
LL		JJ	SSSSSSSSSS	CCCCCCCCCC	GGGGGGGGGG	6666666666	3333333333	8888888888
LL		JJ	SSSSSSSSSSSS	CCCCCCCCCC	GGGGGGGGGGGG	66666666666	333333333333	888888888888
LL		JJ	SS SS	CC CC	GG GG	66 66	33 33	88 88
LL		JJ	SS	CC	GG	66	33	88
LL		JJ	SSSSSSSSSS	CC	GG	6666666666	333	8888888888
LL		JJ	SSSSSSSSSSSS	CC	GG GGGG	66666666666	333	88888888888
LL		JJ	SS SS	CC	GG GGGG	66 66	33	88 88
LL	JJ	JJ	SS SS	CC CC	GG GG	66 66	33 33	88 88
LL	JJ	JJ	SS SS	CC CC	GG GG	66 66	33 33	88 88
LLLLLLLLLLLL	JJJJJJJJJJJ	JJJJJJJJJJJ	SSSSSSSSSSSS	CCCCCCCCCC	GGGGGGGGGGGG	66666666666	33333333333	888888888888
LLLLLLLLLLLL	JJJJJJJJJJJ	JJJJJJJJJJJ	SSSSSSSSSS	CCCCCCCCCC	GGGGGGGGGG	6666666666	3333333333	888888888888

JJ	0000000000	BBBBBBBBBB	7777777777	7777777777	7777777777	9999999999
JJ	000000000000	BBBBBBBBBB	7777777777	7777777777	7777777777	999999999999
JJ	00 00	BB BB	77 77	77 77	77 77	99 99
JJ	00 00	BB BB	77 77	77 77	77 77	99 99
JJ	00 00	BBBBBBBBBB	77 77	77 77	77 77	9999999999
JJ	00 00	BBBBBBBBBB	77 77	77 77	77 77	9999999999
JJ	00 00	BB BB	77 77	77 77	77 77	99 99
JJ	00 00	BB BB	77 77	77 77	77 77	99 99
JJJJJJJJJJJ	000000000000	BBBBBBBBBB	77 77	77 77	77 77	999999999999
JJJJJJJJJJJ	00000000000	BBBBBBBBBB	77 77	77 77	77 77	999999999999

BBBBBBBBBB	IIIIIIIIII	NN NN	8888888888	8888888888	8888888888
BBBBBBBBBB	IIIIIIIIII	NNN NN	888888888888	888888888888	888888888888
BB BB	II	NNNN NN	88 88	88 88	88 88
BB BB	II	NN NN NN	88 88	88 88	88 88
BB BB	II	NN NN NN	88 88	88 88	88 88
BBBBBBBBBB	II	NN NN NN	8888888888	8888888888	8888888888
BBBBBBBBBB	II	NN NN NN	8888888888	8888888888	8888888888
BB BB	II	NN NN NN	88 88	88 88	88 88
BB BB	II	NN NN NN	88 88	88 88	88 88
BB BB	II	NN NN NN	88 88	88 88	88 88
BBBBBBBBBB	IIIIIIIIII	NN NNN	888888888888	888888888888	888888888888
BBBBBBBBBB	IIIIIIIIII	NN NN	888888888888	888888888888	888888888888

PRINT STARTED AT 01:27:04 ON PRT4 , SATURDAY OCTOBER 02, 1976

SLAC CENTER, STANFORD CENTER FOR INFORMATION PROCESSING

UGT 17

```

ISV40 JOB ORIGIN FROM GROUP=LOCAL , DSP=IJP, DEVICE=SYA , 9F8 A
//LJSCG638 JOB LJS$CG, TIME=(1,30), CLASS=E 0.002
/**MAIN LINES=25 0.004
/**MAIN HOLD=OUTPUT
//MCS80 EXEC ASM, ASMTIM='(1,25)', ASMVER=NEW, ASMRGN=300K, 0.022
// ASMPRM='TERM, LINECOUNT(115)', 0.024
// ASMPGM=ASMH, 0.026
// ASMLB5='SYS1.DUMMYC', ASMLB6='WYL.CG.MCS.MACLIB' 0.028
//ASM.SYSGO DD UNIT=SYSDA, DSN=&&PNCH, DISP=(,PASS), SPACE=(TRK,(9,5)) 0.03
//ASM.SYSPRINT DD UNIT=SYSDA, DSN=&&LIST, DISP=(,PASS), SPACE=(CYL,(3,1)) 0.032
//OBJOUT DD DSN=WYL.CG.LJS.OBJ, DISP=(MOD,DELETE), 0.034
// UNIT=DISK, VOL=SER=SCFEV5, SPACE=(TRK,(10,5)) 0.036
//SYSIN DD * 0.038
/* 3044.004
//CLEANUP EXEC PGM=CLEANUP, REGION=76K 3044.006
//STEPLIB DD DSN=WYL.CG.PUB.LOADMODS, DISP=SHR 3044.008
//IN DD DSN=&&LIST, DISP=(OLD,DELETE) 3044.01
//OUT DD SYSOUT=A, DCB=(BLKSIZE=1936, RECFM=FBM) 3044.012
//OBJIN DD DSN=&&PNCH, DISP=(OLD,DELETE) 3044.014
//OBJOUT DD DSN=WYL.CG.LJS.OBJ, UNIT=2314, 3044.016
// VOL=SER=SCFEV5, DISP=(NEW,KEEP), SPACE=(TRK,(10,5),RLSE), 3044.018
// DCB=BLKSIZE=1600 3044.02
//CIAO EXEC PGM=CIAO, COND=EVEN, REGION=76K 3045.002
//STEPLIB DD DSN=WYL.CG.LJS.LOADMODS, DISP=SHR 3045.004
//JOBQ DD DSN=SYS1.SYSJOBQ, DISP=SHR 3045.006

```

```

LOCATE' 7779WYL.CG.MCS.MACLIB .
AL77790E001/WYL0110003
LOCATE' 7779WYL.CG.PUB.LOADMODS .
AL77790E001/WYL0060004
LOCATE' 7779WYL.CG.LJS.LOADMODS .
AL77790E001/WYL0050004
ARIX63 JOB (7779) LJSCG638 IS PRIORITY 09 CLASS B

```

```

AMDS01 JOB 7779 (LJSCG638) IN SETUP ON MAIN=SYB TIME 01:22:49
AMDS02 SYSLIB B USING D WYL011 ON 863
AMDS02 OBJOUT B USING D SCFEV5 ON 232
AMDS02 STEPLIB B USING D WYL006 ON 233
AMDS02 STEPLIB B USING D WYL005 ON 533
LJSCG638 IEF403I LJSCG638 STARTED TIME=01.22.54
LJSCG638 IEF234E D 906, ASP906
*LJSCG638*74 IECASP0 90A IS LJSCG638 ASM MCS80 ASPI0001
*LJSCG638*75 IECASP0 907 IS LJSCG638 A ASM MCS80 SYSTEM
LJSCG638 IEC202E K 90A, 017779, NL, LJSCG638, ASM
*LJSCG638*76 IECASP0 906 IS LJSCG638 A CLEANUP OUT
@LJSCG638@77 INTASPC ASP *SYA @TO LJS LJSCG638 IS ENDING; ASM=8, CLEANUP=0
LJSCG638 IEF404I LJSCG638 ENDED TIME=01.26.35

```

----- OPERATING SYSTEM MESSAGES -----

```

//LJSCG638 JOB LJS$CG, TIME=(1,30), CLASS=E 0.002
//MCS80 EXEC ASM, ASMTIM='(1,25)', ASMVER=NEW, ASMRGN=300K, 0.022
// ASMPRM='TERM, LINECOUNT(115)', 0.024
// ASMPGM=ASMH, 0.026
// ASMLB5='SYS1.DUMMYC', ASMLB6='WYL.CG.MCS.MACLIB' 0.028
//ASM.SYSGO DD UNIT=SYSDA, DSN=&&PNCH, DISP=(,PASS), SPACE=(TRK,(9,5)) 0.03
//ASM.SYSPRINT DD UNIT=SYSDA, DSN=&&LIST, DISP=(,PASS), SPACE=(CYL,(3,1)) 0.032
//OBJOUT DD DSN=WYL.CG.LJS.OBJ, DISP=(MOD,DELETE), 0.034

