL INDICATOR BYTE:
GMSILDM   EQUATE    X'80'            LOSS OF CONTACT MASK
GMSILRM   EQUATE    X'40'            LINK DOWN MASK
GMSIWSM   EQUATE    X'20'            WARM START MASK
GMSIDSM   EQUATE    X'10'            2-SIDED DISKETTE ON PRIMARY DRIVE
GMSISMM   EQUATE    X'04'            SYS MONITOR LOGGED ON AT 3604
GMSIDCM   EQUATE    X'02'            DISK CREATE IN PROGRESS FLAG
GMSIMAM   EQUATE    X'01'            MSG REQ ACTION WRITTEN TO LOG

GMSFLG    DEFLD     15,,1          FLAG BYTE:
GMSFPCM   EQUATE    X'80'            MON PROC CMD IN PROG I/P MODE
GMSID2M   EQUATE    X'40'            2-SIDED DISKETTE ON SECONDARY DRIVE
GMSD2PM   EQUATE    X'20'            SECONDARY DISK DRIVE PRESENT
GMSCSCM   EQUATE    X'10'            CNM SPECIFIED IN CONFIGURATION
GMSAPCM   EQUATE    X'08'            AP'S TO DISK REQ'TD IN CONFIGURATION

          DEFLD     15,,2          RESERVED
GMSBSN    DEFLD     15,,2          CURRENT TF BLOCK NUMBER ON IPL DRIVE
          DEFLD     15,,2          RESERVED
GMSBSA    DEFLD     15,,2          CURRENT TF BLOCK NUMBER ON 2ND DRIVE
GMSBSD    DEFLD     15,,4          CURRENT TF BLOCK NUMBER ON 1ST DISK
GMSBSD2   DEFLD     15,,4          CURRENT TF BLOCK NUMBER ON 2ND DISK
GMSPRI    DEFLD     15,,1          PRIORITY DISPATCH (FLAG AND TABLE NUM)

GMSTYP    DEFLD     15,,1          DISKETTE TYPE MOUNTED FLAGS
GMSITYM   EQUATE    X'F0'            MASK TO EXTRACT IPL DRIVE FLAGS
GMSID1M   EQUATE    X'10'              TYPE 1  DKT  ON IPL DRIVE
GMSIDT2M  EQUATE    X'20'              TYPE 2  DKT  ON IPL DRIVE
GMSID2DM  EQUATE    X'30'              TYPE 2D DKT  ON IPL DRIVE

GMSSTYM   EQUATE    X'0F'            MASK TO EXTRACT 2ND DRIVE FLAGS
GMSST1M   EQUATE    X'01'              TYPE 1  DKT  ON 2ND DRIVE
GMSST2M   EQUATE    X'02'              TYPE 2  DKT  ON 2ND DRIVE
GMSST2DM  EQUATE    X'03'              TYPE 2D DKT  ON 2ND DRIVE

GMSSID    DEFLD     15,,2          SESSION ID

GMSDKTP   DEFLD     15,,1          DISK DRIVE FLAGS
GMSDSK1   EQUATE    X'F0'          1ST DISK DRIVE CAPACITY MASK
GMSD1LM   EQUATE    X'20'          1ST DISK DRIVE IS LARGE CAPACITY
GMSD1SM   EQUATE    X'10'          1ST DISK DRIVE IS SMALL CAPACITY
GMSDSK2   EQUATE    X'0F'          2ND DISK DRIVE CAPACITY MASK
GMSD2LM   EQUATE    X'02'          2ND DISK DRIVE IS LARGE CAPACITY
          DEFLD     15,,1          RESERVED
GMSCSN    DEFLD     15,,3          CONTROLLER SERIAL NUMBER

GMSMFS3   DEFLD     15,,1          ALA LINE
GMSALIM   EQUATE    X'F0'          MASK TO EXTRACT ALA LINE :
GMSM1SM   EQUATE    X'20'          ALA  - SNAP (HPCA) ADAPTER
GMSM1AM   EQUATE    X'40'          ALA  - START/STOP (CCA) ADAPTER

GMSMFS4   DEFLD     15,,1          RESERVED

GMSMFS5   DEFLD     15,,1          RESERVED

GMSMFS6   DEFLD     15,,1          RESERVED

GMSIND2   DEFLD     15,,1         GLOBAL INDICATOR BYTE 2:
GMSILD2M  EQUATE     X'80'          LINK2 LOSS-OF-CONTACT MASK
GMSILR2M  EQUATE     X'40'          LINK2 LINK DOWN MASK
GMSCUA2   DEFLD     15,,1          LINK2 CONTROL UNIT ADDRESS
GMSLID2   DEFLD     15,,1          LINK2 COMMUNICATIONS LINK ID

GMSRES    DEFLD     15,,12         RESERVED

GMSFUF    DEFLD     15,,0          FIRST USER FIELD
```

- **DEFINP**

```
INPPAR    DEFxx    s,,35         INSTRUCTION PARAMETER LIST
INPINTYP  DEFxx    s,INPPAR,1    INPUT PIN FORMAT TYPE
INPCLEAR  EQUATE     X'00'               CLEAR PIN FORMAT
INPEPPAD  EQUATE     X'80'               ENCRYPTING PIN PAD PIN FORMAT
INP3624   EQUATE     X'40'               3624 PIN FORMAT
INPANSI   EQUATE     X'20'               ANSI X9.8 STANDARD PIN FORMAT
INPRPQ    EQUATE     X'10'               RESERVED FOR RPQ 7B0570
INPINPAD  DEFxx    s,,1          INPUT PIN PAD CHARACTER
INPINKEY  DEFxx    s,,8          INPUT PIN KEY (ENCRYPTED)
INPAN     DEFxx    s,,8          INPUT PRIMARY ACCOUNT NUMBER
INPAND    DEFxx    s,INPAN,2      4 DIGITS (2 BYTE) OF ZEROES
INPANA    DEFxx    s,,6           12 DIGIT (6 BYTE) ACCOUNT NUM
INPINLEN  DEFxx    s,,1          INPUT CLEAR PIN LENGTH
INPIN     DEFxx    s,,16         INPUT PIN
```

- **DEFINT**

