

**PROPRIETARY INFORMATION OF
INFORMATION INTERNATIONAL, INC.**

**POCKET
REFERENCE MANUAL**

NOTICE

This material contains information proprietary to Information International, Inc. It is provided with the express condition that it will not be copied, disclosed, or used for any purpose except as directly related to the operation of Information International Products.



INFORMATION INTERNATIONAL®

5933 SLAUSON AVE., CULVER CITY, CALIFORNIA 90230
TELEPHONE (213) 390-8611

MARCH 1976

CHANGE RECORD

CHANGE NO.	DATE OF ISSUE	SIGNATURE OF PERSON ENTERING CHANGE IN THIS BOOK	DATE OF ENTRY


Copyright © 1976 by Information International, Inc.

All rights reserved. No part of this work covered by the copyright hereon may be reproduced or copied in any form or by any means -- graphic, electronic, or mechanical, including photocopying, recording, taping, or information storage and retrieval systems -- without written permission of the publisher.

Made in the United States of America.

Published by Information International, Inc.
5933 Slauson Avenue
Culver City, California 90230

INSERT CHANGE NOTICES
 ON OPPOSITE PAGE.



GENERAL

DECIMAL TO OCTAL CONVERSION CHART

	0-2	3-5	6-8	9-11	12-14	15-17
000000.	000000.	000000.	000000.	000000.	000000.	000000.
32768.	4096.	512.	64.	8.	1	.125
65536.	8192.	1024.	128.	16.	2	.250
98304.	12288.	1536.	192.	24.	3	.375
131072.	16384.	2048.	256.	32.	4	.500
163840.	20480.	2560.	320.	40.	5	.625
196608.	24576.	3072.	384.	48.	6	.750
229376.	28672.	3584.	448.	56.	7	.875

.5	.0625	.0078125
.25	.03125	.00390625
.125	.015625	.001953125

/HEXADECIMAL TO DECIMAL

0	0	0	0	0
1	4096	256	16	1
2	8192	512	32	2
3	12,288	768	48	3
4	16,384	1024	64	4
5	20,480	1280	80	5
6	24,576	1536	96	6
7	28,672	1792	112	7
8	32,768	2048	128	8
9	36,864	2304	144	9
A	40,960	2560	160	10
B	45,056	2816	176	11
C	49,152	3072	192	12
D	53,248	3328	208	13
E	57,344	3584	224	14
F	61,440	3840	240	15

INSTALLATION NOTES

/Single phase power

1000

KVA x ----- = AMPS approx KVA x 8.3
VOLTS

/Three phase power

1000

KVA x ----- = AMPS approx KVA x 2.6
1.73 x VOLTS

KVA x .8 = Kilowatts

Tons (of air conditioning) x 12,000 = British Thermal Units

KW x 3.413 = BTU

KW x .284 = Tons of air conditioning

'Y' is preferred type of xfrmr

Voltage < 1v between neutral and ground

Resistance > 1k ohm between neutral and ground

Temperature 65 to 75 degrees F

Humidity 50 to 65 %

/	UNIT	AMPS	TONS AC	POWER	WEIGHT
270	PCU	40	1.4-2.4		
1600	CPU	10	0.75		
432-1		8	0.7		
432-2		11	0.7		
564R		3.1	0.25		
FR80	Recorder	20	0.25	1.1KW	850
PDP9L	CPU	30	0.6	2.1KW	1600
PEC	Tape Deck	20	0.1	0.3KW	200

WIRE	CURRENT	WIRE	CURRENT
00	110-130	8	30-40
0	90-110	10	20-30
2	70-90	12	10-20
4	50-70	14	5-10
6	40-50	16	3-5

GENERAL ELECTRONICS

/CHIP EQUIVALENCE CHART

SG80 <- MC476P	SG83 <- MC476P	8281 <- SN74197
MC3000 = 74H00	MC3001 = 74H08	MC3002 = 74H02
MC3004 = 74H01	MC3005 = 74H10	MC3006 = 74H11
MC3008 = 74H04		
MC3009 = 74H05	MC3010 = 74H20	MC3011 = 74H21
MC3018 = 74H62	MC3021 = 74S86	MC3023 = 74H51
MC3030 = 74H60	MC3031 = 74H52	MC3032 = 74H53
MC3033 = 74H54	MC3034 = 74H55	MC3060 = 74H74
MC3061 = 74H108		

198407-1 <- SN7400
198407-2 <- SN7410
198407-3 <- SN7420
198407-4 <- SN7430
198407-5 <- SN7440
198407-6 <- SN7450
198407-7 <- SN7460
198407-8 <- SN7438
198406-2 <- SN7472
198408-1 <- 93407
198415-1 <- SN7474

/MIL Spec Resistor numbering

R C ww G F xxm t

R = Resistor

C = Composition

ww = Wattage

05=.125wt 07=.25wt

20=.500wt 32=1.0wt

42=2.0wt

xxm = Two significant digits & multiplier.

t = Tolerance J = 5% K = 10%

.....NOTES.....

MC2010 is high fanout 2010

SYSTEM TAPE AND DISK FORMATS

FH	0-37	Image of current program
MH	1350-1377	
FH	40-77	Image of current DEBUG
MH	1400-1457	
FH1D	1754	Master directory
FH2D	3736	
MH	1460	
FH1D	1755-73	User directories
FH2D	3737-3774	
MH	1361-1516	
FH1D	1774	Track usage table
FH2D	3775	
MH	1517-1524	
FH1D	1775	Message of the day
FH2D	3776	
MH	0	
FH1D	1776	Tape names for DIR
FH2D	3777	
MH	1	
FH	1777	Bootstrap
MH	14	

/TRACK USAGE TABLE

W=DEBUG swapping blocks	X=Allocated
--Available	M=Master directory
D=User directory	T=Track Usage Table
U=Unused	A=Message
N=Tape names for each direct.	B=Bootstrap

*****/DISC DUMP FORMAT ON TAPE*****

HEADER: 400 OCTAL WORDS

0	DIRNAME 3CHR-WD(ASCII-40)
1-4	FILE NAME
5	0-556,1-800
	-1-9TK,CORE DMP TMS TWOS COMPLEMENT
	-2-9TK,NORMAL TMS TWOS COMPLEMENT
6-270	DATE (8 bit ASCII) 1 chr-wd
271-377	Bootstrap (Search R BIN)

DATA: 400 OCTAL WDS(BLKD 1-8)

0	Contains BLK SER#(0 for last).
	Sign bit is flag for cksum error.
1-377	Data direct from disc
400	Unused

Symbolic files contain 2 chrs-wd.

8 bit ASCII right justified in each 0.5 word.