```

```
// UNIT=DISK,VOL=SER=SCFEV5,SPACE=(TRK,(10,5)) 0.036
//SYSIN DD DSN=88ASPI0001,UNIT=(CTC,,DEFER),VOL=SER=017779, 0.038
// DISP=(OLD,DELETE),DCB=(RECFM=FB,LRECL=80,BLKSIZE=2000,BUFNO=02)
```

```
IEF236I ALLOC. FOR LJSCG638 ASM MCS80
IEF237I STEPLIB ALLOCATED (6C0,535)
IEF237I SYSGO ALLOCATED (232) SYSLIB ALLOCATED (6C0,6C0,6C0,6C0,6C0,863)
IEF237I SYSPRINT ALLOCATED (237) SYSPUNCH ALLOCATED (906)
IEF237I SYSTEM ALLOCATED (907) SYSUT1 ALLOCATED (860)
IEF237I SYSUT2 ALLOCATED (861) SYSUT3 ALLOCATED (860)
IEF237I OBJOUT ALLOCATED (232) SYSIN ALLOCATED (90A)
```

IEF142I - STEP WAS EXECUTED - CCND CODE 0008

```
IEF285I SYS1.DUMMYL KEPT DRUM2 .
IEF285I SYS1.LINKNEW KEPT SYSDV1.
IEF285I SYS76276.T012246.RV001.LJSCG638.PNCH PASSED SCFEV5.
IEF285I SYS1.DUMMYC KEPT DRUM2 .
IEF285I SYS1.DUMMYC KEPT DRUM2 .
IEF285I SYS1.DUMMYC KEPT DRUM2 .
IEF285I SYS1.MAC KEPT DRUM2 .
IEF285I SYS1.DUMMYC KEPT DRUM2 .
IEF285I WYL.CG.MCS.MACLIB KEPT WYL011.
IEF285I SYS76276.T012246.RV001.LJSCG638.LIST PASSED SCFEV4.
IEF285I SYS76276.T012246.RV001.LJSCG638.ASP0A002 DELETED ASP907.
IEF285I SYS76276.T012246.RV001.LJSCG638.R0005585 DELETED WORK01.
IEF285I SYS76276.T012246.RV001.LJSCG638.R0005586 DELETED WORK02.
IEF285I SYS76276.T012246.RV001.LJSCG638.R0005587 DELETED WORK01.
IEF283I WYL.CG.LJS.OBJ NOT DELETED 8 SCFEV5 1.
IEF285I SYS76276.T012246.RV001.LJSCG638.ASPI0001 DELETED 017779.
```

```
IEF373I STEP /ASM / START 76276.0122 STOP 76276.0125 CPU OMIN 21.97SEC STOR VIRT 320K
SMF001I STEP ASM STEP NUMBER= 1 RETURN= 8 DEC
SMF002I DATE= 10/02/76 STRT= 1:22:54.57 STOP= 1:25:04.97 E.T.= 2:10.40 CPU= 0:21.97
SMF003I CPU ID= 168-A SYSTEM= VS2 01.6 MEM REQ= 300K MEM USED= 320K MEMRY FUNC= 0.85
SMF004I CORE= VIRTUAL PAGE INS= 0 PGE OUTS= 0 WAIT= 1:26.78 PAGE DEACT= 0
SMF005I I/O COUNTS 2301/6C0= 1 2314/535= 0 2314/232= 6 2301/6C0= 0
SMF005I I/O COUNTS 2301/6C0= 0 2301/6C0= 0 2301/6C0= 0 2301/6C0= 0
SMF005I I/O COUNTS 3330/863= 0 2314/237= 627 CTC./906= 0 CTC./907= 1
SMF005I I/O COUNTS 3330/860= 239 3330/861= 0 3330/860= 0 2314/232= 0
SMF005I I/O COUNTS CTC./90A= 138
SMF006I I/O TOTALS OTHER= 139 7 TRK= 0 9 TRK= 0 DASD= 873
SMF007I STEP CHARGES OTHER= 0.70 7 TRK= 0.00 9 TRK= 0.00 DASD= 34.92
SMF008I STEP CHARGES MEM= 18.67 CPU= 21.97 TOTAL= 76.25
```

```
//CLEANUP EXEC PGM=CLEANUP,REGION=76K 3044.006
//STEPLIB DD DSN=WYL.CG.PUB.LOADMODS,DISP=SHR 3044.008
//IN DD DSN=88LIST,DISP=(OLD,DELETE) 3044.01
//OUT DD SYSOUT=A,DCB=(BLKSIZE=1936,RECFM=FBM) 3044.012
//OBJIN DD DSN=88PNCH,DISP=(OLD,DELETE) 3044.014
//OBJOUT DD DSN=WYL.CG.LJS.OBJ,UNIT=2314, 3044.016
// VOL=SER=SCFEV5,DISP=(NEW,KEEP),SPACE=(TRK,(10,5),RLSE), 3044.018
// DCB=BLKSIZE=1600 3044.02
```

```
IEF236I ALLOC. FOR LJSCG638 CLEANUP
IEF237I STEPLIB ALLOCATED (233) IN ALLOCATED (237)
IEF237I OUT ALLOCATED (906) OBJIN ALLOCATED (232)
IEF237I OBJOUT ALLOCATED (232)
```

IEF142I - STEP WAS EXECUTED - CCND CODE 0000

```
IEF285I WYL.CG.PUB.LOADMODS KEPT WYL006.
IEF285I SYS76276.T012246.RV001.LJSCG638.LIST DELETED SCFEV4.
IEF285I SYS76276.T012246.RV001.LJSCG638.ASP0A003 DELETED ASP905.
```

```

IEF285I   SYS76276.T012246.RV001.LJSCG638.PNCH
IEF285I   WYL.CG.LJS.OBJ
IEF373I   STEP /CLEANUP / START 76276.0125   STOP 76276.0126 CPU
SMF001I   STEP CLEANUP   STEP NUMBER=      2
SMF002I   DATE= 10/02/76   STRT= 1:25:05.04   STOP= 1:26:33.95   E.T.= 1:28.91   RETURN= 0 DEC
SMF003I   CPU ID= 168-A   SYSTEM= VS2 01.6   MEM REQ= 76K   MEM USED= 64K   CPU= 0:00.64
SMF004I   CORE= VIRTUAL   PAGE INS= 0   PGE OUTS= 0   WAIT= 1:22.15   MEMRY FUNC= 0.75
SMF005I   I/O COUNTS     2314/233= 3   2314/237= 629   CTC./906= 293   PAGE DEACT= 0
SMF005I   I/O COUNTS     2314/232= 11
SMF005I   I/O TOTALS     OTHER= 293   7 TRK= 0   9 TRK= 0   DASD= 651
SMF007I   STEP CHARGES   OTHER= 1.46   7 TRK= 0.00   9 TRK= 0.00   DASD= 26.04
SMF008I   STEP CHARGES   MEM= 0.48   CPU= 0.64   TOTAL= 28.60
//CIAO   EXEC PGM=CIAO,COND=EVEN,REGION=76K
//STEPLIB DD DSN=WYL.CG.LJS.LOADMODS,DISP=SHR
//JOBQ   DD DSN=SYS1.SYSJOBQE,DISP=SHR
IEF236I   ALLOC. FOR LJSCG638 CIAO
IEF237I   STEPLIB ALLOCATED (533)   JOBQ ALLOCATED (7D0)