```
INTIMR    DEFxx    s,16          INTERVAL TIMER PARMLIST
INTSID    DEFxx     INTTMR,1     WORKSTATION NUMBER
INTTNO    DEFxx    s,1           INTERVAL TIMER NUMBER
INTREQ    DEFxx    s,1           REQUEST TYPE

INTSTRTM EQUATE X'00'                REQUEST IS START TIMER
INTSTOPM EQUATE X'01'                REQUEST IS STOP TIMER
INTSTPRM EQUATE X'11'                REQUEST IS STOP & RETURN INTERVAL
INTREADM EQUATE X'02'                REQUEST IS READ RESULTS
INTRDRSM EQUATE X'03'                REQUEST IS READ & RESET
INTACTVM EQUATE X'04'                REQUEST IS ACTIVATE TIMING
INTDACTM EQUATE X'05'                REQUEST IS DEACTIVATE TIMING
```

```
*
*     THIS ENDS THE BASIC PORTION OF INTERVAL TIMER PARAMETER LIST
*
*
*     THE FOLLOWING FIELD USED WITH READ, READ & RESET, AND
*         STOP & REPORT RESULTS REQUESTS
*
INTFLG   DEFxx   s,1           FLAG BYTE
         LSPACE
INTDAVTM EQUATE X'80'               TIMER IS DEACTIVATED FLAG
INTSTARM EQUATE X'40'               TIMER IS CURRENTLY RUNNING

*
*     THE FOLLOWING FIELD USED WITH STOP & REPORT RESULTS REQUEST
*
INTLTH   DEFxx   s,3           LENGTH OF INTERVAL
*
*     THE FOLLOWING FIELDS USED WITH READ AND READ & RESET REQUESTS
*
INTMIN   DEFxx    INTLTH,3     SHORTEST INTERVAL
INTMAX   DEFxx   s,3           LONGEST INTERVAL
INTTOT   DEFxx   s,4           SUM OF ALL INTERVALS
INTCNT   DEFxx   s,2           NUMBER OF INTERVALS
```

- **DEFKYG**

```
KYGPAR   DEFxx   s,,9          INSTRUCTION PARAMETER LIST
KYGFLAG  DEFxx   s,KYGPAR,1    GENERATED KEY TYPE FLAG
KYGCLR   EQUATE    X'00'              PLAINTEXT KEY
KYGENC   EQUATE    X'80'              ENCIPHERED KEY
KYGBUFF  DEFxx   s,,8           BUFFER FOR GENERATED KEY
```

- **DEFMER**

```
MERPAR   DEFxx   s,30        MERGE PARAMETER LIST
MERIFB   DEFxx   MERPAR,1    INPUT FLAG BYTE
MERIF0M  EQUATE X'80'             BIT ON ---> DESCENDING KEYS
MERIF6M  EQUATE X'02'             BIT ON ---> INPUT BLOCK 1 NULL
MERIF7M  EQUATE X'01'             BIT ON ---> INPUT BLOCK 2 NULL
MERRFB   DEFxx   s,1         RETURN FLAG BYTE
MERRF0M  EQUATE X'80'             OUTPUT BLOCK FULL
MERRF1M  EQUATE X'40'             INPUT BLOCK 1 EMPTY
MERRF2M  EQUATE X'20'             INPUT BLOCK 2 EMPTY
MERRF3M  EQUATE X'10'             SEQ CHK NOT ON MERGE UNIT BOUNDARY
MERRF4M  EQUATE X'08'             SEQ CHK ON MERGE UNIT BOUNDARY
MERRF5M  EQUATE X'04'             BOTH INPUT BLOCKS NULL
MERI1B   DEFxx   s,2         INPUT BLOCK 1 BEGIN DISP
MERI1E   DEFxx   s,2         INPUT BLOCK 1 END DISP
MERI2B   DEFxx   s,2         INPUT BLOCK 2 BEGIN DISP
MERI2E   DEFxx   s,2         INPUT BLOCK 2 END DISP
MERDLN   DEFxx   s,2         DATA ITEM LENGTH
MERKYD   DEFxx   s,2         DISP TO KEY IN DATA ITEM
MERKYL   DEFxx   s,1         KEY LENGTH
MERIS1   DEFxx   s,1         INPUT BLOCK 1 SEGMENT NUMBER
MERIS2   DEFxx   s,1         INPUT BLOCK 2 SEGMENT NUMBER
MEROTS   DEFxx   s,1         OUTPUT BLOCK SEGMENT NUMBER
MEROTB   DEFxx   s,2         OUTPUT BLOCK BEGIN DISP
MEROTE   DEFxx   s,2         OUTPUT BLOCK END DISP
MERI1C   DEFxx   s,2         INPUT BLOCK 1 CURRENT DISPLACEMENT
MERI2C   DEFxx   s,2         INPUT BLOCK 2 CURRENT DISPLACEMENT
MEROTC   DEFxx   s,2         OUTPUT BLOCK CURRENT DISPLACEMENT
MERUNT   DEFxx   s,2         MERGE UNIT SIZE
```

- **DEFREG**

```
* * *    REGISTER EQUATES

R00      EQUATE    0
R01      EQUATE    1
R02      EQUATE    2
R03      EQUATE    3
R04      EQUATE    4
R05      EQUATE    5
R06      EQUATE    6
R07      EQUATE    7
R08      EQUATE    8
R09      EQUATE    9
R10      EQUATE   10
R11      EQUATE   11
R12      EQUATE   12
R13      EQUATE   13
R14      EQUATE   14
R15      EQUATE   15
```

- **DEFRGS**