End chr=141 OCTAL

MAG TAPE DISPLAYER (III-123) M

(cr) New line, cancels param
ctl K Kill tty input buffer
ctl S Stop printing
ctl D Enter debug (RETURN\$X to resume)

*****POSITIONING****

n# Select drive n
L Rewind
m,nR Read nth record of mth file
. Reread current record
/,/R Reads last rec of last file
/R Reads last rec of curr. file
-nR Reads nth rec previous
-R Reads previous record
nW Display starts @ nth word

*****DISPLAY*****

nnnC Compact n word groups
nW Display starts @ nth word
\$ Display starting at next part of record
n\$ Skip 8(n-1) words
nnn= Search for a word equal to nnn
nnn< Search for a word less than to nnn
nnn> Search for a word greater than to nnn
nnnN Search for word not equal to nnn

*****MODES****

Space Change display back to commands
S,U Signed and Unsigned
Z,X Suppress, print zeroes
D,O Decimal, octal
P,Q Page, Line
M,T Monitor, TTY

D O H E F,
DECIMAL, OCTAL, HEX, EBCDIC, BCD

2B - 5B - 8B - 9B
200 556 800 9TK

nR mF 4000 W? BIN odd
BCD even
+ before "w" says buffer too small
? after "w" says parity error

***** BUFFER SIZE

4000(8) - 2048(10) 18 BIT WORDS
6144(10) CHARACTERS
SW0 Request unit address

DISK DEBUG (III-126D) (1 of 2)

LINK	ACCUMULATOR	MQ REG	J+1
1	aaaaaa	mqmqmq	jjjjj

\$K Kill symbols
\$\$Z Zero memory
\$\$K \$K & \$\$Z
a\$\$Z Zero memory above "a"
a**b**\$\$Z Zero from "a" to "b"

/PROGRAM LOADING

\$ Clears
\$\$ Doesn't clear
name\$L Loads with symbols
name\$J No symbols, starts at "J"
name\$G No symbols, starts at "J+1"
name\$\$ Symbols only
name\$H Load SYMBOLS, & Paper Tape DEBUG(DDD)
name\$T Compare program & BINARY file
name\$\$D Delete BINARY program
name\$F Save core as program

/*****NEW FUNCTIONS*****/

/SWITCHES

@ Display on-off
ctlQ Octal-Symbolic
ctlS Scroll down (sw15 on old)
ctlU Scroll up (sw14 on old)
ctlT TTY buffer on-off
? Printer on-off

/FUNCTIONS

ctlL Erase TTY buffer
ctlG Open 1st line below display
ctlH Open 1st line above display
ctlC Center the open cell on display
ctlV Put open cell at top of display
ctlY Continue search
\$\$ On search will stop @ next one
.\$B Reverse breakpoint
ctlO Reverse TAB(ctlI)
O\$R Display leading zeroes
I\$R Suppress leading zeroes
a**b**\$Y Type from a to b on the TTY
\$open Use full 15 bits for address

DISK DEBUG (III-126D) (2 of 2)

\$S	Symbolic
\$C	Constant
\$R	Relative
\$O	Octal
n\$R	Set base to "n" (n is octal)
n.	n=Decimal
\$W	Word search
\$N	Not-word search
\$A	Address search
\$E	Effective address search
\$U	Unsave tape (Punch)
\$X	Execute instruction
\$A	Accumulator
\$L	Link
\$Q	MQ
\$F	Symbol table
\$J	Start address
\$J+1	Return address
\$M	Search mask
\$M+1	Lower limit
\$M+2	Upper limit
\$B	Breakpoint
\$P	Proceed after breakpoint
.	Current address
#	Last value typed
'	Start (Location J)
"	Return (J+1)
s.	Give current tag "s"
a<s>	Give "a" tag "s"
=	Equals current mode
:	Equals opposite mode
+	Plus
-	Minus
*	Times
%	Divide
!	Or
/	Open current mode
\	Open opposite mode
(cr)	Close
(lf)	Open next
↑	Open previous
)	Open addressed
TAB	New sequenced addressed
(Open patch mode

.....**REGISTERS**.....

\$A	Accumulator
\$L	Link
\$Q	MQ
\$F	Bottom of symbol table
\$J	Single quote start
\$J+1	Double quote start
\$J+2	Contents of "SC"
\$J+3	Exlevel mode statue
\$M	Mask for searches
\$M+1	Lower search limit
\$M+2	Upper search limit
#	Value of last quantity typed

DISK AUDIT (III-148) F

...../DIRECTORY MODE.....

(cr) Goes to file mode
\$RESTORE(cr) Restores all previous deletions
\$T(cr) Types monitor display
\$CLEAN(cr) 1 Deletes all "LISTER"
2 All directories with no entries
3 All "BINARY" files that have a logical file
4 All "TEMP" file
\$ZEROaaa(cr) Zero block aaa
@ Enter-exit complete display
#N(cr) Switch to file mode in directory n (SYS is #1)
\$U(cr) Display Track Usage Table
\$M(cr)mmm(lf) Store system message
\$UT(cr) Type Track Usage Table
\$SBaaaa(cr) Shows contents in binary 64 words displayed
\$ Goes to next 64 words.
N Shows contents of block chained
Space Shows contents of next block
A Shows contents of next avail block block #
& chain word
@ (cr) Returns to dir mode
N Shows nxt consec blk
↑ Shows prev consec blk
P Shows back-chain blk
B Types octal # of blk shown
\$Saaaa(cr) Shows contents of block aaaa in symbolic form
uses N, Space and A like "\$SB"

...../FILE MODE.....

\$RESTORE(cr) Restores all previous deletions
\$DELETE(cr) Deletes all files
\$T(cr) Type monitor
\$CLEAN(cr) 1 Deletes all "LISTER"
2 All directories with no entries
3 All "BINARY" files that have a logical file
4 All "TEMP" file
@ Enter-exit complete display
name\$D(cr) Delete file
name\$B(cr) Delete binary file
name1<name(cr)2 Change from name1 to name2
name\$S(cr) Shows contents of block in symbolic form
N Shows contents of block chained
Space Shows contents of next block
A Shows contents of next avail block block #
& chain word
name\$SB(cr) Shows contents in binary 64 words displayed
\$ Goes to next 64 words.
N Shows contents of block chained
Space Shows contents of next block
A Shows contents of next avail block block #
& chain word
name=aaaa(cr) Names block aaaa with the name in current directory
:nnnn(cr) Translate typeout

...../NOTES.....

11409 MAX ON PERTCE
/USRLS2/ USERNM, JKP BEGIN+3

DMAINT COMMMAND SUMMARY

File Commands:

Command (space) INPUT->OUTPUT

'COPY' Copy input file(s) to output
'DELETE' Delete file(s) (input only)
'MOVE' Copy, then delete input
accomplish with RENAME if possible
'AUDIT' List files, length in blocks (input only)
if no arguments, list directory names
'BCOPY', 'BDELETE', 'BMOVE', 'BAUDIT'
Same as above, except 2nd word of
file names assumed 'BINARY'
'INPUT' Get operator commands from disk file
'HELP' Get this display

File Specification:

[FSN|DIR:FILENAME

FSN File structure name
DIR Directory name
FILENAME Disk file name

Special characters:

"-" Separates input from output
>" Direction of process (generated by typing "-")
"-." Exception input file or directory follows
"." More input follows
<" (carriage return) end of command
+ New "INPUT->OUTPUT" for same command follows
** Wild card file name word (6 char)
?* Wild card file name character
<" Argument(s) from previous file command
ctl-I Display list of file structures
ctl-U Retype command entered so far
ctl-O AUDIT, use output argument of last command
LN FD RUB-OUT all of command so far

Formatting commands:

'WIPE' Wipe and reformat file structure
(syntax as in disk reload program)
'REFRESH' Reformat and rebuild TUT without wiping
(syntax as in 'WIPE')
'ZERO' [FSN].N Zero out block N of [FSN]
'\$T' Type contents of display buffer
'\$CLEAN' Delete any directories with no files
'X' Type or display file names processed
'Z' Do not type or display file names processed

Examples:

COPY [SYS]SYM;IIII09.SYN;-IIII17->[FIX]
Copy SYM;IIII09, and all of SYN; except IIII17
from [SYS] to [FIX]
COPY [DIR]-SYS;-SYM;IIII09->[DIF]SYM;
Copy all of [DIR], except directory SYS; and file
SYM;IIII09, from [DIR] to [DIF], directory SYM;
AUDIT MDG;
Audit the MDG directory only