```

IEF142I - STEP WAS EXECUTED - CCND CODE 0020

```

IEF285I   WYL.CG.LJS.LOADMODS
IEF285I   SYS1.SYSJOBQE
IEF373I   STEP /CIAO / START 76276.0126   STOP 76276.0126 CPU
SMF001I   STEP CIAO   STEP NUMBER=      3
SMF002I   DATE= 10/02/76   STRT= 1:26:33.96   STOP= 1:26:35.29   E.T.= 0:01.33   RETURN= 20 DEC
SMF003I   CPU ID= 168-A   SYSTEM= VS2 01.6   MEM REQ= 76K   MEM USED= 64K   CPU= 0:00.04
SMF004I   CORE= VIRTUAL   PAGE INS= 0   PGE OUTS= 0   WAIT= 0:00.24   MEMRY FUNC= 0.75
SMF005I   I/O COUNTS     2314/533= 3   2305/7D0= 0
SMF006I   I/O TOTALS     OTHER= 0   7 TRK= 0   9 TRK= 0   DASD= 3
SMF007I   STEP CHARGES   OTHER= 0.00   7 TRK= 0.00   9 TRK= 0.00   DASD= 0.12
SMF008I   STEP CHARGES   MEM= 0.03   CPU= 0.04   TOTAL= 0.19
IEF375I   JOB /LJSCG638/ START 76276.0122   STOP 76276.0126 CPU   OMIN 22.65SEC

```

AMDS09 JOB 7779 (LJSCG638) IN BREAKDOWN

ASSEMBLER DONE

1911. 5105 GODRAG EQU VGT+X'1FFD'

1-00979 5106+GODRAG XEQU VGT+X'1FFD',,C'U'

IEV044 *** ERROR *** UNDEFINED SYMBOL

1 STATEMENT FLAGGED IN THIS ASSEMBLY 8 WAS HIGHEST SEVERITY CODE

OVERRIDING PARAMETERS- TERM,LINECOUNT(115)

OPTIONS FOR THIS ASSEMBLY

NODECK, OBJECT, LIST, XREF(SHORT), NORENT, NOTEST, BATCH, ALIGN, ESD, RLD, TERM, LINECOUNT(15),

FLAG(0), SYSPARM()

NO OVERRIDING DD NAMES

SYMBOL	TYPE	ID	ADDR	LENGTH	LD ID
VGT17	SD	0001	000000	004100	
SCOPLOAD	LD		001109		0001
VECTOR	LD		001287		0001
CHARPROC	LD		00024C		0001
KEYSTORE	LD		0002B1		0001
KEYDOIT	LD		000267		0001
STUFXMLT	LD		000277		0001
HEXIN	LD		000614		0001
CLEAR	LD		000680		0001
RESUMER	LD		00060C		0001

ASM H V 05 01.23 10/02/76

LOC	OBJECT CODE	ADDR1	ADDR2	STMT	SOURCE STATEMENT	ASM H V 05 01.23 10/02/76
				1	PRINT OFF SUPPRESS LISTING OF MACROS	0.04
				1052	PRINT ON,GEN,NODATA	0.044
				1053	GBLA E@ (256)	0.046
				1054	GBLC &VGTVER	1.
				1055 &VGTVER	SETC 'VGT17'	2.
				1056 &VGTVER	TITLE 'THE VGT - VIDEO GRAPHICS TERMINAL'	3.

LOC	OBJECT CODE	ADDR1	ADDR2	STMT	SOURCE STATEMENT	ASM H V 05 01.23 10/02/76
1057	*				*****	4.
1058	*					5.
1059	*					6.
1060	*					7.
1061	*			TTTTTT	HH HH EEEEEEE V V GGGGG TTTTTT	8.
1062	*			TT	HH HH EE VV VV GG GG TT	9.
1063	*			TT	HH HH EE V V GG TT	10.
1064	*			TT	HH HH EE VVV GG GGG TT	11.
1065	*			TT	HH HH EE VVV GG GG TT	12.
1066	*			TT	HH HH EEEEEEE V GGGGGG TT	13.
1067	*					14.
1068	*					15.
1069	*					16.
1070	*					17.
1071	*					18.
1072	*				THIS IS THE SOFTWARE FOR THE SLAC-STANFORD VIDED GRAPHICS TERMINAL.	19.
1073	*					20.
1074	*				IT HAS BEEN WRITTEN (IN LATE 1975 AND 1976) BY :	21.
1075	*				LEN SHUSTEK AND	22.
1076	*				ED FRANK AND	23.
1077	*				MARK DE LEMOS	24.
1078	*				WITH SUGGESTIONS FROM OUR FRIENDS	25.
1079	*					26.
1080	*				IT SUPPORTS ALL SORTS OF NEAT THINGS	27.
1081	*					28.
1082	*					29.
1083	*					30.
1084	*				MAILING ADDRESS: COMPUTATION RESEARCH GROUP	31.
1085	*				STANFORD LINEAR ACCELERATOR CENTER	32.
1086	*				P.O. BOX 4349	33.
1087	*				STANFORD, CALIFORNIA 94305	34.
1088	*					35.
1089	*					36.
1090	*					37.
1091	*				THIS SOFTWARE IS WRITTEN USING THE SLAC 8080 ASSEMBLY,	38.
1092	*				WHICH IS A MACRO-ASSEMBLER BASED ON THE IBM 370 ASSEMBLER H.	39.
1093	*				FOR DOCUMENTATION SEE COMPUTATION GROUP TECHNICAL MEMO #174,	40.
1094	*				AVAILABLE FROM SLAC.	41.
1095	*					42.
1096	*				THE VGT IS A BEAST, WHOSE RESPONSIBILITY FOR EXISTENCE BELONGS	43.
1097	*				TO:	44.
1098	*					45.
1099	*				LEN SHUSTEK AND	46.
1100	*				FOREST BASKETT.	47.
1101	*					48.
1102	*				IF IT BYTES, BY ALL MEANS BYTE IT BACK.	49.
1103	*					50.
1104	*				FURTHER INFORMATION ABOUT THE INTERNAL STRUCTURE OF THE VGT IS	51.
1105	*				IN COMPUTATION GROUP TECHNICAL MEMO #175, AVAILABLE FROM SLAC.	52.
1106	*					53.
1107	*				*****	54.

LOC	OBJECT CODE	ADDR1	ADDR2	STMT	SOURCE STATEMENT	ASM H V 05 01.23 10/02/76
000000				1109	*DEFINITIONS	56.
				1110	&VGTVER CSECT	57.
				1111	ASCII , DEFINE ASCII CHARACTER SET	57.002
				1148	PUNCH '** &VGTVER ASSEMBLED ON &SYSDATE @ &SYSTIME'	58.
		00020		1149	@SP EQU &(C')	59.
				1151	*INCLUDE #VGTDEFNS USER VGT GRD CG ON CAT	60.
				1152	*****	60.001
				1153	*	* 60.002
				1154	* THE VGT - VIDEO GRAPHICS TERMINAL	* 60.003
				1155	*	* 60.004
				1156	* **** HARDWARE DEFINITIONS ****	* 60.005
				1157	*	* 60.006
				1158	*	* 60.007
				1159	*****	60.008
				1160	* UPDATE LIST 23 MAY 76 LJS	60.009
				1161	* 3 AUG 76 MRD	60.01
				1162	* 5 AUG 76 LJS	60.011
				1163	* 30 AUG 76 EHF (ADDED MODESET2)	60.012
				1164	* 4 SEP 76 LJS (ADDED WRAP ADDRESSES)	60.013
				1165	*	60.014
				1166	* MEMORY MAP	60.015
				1167	*	60.016
00400				1168	K EQU 1024	60.017
00000				1170	ROM EQU 0*K	60.018
02000				1172	CPURAM EQU 8*K	60.019
00400				1174	CPURAMSZ EQU 1*K	60.02
02800				1176	CHGENROM EQU 10*K	60.021
03000				1178	CHGENRAM EQU 12*K	60.022
04000				1180	RAM EQU 16*K	60.023
04400				1182	LWRAPADR EQU 17*K	60.024
07400				1184	HWRAPADR EQU 29*K	60.025
				1186	*	60.026
				1187	*	60.027
				1188	* I/O SYMBOLS MARKED '(PORT)' ARE I/O PORTS,	60.028
				1189	* OTHERS ARE BIT-WITHIN-PORT DEFINITIONS.	60.029
				1190	*	60.03
				1191	*	60.031
				1192	* INPUT PORTS	60.032
				1193	*	60.033
00084				1194	KEYBOARD EQU X'84' (PORT)	60.034
00085				1196	STATBITS EQU X'85' (PORT)	60.035
00010				1198	KBNEWCHR EQU X'10'	60.036
00008				1200	FRAMECNT EQU X'08'	60.037
00004				1202	FRAMEINT EQU X'04'	60.038
00002				1204	KBATTN EQU X'02'	60.039
00001				1206	KBRPT EQU X'01'	60.04
00041				1208	URTSTAT EQU X'41' (PORT)	60.041
00001				1210	URTTXRDY EQU X'01'	60.042
00002				1212	URTRXRDY EQU X'02'	60.043
00038				1214	URTRERR EQU B'00111000'	60.044
00020				1216	URTRVBK EQU B'00100000'	60.045
00001				1218	URTRCV EQU X'01' (PORT)	60.046
00086				1220	ATODVAL EQU X'86' (PORT)	60.047
00087				1222	PARIN EQU X'87' (PORT)	60.048
00001				1224	PRTREADY EQU X'01'	60.049