```
* * *    REGISTER SECTION OF SEGMENT 0

REGS     DEFLD    0,0,96
REG0     DEFLD    0,0,6
REG1     DEFLD    0,,6
REG2     DEFLD    0,,6
REG3     DEFLD    0,,6
REG4     DEFLD    0,,6
REG5     DEFLD    0,,6
REG6     DEFLD    0,,6
REG7     DEFLD    0,,6
REG8     DEFLD    0,,6
REG9     DEFLD    0,,6
REG10    DEFLD    0,,6
REG11    DEFLD    0,,6
REG12    DEFLD    0,,6
REG13    DEFLD    0,,6
REG14    DEFLD    0,,6
REG15    DEFLD    0,,6
```

- **DEFRMK**

```
RMKPAR   DEFxx   s,,18         INSTRUCTION PARAMETER LIST
RMKFLAG  DEFxx   s,RMKPAR,1    RE-ENCIPHER KEY FLAG
RMKNVM   EQUATE    X'00'              USE NVM KEY
RMKUSER  EQUATE    X'80'              USER-SUPPLIED KEY
         DEFxx   s,,1          RESERVED
RMKINKEY DEFxx   s,,8          INPUT KEY (TO BE RE-ENCIPHERED)
RMKXDKEY DEFxx   s,,8          USER CROSS-DOMAIN KEY
```

- **DEFSCA**

```
SCAPAR   DEFxx   s,18          SCALE PARAMETER LIST
SCALEN   DEFxx    SCAPAR,1     LENGTH OF OUTPUT AREA
SCACHR   DEFxx   s,1           SCALE CHARACTER
SCAFAC   DEFxx   s,1           SCALE FACTOR
SCAINP   DEFxx   s,1           INPUT FLAG BYTE
SCAHDR   DEFxx   s,6           SPECIAL HEADER CHARACTERS
SCADEL   DEFxx   s,4           DELETE CHARACTERS
SCARES   DEFxx   s,3           RESERVED (VALUE MUST BE 3X'00')
SCASIG   DEFxx   s,1           NUMBER OF SIGNIFICANT DIGITS
SCABEG   DEFxx   s,0           BEGINNING OF OUTPUT AREA
```

- **DEFSCP**

```
SCPSTR   DEFxx   s,0    START OF PARAMETER LIST

SCPFC    DEFxx   s,1      FUNCTION REQUESTED

SCPINTR  EQUATE X'01'          INITIALIZE SPA REQUEST
SCPADDR  EQUATE X'02'          ADD ELEMENT REQUEST
SCPRPLR  EQUATE X'03'          REPLACE ELEMENT REQUEST
SCPRTRR  EQUATE X'04'          RETRIEVE ELEMENT REQUEST
SCPRTUR  EQUATE X'05'          RETRIEVE ELEMENT FOR UPDATE REQUEST
SCPDLER  EQUATE X'06'          DELETE AN ELEMENT REQUEST
SCPDLAR  EQUATE X'07'          DELETE ALL ELEMENTS REQUEST
SCPARUR  EQUATE X'25'          ADD OR RETRIEVE FOR UPDATE REQUEST
SCPARLR  EQUATE X'23'          ADD OR REPLACE REQUEST
SCPEXCR  EQUATE X'08'          EXCHANGE DATA AREA WITH THE ELEMENT

SCPTYP   DEFxx   s,1      ADDRESSING TYPE

SCPKEYM  EQUATE X'00'          KEYED ADDRESSING
SCPELMA  EQUATE X'01'          ELEMENT ADDRESSING

SCPSPID  DEFxx   s,1      SPA ID
SCPRC    DEFxx   s,1      SCRATCH PAD RETURN CODE

SCPRC80  EQUATE X'80'
SCPRC40  EQUATE X'40'
SCPRC20  EQUATE X'20'
SCPRC10  EQUATE X'10'
SCPRC08  EQUATE X'08'
SCPRC04  EQUATE X'04'
SCPRC02  EQUATE X'02'
SCPRC01  EQUATE X'01'
SCPRC00  EQUATE X'00'

SCPELMN  DEFxx   s,2      ELEMENT NUMBER ADDRESS
SCPKEYL  DEFxx   s,1      KEY LENGTH
SCPDATPT DEFxx   s,3      DATA AREA ADDRESS (SEGMENT,DISPLACEMENT)

SCPELMLN DEFXX    SCPELMN,2 ELEMENT LENGTH
SCPELMNB DEFxx   s,2      NUMBER OF ELEMENTS ALLOCATED

SCPDSG   DEFXX    SCPDATPT,1
SCPDDSP  DEFxx   s,2
SCPEND   DEFxx   s,0    END OF PARAMETER LIST
SCPPRL   DEFXX    SCPSTR,(D:SCPEND-D:SCPSTR)
```

- **DEFSEG**

```
* * *    SEGMENT EQUATES

S00      EQUATE    0
S01      EQUATE    1
S02      EQUATE    2
S03      EQUATE    3
S04      EQUATE    4
S05      EQUATE    5
S06      EQUATE    6
S07      EQUATE    7
S08      EQUATE    8
S09      EQUATE    9
S10      EQUATE   10
S11      EQUATE   11
S12      EQUATE   12
S13      EQUATE   13
S14      EQUATE   14
S15      EQUATE   15
```

- **DEFSKP**