Note: "[" is a shift K and "]" is a shift M

TAPE DUMP RELOAD (III-139) R

ALL(cr)	Dump complete tape to Disc
Bfff(cr)	Same as Ffff BINARY(cr)
Dddd(cr)	Load "ddd onto disk (re-name if desired)
Ddd1.dd2(cr)	Load ddd onto disk (re-name if desired)
E	Search f"or descriptor record(date) Change tape position to read that file next
Ffff(cr)	Search tape for file "fff" & load on disk
Ffff.fff(cr)	Search & replace & copy
l	Index tape (display on monitor)
Kfff(cr)	Delete specific file from disc
Lfff(cr)	Load binary file into core
N	Initialize new tape
P	Punch a bootstrap
R	Rewind the tape
T	Index tape (display on TTY)
X	Type successful copies to disc during ALL & D
Z	Kill X command
?	Type index of tape
WIPE	Clean disk (except swapping blocks) Make new TRACK USAGE TABLE Make new Master directory with "SYS". Make new Bootstrap
-	Back 1 file
+	Forward 1 file
.	Go to beginning of curret file & type

DISK DUMPER (III-138) S

ALL(cr)	Rewind & dump entire disc to tape
B	Backs over 1 file & type name
C	Go to next "DATE ENTRY" & use it for a response to "DATE?"
Dddd	Dump specified directory
Ddd1<dd2	Same only change name to dd2
Ddd1,dd2	Dump directories "dd1" & "dd2"
E	Position tape to end of previous dump
Ffff	Copy file "fff" to tape
Fff1<ff2	Replace + dump.
Hfff	Card to tape (see P)
Kfff	Card to tape verify
LN(cr)	Number of cards treated as page
N	Initiate date request
Pfff	Copy Paper Tape to Mag Tape. If file not binary, Type C-S: C=Continue with next tape, S=Stop
Qfff	Same as P but force BINARY
R	Rewind the tape
Sn	Set tape blocking factor "n"(1 to 8)
T	Type index of tape
V	Verifies Paper Tape file
WEOT	Write filemark to tape
X	Type file names in "D" & "ALL"
Z	Suppres type file names in "D" & "ALL"
?	Index tape

FR80 TEXT EDITOR

AC0 KILLS TTY

/ Total # of lines in buffer.
. Last line.
: Enter character mode
nR Read n lines.
m,nX Read next m lines & insert
after line n.
A Enter text mode & append.
nI Insert before line n.
m,nC Change lines m thru n.
m,nD Delete lines m thru n.
m,nL List(print) lines m thru n.
K Kill buffer.
nALT Move pointer down n lines.
nU Move pointer up n lines.
Q Uncommented print.
nSP In character mode move right.
RO A)Character--backspace.
B)Ends mode.
C)Rubs out characters.
nJ Open line n at 2nd tab & enter text
mode for comment insertion.
V-IV Turn on-turn off line numbering
M String search.
m,nP Punch lines m thru n.
S Leader,form feed & trailer.
nF Punch n lines blank tape.
O Punch 1 page(P.S.K)
N Punch & read next page(O.R).
T Tape duplication.
nW Write n pages(K,R,W,K each page)
nH Same as W with line numbering.
*O In char mode goes to overwrite mode.

COMPOSE TEXT EDITOR (III-180) (1 of 3)

Text mode is used to enter the main body of text.

Carriage Return creates new line, cttL new page

cttA Load & chain to A BINARY(FORMS DESIGN or ASSEMBLER)

cttB Load & chain to B BINARY(COMPOSE)

cttD Load DISK DEBUG

cttE Load & chain to E BINARY(EDITOR)

*****MONITOR DISPLAY COMMANDS

ALT Change the display from line to page mode.

n: Display "n" lines at a time

cttX Increase-stop char size

cttZ Decrease-stop char size

cttC Center displ at curr line

cttG Cursor to 1st below screen

cttH Cursor to 1st above screen.

@ On-off special display chars

*****MOVING THE CURSORS

rSP Cursor down "r" lines (page) cursor rt "r" chars (line)

rRO Cursor up "r" lines (page) cursor left "r" chars (line)

nN Output curr pg.move "n" pages fwd or bkwd for neg n

n. Cursor to line n of buffer

nP Position at pg n

rG Move down r lines & srch 1st line of a group (no cr.sp.tab

rH Move up r lines & srch last line of a group

*****COMMANDS ENTER TEXT MODE: (FRM PG MODE

rA Insrt new line blw r. enter text

rI Insrt new above r. enter text mode

rC Delete curr line thru r. enter text mode

*****CMDS ENTERNG TEXT MODE. FRM LINE MODE:

rA Move rt r chars.enter text mode to rt of that chr

rI Move rt r chars enter text mode. to left of that chr

rC(cr) Delete curr chr thru chr r. enter chr text mode:

No param delete curr chr.enter chr text mode

*****IN EITHER DISPLAY MODE:

O Overwrite mode

rE Change case of curr chr if letter

*****SPECIAL CHRS IN TEXT MODES:

(lf) Exit text chr text.ovrwr mode to command mode

(cr) Text&chr modes:enter cr to text buf ovrrt mode: exit to cmd mode

cttL Enter form feed chr in bufr: must be entered as 1st chr of line.

RUB Text mode:delete last chr entered chr text mode

Same ovrrt mode: move box 1 chr left

cttR Delete all chrs entered on curr line

ALT On-off lower case

cttC Center displ.remain text mode

*****ALTER TEXT:

rK(cr) Delete curr line thru line r

J(cr) Kill curr text bufr.read next page

rD(cr) Delete curr chr thru chr r

nV(cr) Enter chr text mode insert chr code octal n in buff.

Enter in mode

rS(cr) Output curr line thru line r

nR(cr) Copy n lines from beg next pg insert blw-abv curr line neg-pos

(sft.cttL) Output form feed


```

*****COMMENT CHARACTER COMMANDS
rU      Delete comm chr frm beg line r
rX      Insert cmnt chr at beg line r
^a      Define cmnt chr as a
*****CLICHES
*string* Max 100 chrs-string
n:      Do cliché n times
*****TAB STOPS
ctll    Tab in text Reset tabs to desired pos.
string(lf) Set tabs as per string
        --Tab. .-Columns in between
ctlt    Change contig sp before tab stops ls line r to tabs
*****PRINTING ON TELEPR. FRM TEXT BUFR
r.      Cursor to line r & print line
r<      Print this line up to chr r
r>      Print this line frm chr r on
r(lf)   Print betw line r & curr line
R       Print betw line -r & curr line
nw(cr)  Teletype top of frms, print n pages. no param-curr page
r#      Prnt pg & line# (page mode) prnt curr chr in octal ascii
(sft,ctIP) Punch 10 feed frames
*****SETTING TELEPRINTER MODES
ctll    Tell editor telepr at Top Of Form
?       On-off echo keybd
&       On-off print absolute pg & line #s
&       On-off print curr line on tty
ctls    Kill tty output bufr
*****LINE PRINTER COMMANDS
nZ(cr)  Print n pgs beg wth curr & eject after last one
r'      Print line r of curr text bfr
    
```

COMPOSE TEXT EDITOR (III-180) (3 of 3)

*****CHARACTER SEARCH COMMANDS

rLa Search text bfr for chr a
ctlN Repeat prev L search

*****MATCHED DELIMITER SEARCH

r(Search bkwd for unmatched opener
r) Search fwd for unmtchd closer
(a Define opener as a
)a Define closer as a

*****SEARCHING

rQstring(lf) Search text bfr for spec chr string
rTstring(lf) Search file for chr string
rMstring(lf) Search (Q) with delimitrs
rYstring(lf) Search (T) with dlmitrs
FQ(cr)string(lf) If not in FX mode, simulate FW if a file is ope
"T" search 1st file.
If FX mode, simulate FX cmnds else FR cmnds
FM(cr)string(lf) Same as FQ, but Y search
RetlY Repeat prev strng search starting with next chr