00083 1226 SCRWHEEL EQU X*83* (PORT)

SCROLLING WHEEL POSITION

60.05

LOC	OBJECT CODE	ADDR1	ADDR2	STMT	SOURCE STATEMENT	ASM H V 05 01.23 10/02/76
				1229 *		60.052
				1230 *	OUTPUT PORTS	60.053
				1231 *		60.054
00082	1232	RSFRMINT	EQU	X'82' (PORT)	RESET FRAME INTERRUPT	60.055
00083	1234	RSURTINT	EQU	X'83' (PORT)	RESET USART INTERRUPT	60.056
00084	1236	DISADDRH	EQU	X'84' (PORT)	HIGH-ORDER DISPLAY ADDRESS	60.057
00085	1238	DISADDRL	EQU	X'85' (PORT)	LOW-ORDER DISPLAY ADDRESS	60.058
00086	1240	BELL	EQU	X'86' (PORT)	BEEPER	60.059
00087	1242	CHLINE1	EQU	X'87' (PORT)	1ST LINE OF ROW 1 TO DISPLAY	60.06
00080	1244	MODESET	EQU	X'80' (PORT)	CONTROL BITS:	60.061
00080	1246	REVRVID	EQU	X'80'	REVERSE VIDEO CONTROL	60.062
00040	1248	SCRNBLNK	EQU	X'40'	SCREEN BLANKING CONTROL	60.063
00020	1250	NOROMCHR	EQU	X'20'	NO ROM CHARS (RAM ONLY) IN TEXT MODE	60.064
0000C	1252	URTCLOCK	EQU	X'0C'	USART EXTERNAL/INTERNAL CLOCK CONTRL	60.065
00010	1254	GRAPHMD	EQU	X'10'	GRAPH MODE	60.066
00002	1256	QUICKMD	EQU	X'02'	QUICK MODE FOR RAM ACCESS	60.067
00001	1258	SIXTNMD	EQU	X'01'	16 RASTERS/ROW MODE	60.068
000A6	1260	MODESET2	EQU	X'A6' (PORT)	MORE MODEBITS	60.069
00080	1262	HIGHWRAP	EQU	X'80'	WRAP TO X'7600'	60.07
00040	1264	ATODSTRT	EQU	X'40'	START A/D CONVERSION (20 USEC.)	60.071
00041	1266	URTCTL	EQU	X'41' (PORT)	USART CONTROL BITS:	60.072
00057	1268	URTINTRS	EQU	B'01010111'	INTERNAL RESET (TO SET MODE)	60.073
0007A	1270	URTMODE	EQU	B'01111010'	ASYNC, EVEN PARITY, 7BITS, 16XCLK	60.074
00079	1272	URTX1MD	EQU	B'01111001'	ASYNC, EVEN PARITY, 7BITS, 1XCLK--	60.075
0000F	1274	URTBREAK	EQU	B'00001111'	SEND BREAK	60.076
00007	1276	URTRSRK	EQU	B'00000111'	RESET BREAK	60.077
00017	1278	URTRSERR	EQU	B'00010111'	RESET RCV ERR, RCV ENB, DTR, XMT ENB	60.078
00001	1280	URTXMT	EQU	X'01' (PORT)	USART TRANSMITTED CHARACTER	60.079
0008E	1282	URTSPEED	EQU	X'8E' (PORT)	USART BAUD RATE; 4 BITS RCV, 4 XMIT	60.08
00096	1284	KBRESET	EQU	X'96' (PORT)	RESET KB STROBE FF (IE 'KBNEWCHR')	60.081
0009E	1286	ATODSEL	EQU	X'9E' (PORT)	SELECT ANALOG SOURCE BY 1 OF A3-A0	60.082
000AE	1288	KBCLICK	EQU	X'AE' (PORT)	KEYBOARD CLICK	60.083
000B6	1290	PAROUT	EQU	X'B6' (PORT)	PARALLEL DATA OUT	60.084
000BE	1292	PARRESET	EQU	X'BE' (PORT)	PARALLEL OUTPUT RESET	60.085
	1294	*INCLUDE	VGTRAMAP USE VGT GRD CG ON CAT			61.
00080	1295	CMDKEY	EQU	X'80'		61.001

A7

79

LOC	OBJECT CODE	ADDR1	ADDR2	STMT	SOURCE STATEMENT	ASM H V 05 01.23 10/02/76
1298					*****	61.003
1299					* HERE ARE THE COROUTINE MACRO DEFINITIONS	61.004
1300					*****	61.005
1301					MACRO	61.006
1302	&L				CCB &STACKSZ=32	61.007
1303					GBLA &CCBLEN	61.008
1304					AIF (K&L GT 6).ERR LENGTH OF NAME<7	61.009
1305	&L				DS 0X	61.01
1306	&L.SP				DS 2 THIS STACKPTR IS 2 BECAUSE WE DO A	61.011
1307	.*				DOUBLE STORE OF HL TO &L.SP BELOW	61.012
1308	&L.CS				DS 1 POINTER TO CHARACTER SOURCE CCB	61.013
1309	&L.CP				DS 1 POINTER TO CHARACTER PROC. CCB	61.014
1310					DS &STACKSZ HERE IS THE STACK	61.015
1311	&L.ST				DS 0 THE END OF IT	61.016
1312	&CCBLEN				SETA &CCBLEN+&STACKSZ+4 INCREMENT COUNTER	61.017
1313					AIF (&CCBLEN LT 256).OK AND MAKE SURE WE STAY ON ONE PAGE	61.018
1314					MNOTE 8, '***ERROR*** CONTROL BLOCKS EXCEED 255'	61.019
1315	.OK				MEXIT	61.02
1316	.ERR				MNOTE 8, '***ERROR*** COROUTINE NAME TOO LONG'	61.021
1317					MEND	61.022
1319					MACRO	61.024
1320	&LABEL				CCBSET &CCB,&EP=,&CS=,&CP=	61.025
1321					AIF ('&LABEL' EQ '').SKIP	61.026
1322	&LABEL				EQU *	61.027
1323	.SKIP				ANOP	61.028
1324					AIF ('&EP' EQ '').A1	61.029
1325					LODI A,&CCB.ST-6,> SET THE INITIAL ENTRY POINT	61.03
1326					ST A,&CCB.SP PUT IT IN THE STACK POINTER	61.031
1327					LODI HL,&EP SET THE INITIAL ENTRY POINT	61.032
1328					ST HL,&CCB.ST-2	61.033
1329	.A1				AIF ('&CS' EQ '').A2	61.034
1330					LODI A,&CS.SP,>	61.035
1331					ST A,&CCB.CS STORE CHARACTER SOURCE CCB LOCATION	61.036
1332	.A2				AIF ('&CP' EQ '').A3	61.037
1333					LODI A,&CP.SP,>	61.038
1334					ST A,&CCB.CP SIMILARLY FOR CHARACTER PROC.	61.039
1335	.A3				ANOP	61.04
1336					MEND	61.041
1338					MACRO	61.043
1339	&LABEL				RESUME &CURTASK,&NEWTASK	61.044
1340	&LABEL				LODI HL,-6	61.045
1341					ADD HL,SP PREPARE TO STORE BC,DE	61.046
1342					ST HL,&CURTASK.SP	61.047
1343					LODI L,&CURTASK,&NEWTASK,> HL NOW POINTS AT &CCB.CP OR CS	61.048
1344					CALL RESUMER CALL GLOBAL ROUTINE	61.049
1345					MEND	61.05
1347					MACRO	61.052
1348	&LABEL				CALL@ &TASK THIS IS FOR CALL INDIRECTS	61.053
1349	&LABEL				LD HL,&TASK GET THE ROUTINE ADDRESS	61.054
1350					CALL JUMPER AND CALL GLOBAL ROUTINE	61.055
1351					MEND	61.056