```
SKPPAR    DEFxx   s,18              LSEEKP PARAMETER LIST
SKPFLG1   DEFxx    SKPPAR,1         LSEEKP OPTION FLAG
SKPFPA    EQUATE X'80'                  TABLE/PARM LIST MATCH ADDRESS
*                                        = 0 USE TABLE MATCH ADDRESS
*                                        = 1 USE PARM LIST MATCH ADR
SKPFBS    EQUATE X'40'                  SEQUENT/BINARY TABLE SEARCH
*                                        = 0 SEQUENTIAL TABLE SEARCH
*                                        = 1 BINARY TABLE SEARCH
SKPFTI    EQUATE X'20'                  TABLE LOCATION
*                                        = 0 IN NON-SPLIT AP OR
*                                            CONSTANT PORTION OF SPLIT
*                                        = 1 IN INSTRUCTION PORTION
*                                            OF A SPLIT AP
SKPFCE    EQUATE X'10'                  NO COPY/COPY TABLE ELEMENT
*                                        = 0 DO NOT COPY TABLE ELEMNT
*                                        = 1 COPY TABLE ELEMENT
SKPFEQ    EQUATE X'08'                  LOCATE EQUAL TABLE ENTRY
*                                        (REQUIRED SETTING = 1)
*                                        = 1 RETURN EQUAL TABLE ENTRY
SKPFBL    EQUATE X'04'                  BINARY RETURN < IF NOT EQUAL
*                                        = 0 DO NOT RETURN LESS THAN
*                                        = 1 RETURN IF < IF NOT EQUAL
SKPFBG    EQUATE X'02'                  BINARY RETURN > IF NOT EQUAL
*                                        = 0 DO NOT RETURN GRTR THAN
```

```
                                                = 1 RETURN GRTR THAN IF NO

SKPFLG2   DEFxx   s,1                  LSEEKP OPTION FLAG BYTE 2
SKPFNS    EQUATE X'80'                      DO NOT/ DO RETURN NSI
*                                             = 0 DO NOT RETURN NSI
*                                             = 1 RETURN NSI
SKPREG    DEFxx   s,1                  RETURN NSI IN STACK/REGISTER
*                                             = 0 RETURN NSI IN STACK
*                                             > 0 REGISTER TO CONTAIN NSI
SKPARS    DEFxx   s,1                  SEGMENT CONTAINING ARGUMENT
SKPARD    DEFxx   s,2                  DISPLACEMENT TO ARGUMENT
SKPARL    DEFxx   s,2                  LENGTH OF ARGUMENT
SKPSCS    DEFxx   s,1                  TABLE SEGMENT
SKPSCD    DEFxx   s,2                  DISPLACEMENT TO TABLE
SKPMTH    DEFxx   s,2                  MATCH ADDRESS
SKPENS    DEFxx   s,1                  SEGMENT CONTAINING COPY FIELD
SKPEND    DEFxx   s,2                  DISPLACEMENT TO COPY FIELD
SKPENL    DEFxx   s,2                  LENGTH OF COPY FIELD


• DEFSMS

* * *    SEGMENT ONE MACHINE SECTION DEFINITIONS

*          GENERAL FIELDS AND EQUATES SUBSECTION:

SMSPSW    DEFLD     1,0,4        PSW
SMSSID    DEFLD     1,0,1        STATION ID
SMSCCD    DEFLD     1,,1         COMPLETION CODE
SMSUIC    DEFLD     1,,2         RELATIVE INSTRUCTION COUNTER
SMSPCA    DEFLD     1,,2         ADDRESS OF INSTRUCTION THAT CAUSED PC
SMSPCC    DEFLD     1,,1         PROGRAM CHECK CODE
SMSABK    DEFLD     1,,1         OPB A/B INDICATION
SMSIML    DEFLD     1,,2         I/P MESSAGE LENGTH
SMSICT    DEFLD     1,SMSIML,2   COUNT OF IMPLIED DATA SET OPENS
SMSDST    DEFLD     1,,2         DEVICE STATUS BYTES
SMSDS1    DEFLD     1,SMSDST,1   STATUS FIELD 1
SMSDS2    DEFLD     1,,1         STATUS FIELD 2
SMSLTC    DEFLD     1,,2         LOOP THRESHOLD COUNT
SMSLTH    DEFLD     1,,2         LOOP THRESHOLD VALUE

SMSIND    DEFLD     1,,1         INDICATOR BYTE:
SMSDATSM  EQUATE    X'10'          STATION CONFIGURED FOR DATSM USE
                                     MASK
SMSICAM   EQUATE    X'40'          CANCEL MASK
SMSICAO   EQUATE    X'BF'          TURN OFF CANCEL FLAG MASK

SMSDCB    DEFLD     1,,1         DELIMITER CONTROL BYTE
SMSDEL    DEFLD     1,,2         ALTERNATE DELIMITER TABLE ADDR
SMSDSS    DEFLD     1,,1         SWITCHED MSG STB ID FIELD

SMSAFL    DEFLD     1,,1         ASYNCHRONOUS FLAG FIELD:
SMSACP    DEFLD     1,SMSAFL,1     ASYNCHRONOUS CPU INTERRUPT FLAG
SMSACPM   EQUATE    X'80'          ASYNCHRONOUS CPU INTERRUPT MASK
SMSAST    DEFLD     1,SMSAFL,1     ASYNCHRONOUS STATION INTERRUPT FLAG
SMSASTM   EQUATE    X'40'          ASYNCHRONOUS STATION INTERRUPT MASK
SMSAAP    DEFLD     1,SMSAFL,1     ASYNCHRONOUS ALA INTERRUPT FLAG
SMSAAPM   EQUATE    X'20'          ASYNCHRONOUS ALA INTERRUPT MASK

SMSTMR    DEFLD     1,,4         SMS TIMER FIELD
SMSHTM    DEFLD     1,SMSTMR,2   HIGH ORDER 2 BYTES OF TIMER
SMSLTM    DEFLD     1,,2         LOW ORDER 2 BYTES OF TIMER
SMSPCT    DEFLD     1,,2         PAUSE INSTRUCTION COUNTER

SMSFG2    DEFLD     1,,1         FLAG BYTE:
SMSPCRM   EQUATE    X'80'          PROGRAM CHECK ROUTINE IN CONTROL
SMSLWSM   EQUATE    X'40'          LOGICAL WAIT STATE