*****SECONDARY DISPLAY COMMANDS

% On-off display
I Invert 2nd disp-teleprinter
ctlO Clear 2nd disp buffer
PI\$(cr) 2nd disp buff=insert file
n(sft&ctl&M) Increase-stop hor size of secondary display

*****MANIPULATING FILES

FOfff(cr) Select outpt file fff
FNfff(cr) Create-replace spcfd disc file with curr output
FWfff(cr) Copy remaindr of input file onto end of curr output
then create-replace specified file
FCfff(cr) Same as FW, but returns to current line & page
FEfff(cr) Same as FW, but external default file name is
established, and pointer is 1st line 1st page
FAfff(cr) Same as FE, but chains to A BINARY
FBfff(cr) Same as FE, but chains to B BINARY
FRfff(cr) Kill curr output & read 1st page of fff
FIfff(cr) Insert "fff" into current file
FP(cr) Truncate curr input file
FK(cr) Kill curr output
FDfff(cr) Delete "fff" from disk
FXfff(cr) Examine "fff", but current is still default
FS(cr) Return to curr file from FX
FT(cr) Try again after disc full

ctlA Load & chain to A BINARY(FORMS DESIGN or ASSEMBLER)
ctlB Load & chain to B BINARY(COMPOSE)
ctlE Load & chain to E BINARY(EDITOR)

BASIC OP CODES

/MEMORY REFERENCE

CAL=000000	Call subroutine (JMS 20)
DZM=140000	Deposit zero in memory
LAM=777777-n	Load ACC with minus number
XOR=240000	Exclusive OR
TAD=340000	Add, two's complement
(No end around carry, goes to Link)	
ISZ=440000	Increment and skip if zero
SAD=540000	Skip if ACC different from memory
DAC=040000	Deposit accumulator
LAC=200000	Load accumulator
JMS=100000	Jump to subroutine
ADD=300000	Add, ones complement
XCT=400000	Execute the instruction at EA
AND=500000	And (logical product)
JMP=600000	Unconditional jump
LAW=760000+n	Load the ACC with this instruction

/OPERATES

OPR=740000	No operation
NOP=740000	No operation
CLA=750000	Clear the ACC
CLL=744000	Clear the link
CMA=740001	Complement the AC
CML=740002	Complement the LINK
CLC=750001	Clear and complement the AC
STL=744002	Set the link
CCL=744002	Set the link
RAL=740010	Rotate AC and L one left
RAR=740020	Rotate AC and L one right
RTL=742010	Rotate AC and L two left
RTR=742020	Rotate AC and L two right
RCR=744020	Clear link and rotate right
RCL=744010	Clear link and rotate left
OAS=740004	"OR" console switches with AC
LAS=750004	Load ACC from console switches
SPA=741100	Skip if AC positive (BIT 00 = 0)
SMA=740100	Skip if AC negative (BIT 00 = 1)
SZA=740200	Skip if AC is zero
SNA=741200	Skip if AC is non-zero
SZL=741400	Skip if Link is zero
SNL=740400	Skip if Link is non-zero
SPL=741400	Skip if Link positive (L = 0)
SML=740400	Skip if Link negative (L = 1)
SKP=741000	Unconditional skip
GLK=740010	Get the Link
HLT=740040	Halt the CPU

BASIC OP CODES

/EXTEND MODE IOTS
SEM=707701 Skip if extend mode
EEM=707702 Enter extend mode (and bank mode on 115)
LEM=707704 Leave extend mode
EMIR=707742 Extend mode interrupt restore (PDP 7)
P15SBA=707761 Skip if bank addressing (P15'S)
P15DBA=707762 Disable bank addressing (enter page mode)
EBA=707764 Enable bank addressing

/ADDITIONAL PDP-15 OP CODES
IAC=740030 Increment accumulator
Overflow complements link
SWHA=742030 Swap halves accumulator
TCA=740031 Two's complement accumulator
Complement and add one
PAX=721000+n Transfer the ACC to the XR
PAL=722000+n Transfer the ACC to the LR
PXA=724000+n Transfer the XR to the ACC
PXL=726000+n Transfer the XR to the LR
PLA=730000+n Transfer the LR to the ACC
PLX=731000+n Transfer the LR to the XR
AXS=725000* Skip if the XR is greater than or equal to the LR
AXR=737000* Add "n" to XR
AAC=723000* Add "n" to ACC
CLX=735000+n Clear and add "n" to XR
CLLR=736000+n Clear and add "n" to LR
*On PDP15 these and only these OXR instruction have signed deltas.
while on 115 all OXR instructions have signed deltas.

/ADDITIONAL III-15 OP CODES
AAS=720000+n Skip if the ACC is greater than or equal to the LR
LACC=734000+n Clear and add "n" to XR
ALR=732000+n Add "n" to LR
DSPADR=703702 Display ACC in ADDRESS lights
DSPREG=703722 Display ACC in DATA lights
SBA=707741 Skip if bank addressing
DBA=707744 Disable bank addressing

EXTENDED ARITHMETIC ELEMENT

00	01	02	*	03	04	05	*	06	07	08	*	09	10	11	*	12	13	14	*	15	16	17
		20K		04						AC 0 to Link at Time 1												
		10K		05						Clear MQ at Time 1												
		4K		06						AC 0 to "EAES" FF at Time 1												
		2K		07						ACC or'ed with MQ to MQ at Time 2												
		1K		08						Clear ACC at Time 2												
		400		09						EAE Command bit												
		200		10						EAE Command bit												
		100		11						EAE Command bit												
			0	EAE OPR						4												
			1	MultiPLY						5												
			2	unused						6												
			3	Divide						7												
		40		12						Step Counter - unused												
		20		13						Step Counter - unused												
		10		14						Step Counter - unused												
		4		15						Step Counter - Complement MQ at Time 3												
		2		16						Step Counter - ACC or MQ to ACC at Time 3												
		1		17						Step Counter - ACC or SC to ACC at Time 3												
				EAE=640000						No operation (EAE)												
				CLQ=650000						Clear the MQ												
				ABS=644000						Load AC with absolute value of the AC												
				GSM=664000						Get sign and magnitude of the AC												
				OSC=640001						Inclusive OR SC with AC												
				OMQ=640002						Inclusive OR MQ with AC												
				ACOMQ=660000						Or ACC and MQ only leave results in MQ												
				CMQ=640004						Complement the MQ												
				LMQ=652000						Load the MQ												
				LACQ=641002						Load AC from MQ												
				LACS=641001						Load AC from SC												
				NORM=640444						Normalize												
				NORMS=660444						Normalize, signed												
				LRS=640500+n						Long right shift												
1.				LRSS=660500+n						Long right shift, signed												
				LLS=640600+n						Long left shift												
1.				LLSS=660600+n						Long left shift, signed												
				ALS=640700+n						AC left shift												
1.				ALSS=660700+n						AC left shift, signed												
2.				MUL=653122						Multiply, unsigned												
3.				MULS=657122						Multiply, signed, Link is zeroed												
				ACC and MQ divided by the next location (Y)																		
2.				DIV=640323						Divide, unsigned												
3.				DIVS=644323						Divide, signed												
				36 bit divide (ACC > 0)																		
				ACC (18 bits) divided by the next location (Y)																		
2.				IDIV=653323						Integer divide, unsigned												
				IDIVS=657323						Integer divide, signed												
				ACC goes to the MQ at setup (ACC > Y)																		
				The ACC and Y form a fraction, so the answer has decimal point to the left of ACC 00.																		
2.				FRDIV=650323						Fraction divide, unsigned												
				FRDIVS=654323						Fraction divide, signed												
				MQ is zeroed at setup (Y > ACC)																		
1.				ACC 0 goes to Link on first step.																		
2.				Link must be preset to zero.																		
3.				Format of ACC and MQ.																		
				*S	*S	*		ACC	*		MQ	*										
				00	01	02								17	00							17