LOC	OBJECT CODE	ADDR1	ADDR2	STMT	SOURCE STATEMENT	ASM H V 05 01.23 10/02/76
				1353	*****	61.058
				1354	*	61.059
				1355	* ASSEMBLY CONSTANTS AND RAM LOCATIONS	61.06
				1356	*	61.061
				1357	*****	61.062
				1358	*	61.063
		04400		1359	ITEXT EQU X'4400'	61.064
		00051		1361	LINESIZE EQU 81	61.065
		00025		1363	NLINES EQU 37	61.066
		00BB5		1365	SCRNSIZE EQU LINESIZE*NLINES	61.067
		058F0		1367	GRAPHEVN EQU X'10000'-(512+15+1)*LINESIZE	61.068
		0AA41		1369	GRAPHODD EQU GRAPHEVN+257*LINESIZE	61.069
		004BF		1371	TOPOFF EQU 15*LINESIZE	61.07
		0000D		1373	CHARHITE EQU 13	61.071
		0BC00		1375	ITXTSIZE EQU X'10000'-ITEXT	61.072
		0000E		1377	IWASTED EQU ITXTSIZE-ITXTSIZE/LINESIZE*LINESIZE	61.073
				1379	*	61.074
000000		02000		1380	ORG CPURAM	61.075
				1382	*	61.076
002000				1383	DISPL CCB	61.077
002000				1385		
002002				1386		
002003				1387		
002004				1388		
		02024		1389		
002024				1390	LOADER CCB	61.078
002024				1392		
002026				1393		
002027				1394		
002028				1395		
		02048		1396		
002048				1397	HEXER CCB	61.079
002048				1399		
00204A				1400		
00204B				1401		
00204C				1402		
		0206C		1403		
00206C				1404	GETCHR CCB	61.08
00206C				1406		
00206E				1407		
00206F				1408		
002070				1409		
		02090		1410		
				1411	*	61.081
		00070		1412	ALIGN 0,256,FILL=	61.082
				1414		
002090				1415	CURSLOC DS 2	61.083
002100				1417	CURSX DS 1	61.084
002102				1419	CURSCTR DS 1	61.085
002103				1421	CURSCHAR DS 1	61.086
002104				1423	PREVCHAR DS 1	61.087
002105				1425	*	61.088
002106				1426	ESCCHAR DS 1	61.089
002107				1428	PRVCCHAR DS 1	61.09
002108				1430	KEYCTR DS 1	61.091

002109		1432	KEYSAVED	DS	1	CHAR TO BE REPEATED	61.092
00210A		1434	NXTCHL1	DS	2	ADDR OF NEXT CHL1TAB ENTRY TO USE	61.093
00210C		1436	NXTDISA	DS	2	NEXT DISPLAY ADDR TO BE LOADED	61.094
00210E		1438	SAVDISA	DS	2	SAVED DISPLAY ADDRESS AT "HOME" TIME	61.095
002110		1440	SCRCTR	DS	1	SCROLL COUNTER	61.096
002111		1442	BRKSTATE	DS	1	IS BREAK BEING ISSUED	61.097
002112		1444	TIME	DS	4	HHMMSSTT, IN DECIMAL	61.098
002116		1446	TIMELC	DS	2	LOCATION OF TIME MSG	61.099
002118		1448	CLRFLG	DS	1	MSB=1 FOR CLR NEEDED, LSB FOR BANNER	61.1
		1450	*			X'CO' FOR CLR TO ZERO, NOT BLANK	61.101
002119		1451	FLAGS	DS	1	MISCELLANEOUS FLAGS	61.102
	00080	1453	FULLDUPL	EQU	X'80'	FULL DUPLEX MODE	61.103
	00001	1455	TABREF	EQU	X'01'	HAVE TABS BEEN CLEAR?	61.104
		1457	*				61.105
		1458	* COROUTINE CONTROL BLOCKS...				61.106
		1459	*				61.107
		1460	*				61.108
00211A		1461	SOURCE	DS	2	CURRENT CHARACTER SOURCE	61.109
00211C		1463	PATSTAT	DS	1	PATCHING STATUS	61.11
	00080	1465	LOUSYPAR	EQU	X'80'	PARITY NO GOOD	61.111
		1467	NOTHEX	EQU	X'40'	NONHEX CHAR RECVD	61.112

LOC	OBJECT CODE	ADDR1	ADDR2	STMT	SOURCE	STATEMENT	ASM H V 05 01.23 10/02/76
				00040		1468	
				00020		1469 TYPERR EQU X'20'	BAD PATCH TYPE 61.113
				00010		1471 CKSMERR EQU X'10'	CHECKSUM BAD 61.114
				00008		1473 PATFAIL EQU X'08'	PATCH DID NOT STICK 61.115
				00001		1475 PATCHING EQU X'01'	INDICATES PATCHING BEING DONE 61.116
00211D						1477 PATCKSM DS 1	CHECKSUM 61.117
00211E						1479 PATCNT DS 1	COUNTER 61.118
						1481 *	61.119
00211F						1482 MODEBITS DS 1	CURRENT MODE BITS (SEE "MODESET") 61.12
002120						1484 PUTPTR DS 2	NEXT FREE BUFFER LOCATIN 61.121
002122						1486 GETPTR DS 2	NEXT BUFFERED CHAR TO PRCESS 61.122
002124						1488 LCLMODE DS 1	SEND TO COMPUTER=0,NOT=ANYTHING 61.123
				00004		1490 NORCV EQU X'04'	61.124
				00002		1492 NOSEND EQU X'02'	61.125
				00020		1494 PATFLAG EQU X'20'	61.126
002125						1496 SAVSTK DS 2	61.127
002127						1498 POINTER DS 2	CURRENT CHARACTER TO BE SENT 61.128
002129						1500 HEXADDR DS 2	61.129
00212B						1502 INTROUT DS 2	61.13
00212D						1504 FOOADDR DS 2	ADDRESS OF INTERRUPT HANDLER 61.131
						1506 *	HANDLER FOR UNDEF. COMMANDS 61.132
						1507 *	PATCH THIS ADDRESS TO GET CONTROL 61.133
						1508 *	FOR ALL KEYBOARD COMMANDS (I.E. 61.134
						1509 *	<CMD X> WHICH ARE NOT DEFINED IN 61.135
						1510 *	THE ROMS. YOU ARE CALLED, SO YOU 61.136
						1511 *	SHOULD DO A RETURN AT THE END. 61.137
						1512 *	ALSO INTERRUPTS ARE DISABLED. 61.138
						1513 *	YOU NEED NOT RESTORE ANY REGISTERS. 61.139
						1514 *	E REGISTER HAS PREVIOUS COMMAND 61.14
00212F						1515 PREVSCRL DS 1	PREVIOUS SCROLLING POSITION 61.141
						1517 *	TO GETCHAR IF GETBUF IS NOT SOURCE 61.142
002130						1518 CRLFOVR DS 1	61.143
002131						1520 CNTLOMD DS 1	IS COPY OF FLAGS FOR FULLDPLX FAKE 61.144
002132						1522 ALLOWGMD DS 1	0=ALLOW GRAPH MODE, NZ = NO GMODE 61.145
						1524 *	61.146
002133						1525 XMITSTAT DS 1	1 IF XMITBUF IS FULL, 0 OTHERWISE 61.147
002134						1527 XMITTYPE DS 1	61.148
002135						1529 XMITGET DS 2	ADD OF WHERE TO GET CHAR TO SND 61.149
002137						1531 XMITPUT DS 2	ADD OF WHERE TO PUT CHAR IN BUF 61.15
002139						1533 XMITSAV DS 2	SAVED XMIT POINTER 61.151
00213B						1535 CRCNT DS 1	NUMBER OF <CR> IN XMIT BUF 61.152
00213C						1537 DC1CNT DS 1	WHETHER OR NOT DC1 HAS BEEN SENT 61.153
						1539 *	(I.E. CLEAR TO SEND) 61.154
						1540 *	MUST FOLLOW CRCNT 61.155
00213D						1541 SHIFTMD DS 1	WHICH CHAR SET TO USE 61.156
00213E						1543 XMITHAN DS 2	ADD OF XMITING ROUTINE 61.157
002140						1545 HTABTAB DS 10	TABTABLE 61.158
						1547 *****	61.159
						1548 *	61.16
						1549 *	61.161
						1550 *	61.162
						1551 *	61.163
						1552 *****	61.164
00214A						1553 TEKXXX DS 0X	STORGE FOR 5-BIT COORD ADDRESSES 61.165
00214A						1555 TEKXHI DS 1X	(ORDER-DEPENDENT) 61.166