SMSLSB    DEFLD     1,,1         PARENT AP LINK STACK USE COUNT
SMSWAIT   DEFLD     1,,1         WAIT INSTRUCTION TERMINATING CONDITION
SMSICPM   EQUATE    X'10'          CPU MESSAGE
SMSIAPM   EQUATE    X'20'          ALA MESSAGE
SMSITPM   EQUATE    X'30'          TERMINAL MESSAGE
SMSISPM   EQUATE    X'40'          SWITCHED MESSAGE
SMSIPPM   EQUATE    X'50'          PROGRAM INTERRUPT VIA POST
                                     INSTRUCTION
SMSITFM   EQUATE    X'60'          TIMER INTERRUPT
SMSIATM   EQUATE    X'70'          ATTENTION INTERRUPT

SMSDRG    DEFLD     1,,1         REGISTER FOR DELIMITER TABLE ADDRESS
          DEFLD     1,,2         RESERVED

*          TERMINAL FIELDS AND EQUATES SUBSECTION:

SMSEID    DEFLD     1,,1         EOM IDENTIFIER
SMSECT    DEFLD     1,,1         NUMBER OF CHARS IN THE CURRENT EOM
                                     STRING
SMSTGU    DEFLD     1,,1         ADDRESS OF TERMINAL GROUP UNIT
SMSSSU    DEFLD     1,SMSTGU,1   ADDRESS SHARED SLOT UNIT (SUB-ADDR)

SMSSAM    DEFLD     1,,1         LDA ATTENTION SUMMARY MASK:
SMSLDA0   EQUATE    X'80'          MESSAGE AVAILABLE ON LDA-0
SMSLDA1   EQUATE    X'40'          MESSAGE AVAILABLE ON LDA-1
SMSLDA2   EQUATE    X'20'          MESSAGE AVAILABLE ON LDA-2
SMSLDA3   EQUATE    X'10'          MESSAGE AVAILABLE ON LDA-3
SMSLDA4   EQUATE    X'08'          MESSAGE AVAILABLE ON LDA-4
SMSLDA5   EQUATE    X'04'          MESSAGE AVAILABLE ON LDA-5
SMSLDA6   EQUATE    X'02'          MESSAGE AVAILABLE ON LDA-6
SMSLDA7   EQUATE    X'01'          MESSAGE AVAILABLE ON LDA-7

SMSSPR    DEFLD     1,,2         RESERVED
SMSCUR    DEFLD     1,SMSSPR,2   LOCAL KEY-TRACKING CURSOR ADDR
SMSMSL    DEFLD     1,,1         LENGTH OF MAGNETIC STRIPE DATA
SMSKSM    DEFLD     1,,1         LDA FIRST KEYSTROKE SUMMARY MASK:

*          DISKETTE FIELD AND EQUATES SUBSECTION:

SMSUNK    DEFLD     1,,2         UNIQUE ID FOR UNKEYED DATA SET
SMSFG1    DEFLD     1,,1         FLAG BYTE:
SMSMBTM   EQUATE    X'40'          MULTI-BLOCK I/O
SMSNXRM   EQUATE    X'20'          AVOID RECORD TRANSFER
SMSDSKM   EQUATE    X'01'          DISK DRIVE
SMSDT2M   EQUATE    X'02'          ALTERNATE DRIVE - DISK OR DISKETTE
SMSDKTI   EQUATE    X'03'          MASK TO ISOLATE DRIVE SELECT BITS

SMSDID    DEFLD     1,,1         DISK DATA SET ID
SMSRNM1   DEFLD     1,,2         1ST HALFWORD OF RECORD SEQUENCE NUMBER
SMSRSN    DEFLD     1,,2         RCD SEQ NUMBER FOR DISKETTE READ/REPLACE
SMSRSN1   DEFLD     1,SMSRSN,1   1ST BYTE OF RCD SEQ NUMBER
SMSRSN2   DEFLD     1,,1         2ND BYTE OF RCD SEQ NUMBER
```

```
SMSRPS    DEFLD     1,SMSRNM1,4 FOUR BYTE RCD SEQ NUMBER
SMSRSNH   DEFLD     1,SMSRPS,2 1ST HWD OF RCD SEQ NUMBER
SMSRSNL   DEFLD     1,,2       2ND HWD OF RCD SEQ NUMBER

SMSSFW    DEFLD     1,,1       SUB-FILE ID FOR DISKETTE WRITE
SMSSFR    DEFLD     1,,1       SUB-FILE ID FOR DISKETTE READ/REPLACE
SMSKEY    DEFLD     1,,4       CURRENT POSITION OF KEYED DATA SET
SMSKEY1   DEFLD     1,SMSKEY,2 FIRST  HWD OF 4 BYTE KEY
SMSKEY2   DEFLD     1,,2       SECOND HWD OF 4 BYTE KEY
          DEFLD     1,,2       RESERVED
SMSFSN    DEFLD     1,,2       FILE SEQ NUMBER FROM LAST DISKETTE WRITE
          DEFLD     1,,2       RESERVED
SMSSSN    DEFLD     1,,2       SUB-FILE SEQ NUM FROM LAST DISKETTE WRITE
          DEFLD     1,,2       RESERVED
SMSCSN    DEFLD     1,,2      COMPOSITE SEG NUM FROM LAST DISKETTE WRITE
SMSADS1   DEFLD     1,,2       (1ST HWD OF 4 BYTE D. S. BLKS AVAILABLE)
SMSADS    DEFLD     1,,2       NUM OF D.S. BLKS AVAIL FOR LWRITE

*           CPU FIELDS AND EQUATES SUBSECTION:

SMSCRC    DEFLD     1,,2       CPU READ CONTROL FIELDS
SMSCRF    DEFLD     1,SMSCRC,1 CPU READ FLAGS
SMSBRL    DEFLD     1,SMSCRC,1 BSC READ CONTROL
SMSCRT    DEFLD     1,,1       CPU READ TYPE
SMSCST    DEFLD     1,,1       LINK STATUS BYTE
SMSCCR    EQUATE    X'04'      OPERATIONAL CIRCUIT FLAG
SMSCID    DEFLD     1,SMSCST,1 S/S UNIT ID
SMSTR2    DEFLD     1,,1       RESERVED
SMSCWC    DEFLD     1,,2       CPU WRITE CONTROL FIELDS
SMSCWF    DEFLD     1,SMSCWC,1 CPU WRITE FLAGS
SMSBWC    DEFLD     1,SMSCWC,1 BSC WRITE CONTROL
SMSCWT    DEFLD     1,,1       CPU WRITE TYPE
SMSCRS    DEFLD     1,,2       READ SEQ NUMBER OR ID
SMSBIH    DEFLD     1,SMSCRS,2 INPUT HEADER LENGTH
SMSCWS    DEFLD     1,,2       WRITE SEQ NUMBER OR ID
SMSBOH    DEFLD     1,SMSCWS,2 OUTPUT HEADER LENGTH
SMSCSR    DEFLD     1,,2       READ RESPONSE TO DATA SEQ NUMBER
SMSCHL    DEFLD     1,SMSCSR,2 HEADER LENGTH
SMSCPE    DEFLD     1,,2       CPU READ/WRITE FLAGS EXTENSION
SMSCRE    DEFLD     1,SMSCPE,1 CPU READ FLAGS EXTENSION
SMSCWE    DEFLD     1,,1       CPU WRITE FLAGS EXTENSION
          DEFLD     1,,4       RESERVED
          DEFLD     1,,1       RESERVED
SMSAMS    DEFLD     1,,3       ALA MACHINE SEGMENT, SEG NO./DISPLACE
                                   MENT
SMSAMS1   DEFLD     1,SMSAMS,1 AMS SEGMENT NUMBER
SMSAMS2   DEFLD     1,,2       AMS SEGMENT DISPLACEMENT

*           LINKSTACK FIELDS SUBSECTION:

SMSLSH    DEFLD     1,,2       LINK STACK HEADER
SMSLSM    DEFLD     1,SMSLSH,1 MAXIMUM NUMBER OF ENTRIES
SMSLSE    DEFLD     1,,1       CURRENT NUMBER OF ENTRIES
*                               SEE SMSLSB FOR BASE COUNT OF STACK
SMSLSA    DEFLD     1,,12      LINK STACK AREA
```

- **DEFSOR**

```
SORPAR    DEFxx     s,17       SORT PARAMETER LIST
SORIFB    DEFxx     SORPAR,1   INPUT FLAG BYTE
SORIFOM   EQUATE X'80'             BIT ON ---> DESCENDING KEYS
SORRFB    DEFxx     s,1        RESERVED
SORSBG    DEFxx     s,2        SORT BLOCK BEGIN
SORSND    DEFxx     s,2        SORT BLOCK END
SORWBG    DEFxx     s,2        WORK BLOCK BEGIN
SORWND    DEFxx     s,2        WORK BLOCK END
SORDLN    DEFxx     s,2        DATA ITEM LENGTH
SORKYD    DEFxx     s,2        DISPLACEMENT TO KEY IN DATA ITEM
SORKYL    DEFxx     s,1        KEY LENGTH
SORSSG    DEFxx     s,1        SEGMENT NUMBER CONTAINING SORT BLOCK
SORWSG    DEFxx     s,1        SEGMENT NUMBER CONTAINING WORK BLOCK
```

- **DEFTNP**

```
TNPPAR    DEFxx     s,,20          INSTRUCTION PARAMETER LIST
TNPINFLG  DEFxx     s,TNPPAR,1     TRANSLATE PIN FLAG
TNPEPPAD  EQUATE      X'80'                ENCRYPTING PIN PAD FORMAT
TNP3624   EQUATE      X'40'                3624 PIN FORMAT
TNPANSI   EQUATE      X'20'                ANSI PIN FORMAT
TNPRPQ    EQUATE      X'10'                RESERVED FOR RPQ 7B0570
TNPSAMEK  EQUATE      X'01'                USE INPUT PIN KEY INDICATOR
TNPINPAD  DEFxx     s,,1           OUTPUT PIN KEYPAD CHARACTER
TNPINSEQ  DEFxx     s,,2           OUTPUT PIN SEQ NO (RPQ 7B0570)
          DEFxx     s,TNPINSEQ,1   RESERVED
TNPSEQPP  DEFxx     s,,1           OUTPUT PIN SEQUENCE (PIN PAD)
TNPINKEY  DEFxx     s,,8           OUTPUT PIN KEY (ENCRYPTED)
TNPAN     DEFxx     s,,8           OUTPUT PRIMARY ACCOUNT NUMBER
TNPAN0    DEFxx     s,TNPAN,2      # DIGITS (2 BYTES) OF ZEROS
TNPANA    DEFxx     s,,6           12 DIGIT (6 BYTE) ACCOUNT NUMBER
```

- **DEFTRP**