BASIC IOT'S

Output - 4 cycles Input - 3 cycles

00 01 02 03 04 05 06 07 08 09 10 11 12 13 14 15 16 17

10K	05	On I15 says make IOT a "FAST IOT"
4K	06	Device selection DS0
2K	07	Device selection DS1
1K	08	Device selection DS2
400	09	Device selection DS3
200	10	Device selection DS4
100	11	Device selection DS5
40	12	SubDevice SD0
20	13	SubDevice SD1
10	14	Clear accumulator at time 1
4	15	IOP1 <usually SKIP command>
2	16	IOP2 <READ (fetch data to CPU) or CLEAR command>
1	17	IOP4 <WRITE (send data from CPU) command>
IOT=700000		No operation
IOF=700002		Turn interrupt off
ION=700042		Turn interrupt on
CAF=703302		Clear all flags
DBR=773304		Debreak and restore
IORS=700314		I-O read status
400K	00	Program interrupt on
200K	01	Paper tape reader, flag true
100K	02	Paper tape punch, flag true
40K	03	Teletype keyboard, flag true
20K	04	Teletype printer, flag true
10K	05	Light Pen Flag
4K	06	Real time clock, overflow flag true
2K	07	Real time clock enabled
1K	08	Paper tape reader, no tape <on I15, really means not ready>
400	09	Paper tape punch, no tape
200	10	DECTAPE flag
100	11	Conventional MAG TAPE flag
40	12	Disk
20	13	DECdisk
10	14	Line Printer
4	15	
2	16	
1	17	DRUM flag
		/CLOCK IOT'S
CLSF=700001		Skip if clock flag
CLOF=700004		Disable the clock
CLON=700044		Enable the clock

TELETYPE-PAPER TAPE

/TELETYPE KEYBOARD IOTS

KSF=700301 Skip if keyboard flag is true
KRB=700312 Read keyboard buffer to bits 10-17 of acc
also clear keyboard flag (half duplex)
KRS=700332 Read keyboard buffer and select keyboard
reader and clear flag (full duplex)

/TELETYPE PRINTER IOTS

TSF=700401 Skip if printer flag is true
TCF=700402 Clear printer flag
TLS=700406 Load printer buffer from bits 10-17 of ACC
this will also clear printer flag.

/PAPER TAPE READER IOTS

RSF=700101 Skip if reader flag is true
RCF=700102 "or" contents of reader buffer to ACC
this will also clear reader flag
RRB=700112 Loads contents of reader buffer to ACC
this will also clear reader flag
RSA=700104 Select alphanumeric mode
says reader buffer is 8 bits long and will go to
bits 10-17 of accumulator.
RSB=700144 Select binary mode
Takes holes 1-6, ignores hole 7 if hole 8 is punched
and creates an 18 bit word with three characters
in the binary mode format

/PAPER TAPE PUNCH IOTS

PSF=700201 Skip if punch flag is true
PCF=700202 Clear the punch flag
PSA=700204 Punch a line in the alphanumeric mode
PLS=700206 Punch a line in the alphanumeric mode and
also clear the flag
PSB=700244 Punch a line in the binary mode

/BINARY MODE FORMAT

1st character * 2nd character * 3rd character
00 01 02 03 04 05 06 07 08 09 10 11 12 13 14 15 16 17
HOLE 6**5**4**3**2**1 6**5**4**3**2**1 6**5**4**3**2**1

/ALPHANUMERIC MODE FORMAT

character
00 01 02 03 04 05 06 07 08 09 10 11 12 13 14 15 16 17
HOLE 8**7**6**5**4**3**2**1

NOTE: TTY must be in "KT" mode.

ASCII CHARACTER SET

	200-237	240-277	300-337	340-377	
00	NUL	SPACE (00)	@ (40)		40
01	SOH(Ctl A)	! (01)	A (41)	a	41
02	(Ctl B)	" (02)	B (42)	b	42
03	ETX(Ctl C)	# (03)	C (43)	c	43
04	EOT(Ctl D)	\$ (04)	D (44)	d	44
05	(Ctl E)	% (05)	E (45)	e	45
06	(Ctl F)	& (06)	F (46)	f	46
07	BELL	' (07)	G (47)	g	47
10	(Ctl H)	((10)	H (50)	h	50
11	HOR TAB) (11)	I (51)	i	51
12	LINE FEED	* (12)	J (52)	j	52
13	VERT TAB	+ (13)	K (53)	k	53
14	FORM FEED	. (14)	L (54)	l	54
15	CARR RET	- (15)	M (55)	m	55
16	(Ctl N)	. (16)	N (56)	n	56
17	(Ctl O)	/ (17)	O (57)	o	57
20	DLE(Ctl P)	0 (20)	P (60)	p	60
21	DC1(Ctl Q)	1 (21)	Q (61)	q	61
22	DC2(Ctl R)	2 (22)	R (62)	r	62
23	DC3(Ctl S)	3 (23)	S (63)	s	63
24	DC4(Ctl T)	4 (24)	T (64)	t	64
25	NACK(Ctl U)	5 (25)	U (65)	u	65
26	(Ctl V)	6 (26)	V (66)	v	66
27	(Ctl W)	7 (27)	W (67)	w	67
30	CNCL(Ctl X)	8 (30)	X (70)	x	70
31	(Ctl Y)	9 (31)	Y (71)	y	71
32	SS(Ctl Z)	: (32)	Z (72)	z	72
33	ESC (ALT)	: (33)	[(Sft K)		73
34	(Ctl-Sft L)	< (34)	\(Sft L)		74
35	(Ctl-Sft M)	= (35)](Sft M)	ESC	75
36	(Ctl-Sft N)	> (36)	^(Sft L)	ESC	76
37	(Ctl-Sft O)	? (37)	^(Sft O)	RUB-OUT	77

Note: P=320 Shift P=300 Cntl P=220 Shift & Cntl P=200

PACKED CHARACTER FORMAT AND CONVENTIONS

TTY characters are packed in two octal digits
 The final two bits are added just before outputting
 01 to 37 have 300 added to them
 40 to 77 have 200 added to them

Code	Character	Symbol	Code	Alpha
00	END OF CHARACTER STREAM	⊙		
01-32	ALPHABETIC CHARACTERS		CODE	ALPHA
33	LEFT BRACKET	[(SH K)	01	A
34	BACK SLASH	\(SH L)	02	B
35	RIGHT BRACKET](SH M)	03	C
36	UP POINTING ARROW	↑	04	D
37	LEFT POINTING ARROW	←	05	E
40	SPACE		06	F
41	EXCLAMATION MARK	!	07	G
42	QUOTES	"	10	H
43	NUMBER SIGN	#	11	I
44	DOLLAR SIGN	\$	12	J
45	PERCENT SIGN	%	13	K
46	AMPERSAND	&	14	L
47	APOSTROPHE	'	15	M
50	LEFT DELIMETER	(16	N
51	RIGHT DELIMETER)	17	O
52	STAR (MULTIPLY)	*	20	P
53	PLUS SIGN	+	21	Q
54	COMMA	,	22	R
55	MINUS SIGN	-	23	S
56	PERIOD	.	24	T
57	RIGHT SLASH (DIVIDE)	/	25	U
60-67	NUMBERS 0-7		26	V
70-71	NUMBERS 8&9		27	W
72	COLON	:	30	X
73	SEMICOLON	;	31	Y
74	LESS THAN	<	32	Z
75	EQUAL SIGN	=		
76	GREATER THAN	>		
77	QUESTION MARK	?		