00214B	1557	TEKXLOW	DS	1X		61.167
00214C	1559	TEKYHI	DS	1X		61.168
00214D	1561	TEKYLOW	DS	1X		61.169
	1563	*				61.17
00214E	1564	GRAPHPOS	DS	4X	CURRENT BEAM POSITION (10 BIT NOS.)	61.171
	1566	*				61.172
002152	1567	#QUADS	DS	X	# OF QUADRANTS FOR THE VECTOR	61.173
	1569	*				61.174
002153	1570	VECTEMP	DS	5X	TEMP SAVE FOR VECTOR RTN	61.175
002158	1572	GRAPHTEM	DS	4X	TEMP SAVE FOR GRAPH RTN	61.176
	1574	*				61.177
00215C	1575	CHARPOS	DS	2X		61.178
00215E	1577	DIREC1	DS	1X		61.179
00215F	1579	DIREC2	DS	1X		61.18
002160	1581	TYPE	DS	1X		61.181

LOC	OBJECT CODE	ADDR1	ADDR2	STMT	SOURCE	STATEMENT	ASM H V 05 01.23 10/02/76
002161			02102	1583	XOFF	EQU CURSX	61.182
				1585		DS 1X	61.183
				1587	*		61.184
			058F0	1588	EVENADD	EQU GRAPHEVN	61.185
		0AA41		1590	ODDADD	EQU GRAPHODD	61.186
002162			05151	1592	NEXT	EQU ODDADD-EVENADD	61.187
				1594	EVENODD	DS 1X	61.188
			02800	1596	ROMGEN	EQU CHGENROM	61.189
				1598	*		61.19
002163				1599	XMASK	DS 1X	61.191
002164				1601	MARGIN	DS 1X	61.192
			02104	1603	GRPOLD	EQU CURSCHAR	61.193
002165				1605	GRPCSET	DS 2X	61.194
002167				1607	TEXT	DS 2	61.195
002169				1609	MTEXT	DS 2	61.196
00216B				1611	TEXTTOP	DS 2	61.197
00216D				1613	TEXTBOT	DS 2	61.198
00216F				1615	WASTED	DS 2	61.199
002171				1617	GRPDRAW	DS 3	61.2
				1619	*		61.201
			00008	1620	CURSTIME	EQU 8	61.202
			00014	1622	KEYRPTD	EQU 20	61.203
			00003	1624	KEYRPTR	EQU 3	61.204
			00004	1626	SCRQTIME	EQU 4	61.205
			00006	1628	SCRSTIME	EQU 6	61.206
			00030	1630	SCRPTIME	EQU 48	61.207
				1632	*		61.208
			01C00	1633	HELP INFO	EQU ROM+7*K	61.209
				1635	*		61.21
				1636	*		61.211
			023FF	1637	STACK	EQU CPURAM+CPURAMSZ-1	61.212
				1639	*		61.213
				1640	*		61.214
				1641	*****		61.215
				1642	*****		61.216
				1643	*		61.217
				1644	*	END OF CPU RAM VARIABLES	61.218
				1645	*	ALL RAM VARIABLES SHOULD GO BEFORE THIS COMMENT	61.219
				1646	*		61.22
				1647	*	THE LABEL 'ENDCPRAM'	61.221
				1648	*	SHOULD BE USED BY PROGRAMS TO ORG THEIR OWN VARIABLES	61.222
			02174	1649	ENDCPRAM	EQU *	61.223
				1651	*		61.224
				1652	*****		61.225
				1653	*****		61.226
002174			04000	1654	ORG	RAM	61.227
004000				1656	XMITBUF	DS 256	61.228
			04100	1658	XMITEND	EQU *	61.229
				1660	*		61.23
				1661	*		61.231
			04100	1662	BUFFER	EQU *	61.232
				1664	*		61.233
			04400	1665	BUFEND	EQU X*4400	61.234
				1667	*		61.235

12
20

LCC	OBJECT CODE	ADDR1	ADDR2	STMT	SOURCE	STATEMENT	ASM H V 05 01.23 10/02/76
				1669		MACRO	61.237
				1670	&LABEL	CMDEF &CHAR,&ROUTINE	61.238
				1671	.*		61.239
				1672	.*	DEFINE AN ENTRY IN THE COMMAND DEFINITION TABLE	61.24
				1673	.*		61.241
				1674		AIF ('&LABEL' EQ '').NOLABEL	61.242
				1675	&LABEL	EQU *-2	61.243
				1676	.NOLABEL	ANOP	61.244
				1677		DC X'&CHAR',AL2(&ROUTINE-&SYSECT)	61.245
				1678		MEND	61.246
				1680		MACRO	61.248
				1681	.*		61.249
				1682	.*	MULTIPLY	61.25
				1683	.*		61.251
				1684	&NAME	MULT &A	61.252
				1685	&NAME	DS 0X	61.253
				1686		AIF ('&A' NE '80').CHK81	61.254
				1687		ADD HL,HL	61.255
				1688		ADD HL,HL	61.256
				1689		ADD HL,HL *8	61.257
				1690		ADD HL,HL *16	61.258
				1691		LOD D,H	61.259
				1692		LOD E,L SAVE IT	61.26
				1693		ADD HL,HL	61.261
				1694		ADD HL,HL *64	61.262
				1695		ADD HL,DE	61.263
				1696		MEXIT	61.264
				1697	.CHK81	AIF ('&A' NE '81').ERROR	61.265
				1698		LOD DE,HL	61.266
				1699		ADD HL,HL	61.267
				1700		ADD HL,HL	61.268
				1701		ADD HL,HL	61.269
				1702		ADD HL,HL	61.27
				1703		XCH HL,DE	61.271
				1704		ADD HL,DE	61.272
				1705		XCH HL,DE	61.273
				1706		ADD HL,HL	61.274
				1707		ADD HL,HL	61.275
				1708		ADD HL,DE	61.276
				1709		MEXIT	61.277
				1710	.ERROR	MNOTE 12,'NOT MULT BE 80 OR 81'	61.278
				1711		MEND	61.279
				1713		MACRO	61.281
				1714	&LABEL	DSALFINE	61.282
				1715	.*	THIS IS A HUMOROUS MACRO WHICH MERELY COUSES A SUBROUTINE	61.283
				1716	.*	TO REPEAT ITSELF BEFORE RETURNING.	61.284
				1717		AIF ('&LABEL' EQ '').NOLABEL	61.285
				1718	&LABEL	EQU *	61.286
				1719	.NCLABEL	ANOP	61.287
				1720		CALL *+3	61.288
				1721		MEND	61.289
				1722	*		62.