```
TRPPAR    DEFxx     s,30       PARM LIST (INCLUDES 'TRPENT1')
TRPINS    DEFxx     TRPPAR,2   INPUT AREA'S SEGMENT
TRPIND    DEFxx     s,2        INPUT AREA'S DISPLACEMENT
TRPINL    DEFxx     s,2        INPUT AREA'S LENGTH
TRPOUS    DEFxx     s,2        OUTPUT AREA'S SEGMENT
TRPOUD    DEFxx     s,2        OUTPUT AREA'S DISPLACEMENT
TRPOUL    DEFxx     s,2        OUTPUT AREA'S LENGTH
TRPMSK    DEFxx     s,1        BREAK CONTROL MASK
TRPNTT    DEFxx     s,1        NUMBER OF TRANSLATE TABLES
TRPPTT    DEFxx     s,1        PRIOR TRANSLATE TABLE NUMBER
TRPCTT    DEFxx     s,1        CURRENT TRANSLATE TABLE NUMBER
TRPLIC    DEFxx     s,1        LAST INPUT CODE PROCESSED
TRPFNC    DEFxx     s,1        FUNCTION CODE
TRPLID    DEFxx     s,2        LAST INPUT CODE DISPLACEMENT
TRPLOD    DEFxx     s,2        LAST INPUT CODE'S OUTPUT DISPLACEMENT
TRPCNT    DEFxx     s,2        TOTAL NUMBER OF CHARACTERS OUTPUT
TRPTST    DEFxx     s,1        TERMINATION STATUS FIELDS
TRPTBR    EQUATE X'80'               TRANSLATE BREAK
TRPTOO    EQUATE X'40'               OUTPUT OVERFLOW
TRPTIO    EQUATE X'20'               INPUT OVERFLOW
TRPTII    EQUATE X'10'               INVALID INPUT CODE
TRPTNL    EQUATE X'08'               INITIAL OUTPUT LENGTH WAS ZERO
          DEFxx     s,1        RESERVED
TRPTTL    DEFxx     s,0        TRANSLATE TABLE LIST
TRPENT1   DEFxx     s,4        ENTRY1 OF XLATE TABLE LIST
TRPTTS1   DEFxx     TRPENT1,2    XLATE TABLE 1'S SEGMENT
TRPTTD1   DEFxx     s,2          XLATE TABLE 1'S DISPLACEMENT
```

- **DEFTRT**

```
TRTHDR   DEFxx   s,12        TRANSLATE TABLE HEADER
TRTLEN   DEFxx    TRTHDR,2   LENGTH OF WHOLE TABLE
TRTDOP   DEFxx   s,2         DISP: TBL-HDR TO OFF POINTED STRINGS
TRTLOW   DEFxx   s,1         LOW CODE IN TABLE
TRTELN   DEFxx   s,1         LENGTH OF ENTRY
TRTHFG   DEFxx   s,1         HEADER FLAGS
TRTTRO   EQUATE X'80'                ONLY TRANSLATE ENTRIES
TRTMDS   EQUATE X'40'                MODE = SKIP (FOR OUT OF RANGE)
TRTMDM   EQUATE X'20'                MODE = MOVE (FOR OUT OF RANGE)
TRTMDU   EQUATE X'10'                MODE = USE (FOR OUT OF RANGE)
TRTUSE   DEFxx   s,1        VALUE FOR 'USER' MODE
TRTNUM   DEFxx   s,2        NUMBER OF CODES IN THE TABLE
         DEFxx   s,2        RESERVED

*              TRANSLATE TABLE ENTRIES' FLAG BYTE

TRTEFG   DEFxx   s,0         ENTRIES' FLAG BYTE:
TRTOPS   EQUATE X'80'                VALUES IN OFF-POINTED STRING
TRTOCS   EQUATE X'40'                OUTPUT CHARACTER STRING DEFINED
TRTFNC   EQUATE X'20'                FUNCTION DEFINED
TRTBRK   EQUATE X'10'                BREAK DEFINED
TRTBAD   EQUATE X'08'                BREAK ADDRESS DEFINED
TRTINL   EQUATE X'04'                INLEN DEFINED TO BE OTHER THAN 1
TRTRES1  EQUATE X'02'                RESERVED  (MUST BE ZERO)
TRTTOC   EQUATE X'01'                2 OUTPUT CHAR'S IN TABLE ENTRY
```

- **DEFTSX**

```
TSXPAR   DEFxx   s,5          'TESTX' PARAMETER LIST
TSXFLG   DEFxx    TSXPAR,1    FLAG BYTE:
TSXFREM  EQUATE    X'80'                REQUEST SEGMENT'S STATUS
TSXFSTM  EQUATE    X'02'                STATION ACTIVE FOR INDEXING
TSXFSEM  EQUATE    X'01'                SEGMENT ACTIVE FOR INDEXING
TSXSID   DEFxx   s,1          STATION ID
TSXSEG   DEFxx   s,1          SEGMENT NUMBER
TSXADR   DEFxx   s,2          INDEX REG-NUM TBL'S ADDRESS
TSXREG1  DEFxx    TSXADR,1    OPERAND ONE'S INDEX REGISTER
TSXREG2  DEFxx   s,1          OPERAND TWO'S INDEX REGISTER
```

- **DEFVER3**

```
VER3PAR  DEFxx   s,,35        INSTRUCTION PARAMETER LIST
V3ALGTYP DEFxx   s,VER3PAR,1  VERIFICATION ALGORITHM TYPE
VA3624   EQUATE    X'00'                3624 VERIFICATION
         DEFxx   s,,1         RESERVED
V3PINKEY DEFxx   s,,8         PIN VERIFICATION KEY (ENCRYPTED)
V3DECTAB DEFxx   s,,8         DECIMALIZATION TABLE
V3VALDAT DEFxx   s,,8         PADDED VALIDATION DATA
V3CHKLEN DEFxx   s,,1         PIN CHECK LENGTH
V3OFFSET DEFxx   s,,8         PIN OFFSET DATA
```

- **DEFVUE**