When a parameter is preceded by a left paren, it is an absolute constant. It will be defined somewhere in memory, even though it won't necessarily show up on the listings.

When a string of code such as ".ASCII / /" is encountered in the listings it shows that the following locations are in the packed ASCII 6 bit notation packed 3 to a word.

MAG TAPE IOTS

/HARDWARE MAG TAPE IOTS

WC-32 Word count (two's complement)
CA-33 Address - 1 of data
MTRS-707352 Mag tape read status
MTAF-707322 Mag tape clear all flags
MTSF-707341 Mag tape skip on flag
MTCR-707321 Mag tape skip on controller ready
MTTR-707301 Mag tape skip on transport ready
MTCM-707324 Mag tape "OR" command register
MTLC-707326 Mag tape load command register
MTRC-707312 Mag tape read command register
MTGO-707304 Mag tape go
MTCC-707356 Mag tape clear command register

/SOFTWARE TAPE CONTROLLER

TCTR-717301 Skip on transport ready
TCTR-717301 Skip on transport ready
TCGS-717341 Skip on gap detected
TCBS-717321 Skip on byte read
TCLG-707302 Load command and go
resets "DATA OVER" , "RECORD GAP" and "EOT"
clears LRC register
1 Unit select
2 Unit select (no unit 0)
4 Odd parity
5,6 & 7 Command register same as hardware TC
TCRS-707372 Read status
200k 1 Rewinding
100k 2 Load point
40K 3 File protect
20K 4 Parity error
4K 6 EOT
400 9 Data late <read only>
20 13 Phase drive
TCWL-707361 Write LRC
Data is transmitted via lines 9-17, 17 is chan 8
TCRD-707332 Read byte
TCWN-707342 Write NRZ byte
TCWP-717342 Write phase byte
27.8 usec <30.87 machine cycles> for 800 BPI

● /SEVEN TRACK FORMAT(CORE DUMP MODE)

1st character * 2nd character * 3rd character
00 01 02 03 04 05 06 07 08 09 10 11 12 13 14 15 16 17
CHANNEL 2**3**4**5**6**7 2**3**4**5**6**7 2**3**4**5**6**7

/NINE TRACK FORMAT

1st character * 2nd character
00 01 02 03 04 05 06 07 08 09 10 11 12 13 14 15 16 17
CHANNEL P1*P2**0**1**2**3**4**5**6**7 0**1**2**3**4**5**6**7

/HEAD LAYOUTS

	OUT	2	3	4	5	6	7	8	INN
9 TRK	2	0	4	P	5	6	7	1	3
7 TRK	0	1	2	3	4	5	P		

7 Track Filemark is an octal 17 with even parity
9 Track Filemark is an octal 23 with odd parity

MAG TAPE STATUS AND COMMAND WORDS

/MAGNETIC TAPE CONTROLLER STATUS WORD

00 01 02 03 04 05 06 07 08 09 10 11 12 13 14 15 16 17

00..400K.	ERROR /	↑	↑	↑	↑	↑	↑	↑	↑	↑	↑		↑	↑	↑	↑
01..200K.	REWINDING/	↑	↑	↑	↑	↑	↑	↑	↑	↑	↑		↑	↑	↑	↑
02..100K.	BOT	/	↑	↑	↑	↑	↑	↑	↑	↑	↑		↑	↑	↑	↑
03..40K..	ILLEGAL CMND ..	/	↑	↑	↑	↑	↑	↑	↑	↑	↑		↑	↑	↑	↑
04..20K..	PARITY ERROR	/	↑	↑	↑	↑	↑	↑	↑	↑	↑		↑	↑	↑	↑
05..10K..	END OF FILE (EOF) ...	/	↑	↑	↑	↑	↑	↑	↑	↑	↑		↑	↑	↑	↑
06..4K...	END OF TAPE (EOT)	/	↑	↑	↑	↑	↑	↑	↑	↑	↑		↑	↑	↑	↑
07..2K...	READ-COMPARE ERROR	/	↑	↑	↑	↑	↑	↑	↑	↑	↑		↑	↑	↑	↑
08..1K...	RECORD LENGTH INCORRECT	/	↑	↑	↑	↑	↑	↑	↑	↑	↑		↑	↑	↑	↑
09..400..	DATA REQUEST LATE	/	↑	↑	↑	↑	↑	↑	↑	↑	↑		↑	↑	↑	↑
10..200..	BAD TAPE (1.5 CHAR TIMES)	/	↑	↑	↑	↑	↑	↑	↑	↑	↑		↑	↑	↑	↑
11..100..	TAPE FLAG, JOB DONE	/	↑	↑	↑	↑	↑	↑	↑	↑	↑		↑	↑	↑	↑
12..40...		↑	↑	↑	↑	↑	↑	↑	↑	↑		↑	↑	↑	↑
13..20...		↑	↑	↑	↑	↑	↑	↑	↑	↑		↑	↑	↑	↑
14..10...	SEVEN TRACK DRIVE	/	↑	↑	↑	↑	↑	↑	↑	↑	↑		↑	↑	↑	↑
15..4....	TWO EXTRA CHARACTERS	/	↑	↑	↑	↑	↑	↑	↑	↑	↑		↑	↑	↑	↑
16..2....	ONE EXTRA CHARACTERS	/	↑	↑	↑	↑	↑	↑	↑	↑	↑		↑	↑	↑	↑
17..1....	III HARDWARE TAPE CONTROLLER (ALWAYS ON)	/	↑	↑	↑	↑	↑	↑	↑	↑	↑		↑	↑	↑	↑

/MAGNETIC TAPE CONTROLLER COMMAND WORD

00 01 02 03 04 05 06 07 08 09 10 11 12 13 14 15 16 17

00..400K.	DECK ADR../	↑	↑	↑	↑	↑	↑	↑	↑	↑	↑		↑	↑	↑	↑
01..200K.	DECK ADR../	↑	↑	↑	↑	↑	↑	↑	↑	↑	↑		↑	↑	↑	↑
02..100K.	DECK ADR../	↑	↑	↑	↑	↑	↑	↑	↑	↑	↑		↑	↑	↑	↑
03..40K..	ODD PARITY.....	/	↑	↑	↑	↑	↑	↑	↑	↑	↑		↑	↑	↑	↑
04..20K..	CORE DUMP MODE.....	/	↑	↑	↑	↑	↑	↑	↑	↑	↑		↑	↑	↑	↑
05..10K..	THREE INCH GAP.....	/	↑	↑	↑	↑	↑	↑	↑	↑	↑		↑	↑	↑	↑
06..4K...	COMMAND CODE.....	/	↑	↑	↑	↑	↑	↑	↑	↑	↑		↑	↑	↑	↑
07..2K...	COMMAND CODE.....	/	↑	↑	↑	↑	↑	↑	↑	↑	↑		↑	↑	↑	↑
08..1K...	COMMAND CODE.....	/	↑	↑	↑	↑	↑	↑	↑	↑	↑		↑	↑	↑	↑
09..400..	INTERRUPT ENABLE-DISABLE.....	/	↑	↑	↑	↑	↑	↑	↑	↑	↑		↑	↑	↑	↑
10..200..	DENSITY CODE.....	/	↑	↑	↑	↑	↑	↑	↑	↑	↑		↑	↑	↑	↑
11..100..	DENSITY CODE.....	/	↑	↑	↑	↑	↑	↑	↑	↑	↑		↑	↑	↑	↑
	0 = 200 BPI 2 = 800 BPI		↑	↑	↑	↑	↑	↑	↑	↑	↑		↑	↑	↑	↑
	1 = 556 BPI 3 = 800 BPI-9 TRK		↑	↑	↑	↑	↑	↑	↑	↑	↑		↑	↑	↑	↑
12..40...	/	↑	↑	↑	↑	↑	↑	↑	↑	↑		↑	↑	↑	↑
13..20...	/	↑	↑	↑	↑	↑	↑	↑	↑	↑		↑	↑	↑	↑
14..10...	/	↑	↑	↑	↑	↑	↑	↑	↑	↑		↑	↑	↑	↑
15..4....	WRITE TWO EXTRA CHARACTERS	/	↑	↑	↑	↑	↑	↑	↑	↑	↑		↑	↑	↑	↑
16..2....	WRITE ONE EXTRA CHARACTERS	/	↑	↑	↑	↑	↑	↑	↑	↑	↑		↑	↑	↑	↑
17..1....	/	↑	↑	↑	↑	↑	↑	↑	↑	↑		↑	↑	↑	↑