1723 *	LUSER ENTRY POINTS. THESE WILL (BY USING THE EXECFILE #AEQU)	63.
1724 *	BE CHANGED INTO EQU STATEMENTS FOR USE IN OTHER PROGRAMS	64.
1725 *	(VIA THE *INCLUDE FACILITY IN A#ACALL). THEY PROVIDE THE ONLY	65.
1726 *	KNOWN LINKAGE INTO COMMONLY USED VGT ROM ROUTINES. IF THERE	66.
1727 *	IS SOME ROUTINE WHICH YOU THINK SHOULD HAVE AN ENTRY POINT	67.
1728 *	DEFINED FOR IT, YOU MIGHT ADD IT YOURSELF (WHICH IS, OF COURSE,	68.
1729 *	FROWNED UPON BY THE HIGHER UPS) OR WRITE TO:	69.
1730 *		70.
1731 *	VGT ENTRY POINT OFFER	71.
1732 *	P.O. BOX A	72.
1733 *	GRAND RAPIDS, MICH 48075	73.
1734 *		74.
1735 *	ALLOW 6 TO 8 WEEKS FOR DELIVERY	75.
1736 *	OFFER VOID WHERE PROHIBITED BY LAW	76.
1737 *	THE DECISION OF THE JUDGES IS FINAL.	77.
1738 *	ALL JUDGING IS PERFORMED BY AN OUTSIDE CONSULTING	78.
1739 *	FIRM, FOR WHOM WE ARE PAYING THROUGH THE NJSE	79.
1740 *	ALL EMPLOYEES OF SLAC OR MEMBERS THEIR IMMEDIATE FAMILIES	80.
1741 *	MAY NOT ENTER THIS COMPETITION.	81.
1742 *		82.
1743	ENTRY SCOPLOAD,VECTOR	83.
1744	ENTRY CHARPROC,KEYSTORE,KEYDOIT	84.
1745	ENTRY STUFXMIT	85.
1746	ENTRY HEXIN	86.
1747	ENTRY CLEAR	87.
1748	ENTRY RESUMER	88.
1749 *		89.

LOC	OBJECT CODE	ADDR1	ADDR2	STMT	SOURCE STATEMENT	ASM H V 05 01.23 10/02/76
				1751	*****	91.
				1752	*	92.
				1753	* BASIC INTERRUPT PROCESSING	93.
				1754	*	94.
				1755	*****	95.
				1756	*	96.
				1757	* MASTER RESET INTERRUPT	97.
				1758	*	98.
				1759	* (THIS CODE MUST NOTE GO BEYOND LOCATION X'35')	99.
				1760	*	100.
004100		00000		1761	ORG 0	101.
000000	31FF23			1763	POWERUP LODI SP,STACK	102.
000003	3E79			1765	LODI A,URTXIMD	103.
000005	D341			1767	OUT URTCTL	104.
000007	3E17			1769	LODI A,URTRSERR	105.
000009	D341			1771	OUT URTCTL	106.
00000B	CD9501			1773	CALL RCVCLR	107.
00000E	CDA401			1775	CALL XMITRCLR	108.
000011	211521			1777	LODI HL,X'2115'	109.
000014	221221			1779	ST HL,TIME	110.
000017	3E81			1781	LODI A,X'81'	111.
000019	321821			1783	ST A,CLRFLG	112.
00001C	218B01			1785	LODI HL,@TODCI	113.
00001F	223E21			1787	ST HL,XMITHAN	114.
000022	97			1789	SUB A	115.
000023	321F21			1791	ST A,MODEBITS	116.
000026	323421			1793	ST A,XMITTYPE	117.
000029	3E0C			1795	LODI A,X'0C'	118.
00002B	D380			1797	OUT MODESET	119.
00002D	3E1B			1799	LODI A,@ESC	120.
00002F	320621			1801	ST A,ESCCHAR	121.
000032	C39007			1803	JMP RESETH	122.
				1805	*	123.
				1806	* FRAME AND/OR USART INTERRUPT	124.
				1807	*	125.
000035		00038		1808	ORG 7*8	126.
000038	E5			1810	INTERRPT PUSH HL	127.
000039	2A2B21			1812	LD HL,INTROUT	128.
00003C	E3			1814	XCH HL,(SP)	129.
00003D	C9			1816	RET U	130.
		0003E		1818	ROM INT EQU *	131.
00003E	F5			1820	PUSH FA	132.
00003F	E5			1822	PUSH HL	133.
000040	D5			1824	PUSH DE	134.
000041	C5			1826	PUSH BC	135.
000042	DB85			1828	INP STATBITS	136.
000044	E604			1830	ANDI FRAMEINT	137.
000046	CACC01			1832	JMP Z,INTURT	138.
000049	D382			1834	OUT RSFRMINT	139.
				1836	*	140.
00004B	DB85			1837	INP STATBITS	141.
00004D	E608			1839	ANDI FRAMECNT	142.
00004F	2A0A21			1841	LD HL,NXTCHL1	143.
000052	7E			1843	LD A,(HL)	144.
000053	C25A00			1845	JMP NZ,SETCHL1	145.

URTMODE
for INT checking

CTE1

000056 07070707

1847
1849 SETCHL1 ROT L,4
OUT CHLINE1

146.
147.

LOC	OBJECT CODE	ADDR1	ADDR2	STMT	SOURCE STATEMENT	ASM H V 05 01.23 10/02/76
00005A	D387			1850		
				1851 *		148.
00005C	2A0C21			1852	LD HL,NXTDISA	149.
00005F	3A1F21			1854	LD A,MODEBITS	150.
000062	E610			1856	ANDI GRAPHMD	151.
000064	CA7400			1858	JMP Z,SENDISA	152.
000067	2100AF			1860	LODI HL,GRAPHODD+TOPJFF	153.
00006A	DB85			1862	INP STATBITS	154.
00006C	E608			1864	ANDI FRAMECNT	'GRAPHODD' OR 'GRAPHEVN'
00006E	CA7400			1866	JMP Z,SENDISA	DEPENDING ON FRAME COUNT BIT
000071	21AF5D			1868	LODI HL,GRAPHEVN+TOPOFF	157.
000074	7D			1870	SENDISA LOD A,L	SEND OUT THE DISPLAY ADDRESS
000075	D385			1872	OUT DISADDRL	158.
000077	7C			1874	LOD A,H	159.
000078	D384			1876	OUT DISADDRH	160.
				1878 *		161.
00007A	DB83			1879	INP SCRWHEEL	GET WHEEL CONTENTS
00007C	47			1881	CMPLOOP LOD B,A	FOR STARTUP
00007D	DB83			1883	INP SCRWHEEL	NOW GET IT FOR REAL
00007F	B8			1885	CMP B	SEE IF ITS SETTLED
000080	C27C00			1887	JMP NZ,CMPLOOP	NO. THEN CHECK AGAIN
				1889 *		166.
				1890 *		167.
				1891 *	AT THIS POINT A AND B CONTAIN THE NEW WHEEL CONTENTS	168.
000083	E6FE			1892	ANDI X'FE'	CLR LOW ORDER BIT. AND A PIZZA TO GO
000085	212F21			1894	LODI HL,PREVSCRL	POINTER TO OLD WHEEL CONTENTS
000088	46			1896	LOD B,M	GET OLD CONTENTS
000089	77			1898	LOD M,A	SAVE NEW CONTENTS
00008A	90			1900	SUB B	GET THE DIFFERNC
00008B	CACB00			1902	JMP Z,ENDWHEEL	173.
00008E	FAAE00			1904	JMP S,NEGLOOP	174.
				1906 *		175.
				1907 *	NOW WE GO THIS WAY	176.
				1908 *		177.
				1909 *	POSITIVE CASE	178.
				1910 *	IF THIS WERE PASCAL WE COULD USE A POSITIVE CASE STATEMENT.	179.
				1911 *	BUT IN CASE YOU HAVEN'T NOTICED THIS IS ASM H	180.
				1912 *		181.
				1913 *	EHF 1976	182.
				1914 *		183.
000091	D60D			1915	POSLOOP SUBI CHARHITE	DIVIDE BY CHARHITE
000093	FA9E00			1917	JMP S,POSTRATR	186.
000096	F5			1919	PUSH AF	187.
000097	CD2E04			1921	CALL SCRUP	SAVE COUNTER
00009A	F1			1923	POP AF	188.
00009B	C39100			1925	JMP POSLOOP	AND CONTINUE DIVISION
		0009E		1927	POSTRATR EQU *	HERE TO UPDATE RASTER COUNTER:
00009E	C60D			1929	ADDI CHARHITE	RESTORE AFTER DIVISION TO GET REM.
0000A0	CACB00			1931	JMP Z,ENDWHEEL	IF ZERO, NO RASTER SCROLLING
0000A3	47			1933	LOD B,A	SAVE THAT
0000A4	CD8704			1935	POSLP2 CALL SCRUPSDO	SCROLL A RASTER LINE
0000A7	05			1937	DEC B	SEE IF DONE
0000A8	C2A400			1939	JMP NZ,POSLP2	MORE RASTER LINES TO SCROLL
0000AB	C3CB00			1941	JMP ENDWHEEL	OTHERWISE CLEANUP AND RETURN
				1943 *		199.
						200.
						201.