```
VUEPAR   DEFxx   s,26         VIEW PARAMETER LIST
VUEREQ   DEFxx    VUEPAR,1    REQUEST CODE
VUESTA   DEFxx   s,1          STATION NUMBER
VUESTK   DEFxx   s,1          STACK ID
VUEFG1   DEFxx   s,1          FLAG BYTE 1
VUEFL0M  EQUATE    X'80'                STACK ENTRY IN USE
VUEFL1M  EQUATE    X'40'                STACK ENTRY PERMANENT
VUEFL2M  EQUATE    X'20'                'B' SET OF REGISTERS ACTIVE
VUEUIC   DEFxx   s,2          USER INSTRUCTION COUNTER
VUELSB   DEFxx   s,1          BOTTOM OF SMS LINK STACK
VUELSE   DEFxx   s,1          TOP OF SMS LINK STACK
VUEDEL   DEFxx   s,2          SAVED VALUE OF SMSDEL
VUEPNT   DEFxx   s,1          PARENT SEGMENT SPACE ID
VUEFG2   DEFxx   s,1          FLAG BYTE 2
VUERTF   EQUATE    X'80'                AP TRANSIENT FLAG
VUEFAC   EQUATE    X'01'                AP CALLABLE
VUEPID   DEFxx   s,8          APPLICATION PROGRAM NAME
VURES1   DEFxx   s,6          RESERVED
```

Chapter 13. Hexadecimal and Decimal Conversion

From hex: Locate each hexadecimal digit in its corresponding columns position and note the decimal equivalents. Add these to obtain the decimal value.

From decimal: (1) Locate the largest decimal value in the table that will fit into the decimal number to be converted, and (2) note its hexadecimal equivalent and hexadecimal column position. (3) Find the decimal remainder. Repeat the process on this and subsequent remainders.

HEXADECIMAL COLUMNS												
6		5		4		3		2		1		
HEX	DEC	HEX	DEC	HEX	DEC	HEX	DEC	HEX	DEC	HEX	DEC	
0	0	0	0	0	0	0	0	0	0	0	0	
1	1,048,576	1	65,536	1	4,096	1	256	1	16	1	1	
2	2,097,152	2	131,072	2	8,192	2	512	2	32	2	2	
3	3,145,728	3	196,608	3	12,288	3	768	3	48	3	3	
4	4,194,304	4	262,144	4	16,384	4	1,024	4	64	4	4	
5	5,242,880	5	327,680	5	20,480	5	1,280	5	80	5	5	
6	6,291,456	6	393,216	6	24,576	6	1,536	6	96	6	6	
7	7,340,032	7	458,752	7	28,672	7	1,792	7	112	7	7	
8	8,388,608	8	524,288	8	32,768	8	2,048	8	128	8	8	
9	9,437,184	9	589,824	9	36,864	9	2,304	9	144	9	9	
A	10,485,760	A	655,360	A	40,960	A	2,560	A	160	A	10	
B	11,534,336	B	720,896	B	45,056	B	2,816	B	176	B	11	
C	12,582,912	C	786,432	C	49,152	C	3,072	C	192	C	12	
D	13,631,488	D	851,968	D	53,248	D	3,328	D	208	D	13	
E	14,680,064	E	917,504	E	57,344	E	3,584	E	224	E	14	
F	15,728,640	F	983,040	F	61,440	F	3,840	F	240	F	15	
0 1 2 3		4 5 6 7		0 1 2 3		4 5 6 7		0 1 2 3		4 5 6 7		
BYTE				BYTE				BYTE				

POWERS OF 2

2^n	n
256	8
512	9
1 024	10
2 048	11
4 096	12
8 192	13
16 384	14
32 768	15
65 536	16
131 072	17
262 144	18
524 288	19
1 048 576	20
2 097 152	21
4 194 304	22
8 388 608	23
16 777 216	24

$2^0 = 16^0$
$2^4 = 16^1$
$2^8 = 16^2$
$2^{12} = 16^3$
$2^{16} = 16^4$
$2^{20} = 16^5$
$2^{24} = 16^6$
$2^{28} = 16^7$
$2^{32} = 16^8$
$2^{36} = 16^9$
$2^{40} = 16^{10}$
$2^{44} = 16^{11}$
$2^{48} = 16^{12}$
$2^{52} = 16^{13}$
$2^{56} = 16^{14}$
$2^{60} = 16^{15}$

POWERS OF 16

16^n	n
1	0
16	1
256	2
4 096	3
65 536	4
1 048 576	5
16 777 216	6
268 435 456	7
4 294 967 296	8
68 719 476 736	9
1 099 511 627 776	10
17 592 186 044 416	11
281 474 976 710 656	12
4 503 599 627 370 496	13
72 057 594 037 927 936	14
1 152 921 504 606 846 976	15

IBM 4700 Finance
Communication System
Reference Digest

Order No. GC31-2034-1

READER'S COMMENT FORM

This manual is part of a library that serves as a reference source for systems analysts, programmers, and operators of IBM systems. You may use this form to communicate your comments about this publication, its organization, or subject matter, with the understanding that IBM may use or distribute whatever information you supply in any way it believes appropriate without incurring any obligation to you.

Your comments will be sent to the author's department for whatever review and action, if any, are deemed appropriate. Comments may be written in your own language; English is not required.

Note: *Copies of IBM publications are not stocked at the location to which this form is addressed. Please direct any requests for copies of publications, or for assistance in using your IBM system, to your IBM representative or to the IBM branch office serving your locality.*

Possible topics for comment are:

Clarity Accuracy Completeness Organization
Coding Retrieval Legibility

If you wish a reply, give your name, company, mailing address, and date:

_____

_____

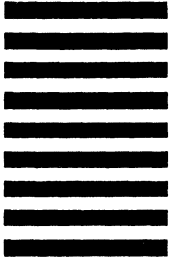
_____

What is your occupation? _____

Number of latest Newsletter associated with this publication: _____

Thank you for your cooperation. No postage stamp necessary if mailed in the U.S.A. (Elsewhere, an IBM office or representative will be happy to forward your comments or you may mail directly to the address in the Edition Notice on the back of the title page.)

NO POSTAGE NECESSARY IF MAILED IN THE UNITED STATES



BUSINESS REPLY MAIL

FIRST CLASS PERMIT NO. 40 ARMONK, N.Y.

POSTAGE WILL BE PAID BY ADDRESSEE:

International Business Machines Corporation
Department 78C
1001 W.T. Harris Boulevard
Charlotte, NC. USA 28257

GC31-2034-1

PRINTED IN U.S.A.