Note: Only the "Universal Tape Controller" can us bits 15 & 16

/5,6 & 7 COMMAND REGISTER

0	NOP	1	REWIND
2	READ	3	READ-COMPARE
4	WRITE	5	WRITE EOF
6	READ FORWARD	7	READ REVERSE

FIXED HEAD DISK

/IOT'S

DDCR=705701	Skip on controller ready
DDSG=705721	Skip on good data
DDSF=705741	Skip on controller flag
DDLA=705726	Load data address (-1) from accumulator
DDLW=705744	Load word count (2's complement) from accumulator
	The DDLA must be done before the DDLW
DDLIC=705766	Load the command register from the accumulator
DDCF=705762	Clear flags
DDRS=705712	Read the disc status register to the accumulator

/DISC COMMAND WORD [DDLC]

	00	01	02	03	04	05	06	07	08	09	10	11	12	13	14	15	16	17
04.. 20K.. INTERRUPT.....	/				↑	↑	↑	↑	↑		↑	↑		↑	↑			↑
05.. 10K.. WRITE.....					/	↑	↑	↑	↑		↑	↑		↑	↑			↑
06.. 4K... GO.....						/	↑	↑	↑		↑	↑		↑	↑			↑
07.. 2K... SET FLAG.....							/	↑	↑		↑	↑		↑	↑			↑
X HEAD (3).....								/			/	↑		↑	↑			↑
Y HEAD (3).....												/		/	↑			↑
SECTOR (4).....															/			/

Note: Disc address (both same, disc zero selected)

/DISC STATUS REGISTER (I15 DISC'S ONLY) [DDRS]

	00	01	02	03	04	05	06	07	08	09	10	11	12	13	14	15	16	17
04.. 20K.. INTERRUPT.....	/	↑	↑	↑	↑	↑	↑	↑	↑	↑	↑	↑		↑	↑			↑
05.. 10K.. WRITING.....					/	↑	↑	↑	↑	↑	↑	↑		↑	↑			↑
06.. 4K... BUSY.....						/	↑	↑	↑	↑	↑	↑		↑	↑			↑
07.. 2K... FLAG.....							/	↑	↑	↑	↑	↑		↑	↑			↑
08.. 1K... ACTIVE.....								/	↑	↑	↑	↑		↑	↑			↑
09.. 400.. CHECKSUM ERROR.....									/	↑	↑	↑		↑	↑			↑
10.. 200..										/	↑	↑		↑	↑			↑
unused (3).....											/			/	↑			↑
CURRENT SECTOR (4).....															/			/

FR-80 DEFINITIONS

RST=706002	Reset page two	
INTS=706022	Intensify a spot (2 usec)	
CRT=706042	Carraige return	
UNBL=706062	Unblank PLS	
BLNK=706064	Blank pls	
LSL=706104*	Load smaller over larger	
	Sign of smaller is bit 5 (1=neg)	
	Y greater than X is bit 4	
LDL=706164*	Load larger (Starts vector generator)	
	Sign of larger is bit 5 (1=neg)	
VGO=706144	Vector go	
LXD=706124	Load X dac	
LXB=706204	Load X buffer	
LYD=706224	Load Y dac	
LYB=706004	Load Y buffer	
LXDI=706244	Load X dac and intensify	
LYDI=706264	Load Y dac and intensify	
ADV=706304	Advance	
CHGO=706324	Character go	
SPC=706024	Space	
LSPS=706344	Load spot size (requires 100usec for cap)	
LBRT=706364	Load brightness	
LSIZ=706404	Load characer size	
LROT=706424	Load rotation word	
	13=X space 14=Y space	
LCDY=706444	Load X character spacing	
LCDX=706464	Load Y character spacing	
SFLM=706001	Skip on film in camera	
SM10=706021	Skip on 10 feet left	
SPNB=706041	Skip on plotter not busy	
SCNB=706141	Skip on cut not busy	
SFNA=706061	Skip on film not advancing	
LXBG=716104	Load X buffer and go	
LYBG=716164	Load Y buffer and go	
SVON=716541	Skip on video on	
LRA=706504**	Load selected register address	
VGRS=706552**	Read vector generator status	
100 11	CGRUN (13-2)	
40 12	CGREQ (13-2) VG wants stroke or spacing info	
20 13	CLKEN (18-21)	
10 14	VGREQ (18-2)	
4 15	KLOCENB (14-3) (18-2old) Enable stroke clock	
2 16	-PANIC (18-2) Beam settling or character spacing	
1 17	-VGEN (18-1)	
RDVG=706512**	Read vector generator <X axis>	
RDVG2=706532**	Read vector generator <Y axis>	
VCON=706501**	Vector continue	
LRA	RDVG	RDVG2
00	X (16)	Y (16)
02	AR (16)	000000
04	unused	unused
06	CGMX (16)	CGMY (16)
10	BR (16)	Q (16)
12	SPD (7)	QS (9)
14	XT (16)	YT (16)
16	SU (7)	unused

* Old FR80's only

** DSA's only

FR80 IOT'S

/HIGH SPEED CHAR GENERATOR

HSLA=701722 Load initial bye address
HSLC=701704 Load character count <two's complement>
HSGO=701724 Character go
HSRA=701752 Read current byte address
HSRB=701772 Load base address
HSKF=701701 Skip on done flag
HSCF=701764 Continue current byte address
HSLB=701702 Load base address
HSCP=701744 Plot character
HSLG=701726 Load byte address and go

/GRAPHIC ARTS CHARACTER GENERATOR

LXV=716322 Load X velocity (9 bits & sign)
LYV=716302 Load Y velocity (9 bits & sign)

/HIGH RESOLUTION SPACING (DSA'S ONLY)

LCDF=716324 Load stroke fraction (14 bits)
LLSP=716204 Load letterspace

/COLOR IOT'S

SCOL=706061 Skip if filters in place
LCOL=706044 Load color filter

Bit	Additive	Subtractive
17	Clear	Clear
16	Red	Magenta
15	Green	Yellow
14	Blue	Cyan
13	Color correcting filter	

/OPTICAL MERGE

OMSR=706601 Skip on ready
OMRD=706612 Read status

400K 00	Slide in place
200K 01	Flash not completed
100K 02	Flash failure if 01 = 0.
06 - 17	Slide number

OMGO=706604 Flash go (also sets intensity)

400K 00	Do not flash
12 - 17	Flash intensity loaded after flash

* Old FR80's only

** DSA's only

NOTES

Max image size at CRT is 3.277 inches
 CRT to Top Hat is 7.000 inches
 1 Scope Addressable Point @ CRT is .0002 inches
 Pincushin 2 2 2 2 1/2

$$P = K_1 (X_1 + Y_1) + K_2 (X_2 + Y_2)$$

$$X(\text{@ coil}) = X(\text{DAC}) - PX$$

$$Y(\text{@ coil}) = Y(\text{DAC}) - PY$$

Dynamic focus 2 2
 $X (X + Y)$

CAMERA	RESO	SAP-IN	ABUT	X in (SAP)	Y in (SAP)
3-8020 35sp*	E	16718	4	0.748(12507)	0.980(16384)
4-8021 16sp*		24480	4	0.300(7344)	0.409(7889)
2-8022 16un	F	24510	6	0.669(16384)	0.550(13467)
1-8023 35un	G	11430	6	1.433(16384)	1.133(12950)
5-8024 16strip	F	16384	8	1.000(16384)	0.550(9011)
1-8026 35un	I	11430	6	1.433(16384)	1.133(12950)
11-8027 216un	H		2007		
9-8025 24x	F	20808	n-a	0.787(16384)	0.650(13517)
9-8032 42x	B	46282	n-a	0.354(16384)	0.291(13470)
14-8028 5in*	I	3641	2931	3.7(13450)	4.5(16384)
16-8060 12in*	H	1394	3300	8.50(11850)	11.75(16384)

*ROTATION = 1

TANK	FULL REVERSAL	NORMAL
1	DEV	DEV
2	DEV	DEV
3	WASH	WASH
4	BLEACH	FIX
5	CLEAR	FIX
6	WASH	WASH

/CHARACTER CODES

- 00 End (terminate character with no spaces)
- 01-07 Terminate after 1-7 spaces
- 10 Terminate after 8 spaces
- 11 Pen up (video off)
- 12 Pen down (video on)
- 13 N-A
- 14-37 VID (vector identification)
- 40 N-A
- 41-77 Number of increments (2's complement form)

'DISPLAYER' ERRORS & 'PATCH' MODE

/.....FR80 DISPLAYER ERROR TYPEOUTS

DLM	Invalid checkpoint
UNC	Unknown com
NAM	Name undefined
CON	Unknown control character
PAG	Invalid pages
DIO	No disc IO
UNA	Unassigned device
TPW	No tape writing
DKR	Disc read error
DKN	Disc name missing
CNV	Bad convert statement
IOX	IO device exchange
EOF	Tape EOF
CVR	Character not in convert table
DKW	Disc write error
FNM	Font name missing
FNF	Font not found
TMF	Too many fonts
BDF	Bad font format
FNL	Font not loaded

/.....PATCH MODE.....

n = Signed Octal Integer
ca = Current address
cc = Contents of current address

n/ **Type cc and enter patch mode**
n' Start program at n
n! Execute instruction n
n= Types all locations that equal n
n< Exit PATCH MODE at n
n(lf) Type ca+1 contents
n† Type ca-1 contents

Nominal FR80 character heights

These character heights are in scope points

CHAR	P9L-P15	I15	CHAR	P9L-P15	I15
00	15	13	32	174	147
01	19	17	33	181	151
02	23	21	34	186	155
03	27	25	35	191	160
04	32	29	36	196	164
05	36	34	37	202	168
06	41	38	38	209	172
07	44	42	39	216	176
08	50	46	40	220	181
09	55	50	41	226	185
10	59	55	42	232	189
11	63	59	43	239	193
12	68	63	44	247	197
13	73	67	45	252	202
14	77	71	46	256	206
15	82	76	47	272	210
16	89	80	48	276	214
17	92	84	49	281	218
18	98	88	50	290	223
19	103	92	51	298	227
20	107	97	52	305	231
21	113	101	53	309	235
22	117	105	54	316	239
23	126	109	55	322	244
24	133	113	56	330	248
25	138	118	57	340	252
26	141	122	58	347	256
27	146	126	59	351	260
28	150	130	60	356	265
29	154	134	61	361	269
30	159	139	62	366	273
31	164	143	63	370	277

SUMMARY OF FR 80 DATA FORMAT

0000 Escape checkpoint delimiter
000 No operation
001 End job
010 --- not defined
011 --- not defined
100 Start job
101 --- not defined
110 Strip chart -- not implemented (ni)
111 Frame advance
0001 Y coordinate (second word)
0010 X move absolute
0011 Y move absolute
0100 Escape (see below)
0101 Escape (see below)
0110 X move relative
0111 Y move relative
1000 X vector not moving
1001 Y vector not moving
1010 X vector absolute move
1011 Y vector absolute move
1100 X vector relative not move
1101 Y vector relative not move
1110 X vector relative move
1111 Y vector relative move

/ESCAPE CODES (octal)

00 Incremental film advance
01 Repeat
02 Picture name
03 Enter justified type
04 Enter nonjustified type
05 Set intensity
06 Set spot size
07 Set character size
10 Set character rotation
11 Set character spacing
12 Text line spacing
13 Page # or frame # (accounting to film)
14 Select color
15 Plot current point
16 Select vector move
17 Draw arc
20 Control interrupt
21 Verify camera and rotation
22 Draw vector family
23 Set x and y offsets
24 Select character height
25 Define character set
26 Assign character definition to code
27 Output fr 80 supplied variable (ni)
30 Fiche title

FR80 FORMS LANGUAGE (1 of 2)

-13+15 Turn on spot position.
13+3 1 frame advance
13+4 1 frame on crt.
17 Tty echo.

.....
n0 Dumps out to mag tape(n tells # of adv).
Carrriage return gives EOF.
T Reads paper tape.
'h' Relative horizontal units used.
'i' Relative vertical units used.
'j' Absolute horizontal units.
'v' Absolute vertical units.
'm,n' Numbers.
'x' horizontal scope points.
'y' Vertical scope points.

.....
m,nB Intentsity,spot size.
i,jD Set deltas.
nR...R Repeat n times.
nN...N String number n.
iI Move along X.
jJ Move along Y.
x,yP Move to position x,y.
m,nQ Plot a point(m=INT,n=SPOT)
h,xX Set hor units to scope points.
v,yY Set vert units to scope points.
n!L\ Number of 1.5 segments.
j,n!A Clockwise arc.
j= Radius
N= start pos. # of 1.5 segments from vertical.
m,n!G m= green n= blue
m,n=H m= clear n= red

FR80 FORMS LANGUAGE (2 of 2)

/.....VECTORS.....

j,j!V No parameter, solid line.
1 Par = dotted line. j= spacebetween
2 Par = dashed line.
i,jL Vector, don't change to end point.
i,jM Vector, change to end point.
iA Horiz, change to end point.
iH Horiz, don't change to end point.
jU Vert, change to end point.
jV Vert, don't change to end point.

/.....CHARACTERS.....

jC Set char size(use chart,370C=size 63<9L-P15>)
iE Set space value
jF Set line feed value.
x,yS Define space & line feed.
n!C Set text mode.
N=0 unjustified N=1 Left just
N=2 Centered N=3 Right just
i,j" Enter-exit text mode.
A) In unjust arguments illegal.
B)
i
p j
!U Upright characters.
n!R Rotate characters.
N=0 000 degrees N=4 180 degrees
N=1 045 degrees N=5 225 degrees
N=2 090 degrees(default)N=6 270 degrees
N=3 135 degrees N=7 315 degrees
W Set border around text(default 8)
m,nG M= # of digits
Frame Page
N=8 Suppressl1e N=16
N=12 don't supp N=20