1944 *
1945 *HERE FOR NEGATIVE CASE
1946 *
1947 *

202.
203.
204.
205.

LOC	OBJECT CODE	ADDR1	ADDR2	STMT	SOURCE	STATEMENT	ASM H V 05 01.23 10/02/76
0000AE	C60D			1948	NEGLOOP	ADDI CHARHITE	DO DIVISION IN THE MINUS WAY 206.
0000B0	CAB600			1950		JMP Z,NEGZERO	IF ZERO WE WANT TO UPDATE 207.
0000B3	F2BE00			1952		JMP NS,NEGRASTR	IF NS, THEN DIV. IS DONE 208.
0000B6	F5			1954	NEGZERO	PUSH AF	209.
0000B7	CD1604			1956		CALL SCRDOWN	210.
0000BA	F1			1958		POP AF	211.
0000BB	C3AE00			1960		JMP NEGLOOP	212.
		000BE		1962	NEGRASTR	EQU *	HERE TO UPDATE RASTERS 213.
0000BE	D60D			1964		SUBI CHARHITE	GET REMAINDER 214.
0000C0	CACB00			1966		JMP Z,ENDWHEEL	IF ZERO, NO RASTER SCROLL 215.
0000C3	47			1968		LOD B,A	SAVE IT 216.
0000C4	CD6804			1970	NEGLP2	CALL SCRDOWNSD	SCROLL DOWN A RASTER LINE 217.
0000C7	04			1972		INC B	SEE IF DONE 218.
0000C8	C2C400			1974		JMP NZ,NEGLP2	NOT DONE 219.
				1976	*		220.
				1977	*		221.
				1978	*		222.
				1979	*		223.
		000CB		1980	ENDWHEEL	EQU *	HERE TO FINISH THINGS OFF 224.
0000CB	211521			1982		LODI HL,TIME+3	UPDATE CLOCK TIME 225.
0000CE	7E			1984	TICKTOCK	LOD A,M	226.
0000CF	3C			1986		INC A	227.
0000D0	27			1988		DAA	228.
0000D1	77			1990		LOD M,A	229.
0000D2	FE60			1992		CMPI X'60'	230.
0000D4	FADD00			1994		JMP S,NOTICK	231.
0000D7	3600			1996		LODI M,0	232.
0000D9	2B			1998		DEC HL	233.
0000DA	C3CE00			2000		JMP TICKTOCK	234.
		000DD		2002	NOTICK	EQU *	235.
				2004	*		236.
0000DD	210321			2005		LODI HL,CURSCTR	TIME TO CHANGE CURSOR? 237.
0000E0	35			2007		DEC M	238.
0000E1	C2FF00			2009		JMP NZ,CHKKBRD	NO... 239.
0000E4	3608			2011		LODI M,CURSTIME	YES - RESET COUNTER 240.
0000E6	2A0021			2013		LD HL,CURSLOC	CURRENT LOCATION 241.
0000E9	7E			2015		LOD A,M	CURRENT CURSOR 242.
0000EA	FE7F			2017		CMPI @DEL	BLOB? 243.
0000EC	CAF400			2019		JMP Z,CURCH1	YES... 244.
0000EF	367F			2021		LODI M,@DEL	NO - MAKE IT A BLOB 245.
0000F1	C3FF00			2023		JMP CHKKBRD	246.
0000F4	3A0421			2025	CURCH1	LD A,CURSCHAR	CURSOR IS A BLOB, 247.
0000F7	FE7F			2027		CMPI @DEL	IS CHAR, TOO? 248.
0000F9	77			2029		LOD M,A	SUPPOSE NOT, PUT IN CHAR 249.
0000FA	C2FF00			2031		JMP NZ,CHKKBRD	CORRECT... 250.
0000FD	3620			2033		LODI M,' '	WRONG - MAKE CURSOR BLANK 251.
0000FF	DB84			2035	CHKKBRD	INP KEYBOARD	GET POSSIBLE KEYBOARD CHAR 252.
000101	B7			2037		IOR A	253.
000102	FA1502			2039		JMP S,KEYRCVED	KEY DEPRESSED... 254.
000105	3EFF			2041		LODI A,X'FF'	NOTHING - RESET KEYBRD REPEAT COUNT 255.
000107	320821			2043		ST A,KEYCTR	256.

*direct KB
if key [load curd char into A] JMP TSCMD*

LOC	OBJECT CODE	ADDR1	ADDR2	STMT	SOURCE	STATEMENT	ASM H V 05 01.23 10/02/76
00010A	320721			2045		ST A,PRVCCCHAR	RESET PREVIOUS CMD KEY 257.
00010D	3E01			2047		LODI A,1	SET SCROLL TIMING 258.
00010F	321021			2049		ST A,SCRCTR	AND RESET IT 259.
000112	2A1621			2051		LD HL,TIMELOC	MUST WE ERASE THE TIME MSG? 260.
000115	7C			2053		LOD A,H	261.
000116	B7			2055		IOR A	262.
000117	CA2701			2057		JMP Z,CHKATTN	NO... 263.
00011A	3E0B			2059		LODI A,11	264.
00011C	3620			2061	TIMERASE	LODI M,' '	265.
00011E	23			2063		INC HL	266.
00011F	3D			2065		DEC A	267.
000120	C21C01			2067		JMP NZ,TIMERASE	268.
000123	97			2069		SUB A	269.
000124	321721			2071		ST A,TIMELOC+1	270.
000127	DB85			2073	CHKATTN	INP STATBITS	271.
000129	E602			2075		ANDI KBATTN	ATTN KEY PRESSED? 272.
00012B	CABC01			2077		JMP Z,DNTBREAK	GO SEE IF SHOULD TURN OFF BRK 273.
00012E	CDA401			2079		CALL XMITRCLR	274.
000131	3E0F			2081		LODI A,URTBREAK	SEND BREAK IF SO 275.
000133	D341			2083	CTLBRK	OUT URTCTL	276.
				2085	*		277.
		00135		2086	INTEND EQU *		HERE TO CHECK FOR XMIT RDY 278.
				2088	* CHK FOR SENDING CHARACTERS		279.
000135	DB41			2089		INP URTSTAT	280.
000137	E601			2091		ANDI URTTXRDY	CAN WE XMIT 281.
000139	CA7B01			2093		JMP Z,INTEND2	NO WE CANT 282.
00013C	2A3E21			2095		LD HL,XMITHAN	YES WE CAN GET HANDLER ADD 283.
				2097	* IF IN @TE1 MODE THIS IS JUST A BRANCH TO @TE1		284.
00013F	E9			2098		JMP (HL)	285.
				2100	*		286.
				2101	* WE COME HERE AFTER THE BRANCH FROM XMIT INTERRUPT.		287.
				2102	* @TCE1 IS A GENERAL PURPOSE ROUTINE USED TO TAKE CHARACTERS OUT		288.
				2103	* OF BUFFER		289.
				2104	*		290.
		00140		2105	@TE1 EQU *		291.
000140	2A3521			2107		LD HL,XMITGET	GET PNT TO THINGS TO BE SNT 292.
000143	3A3721			2109		LD A,XMITPUT	SEE IF ANYTHING THERE 293.
000146	BD			2111		CMP L	294.
000147	C25101			2113		JMP NZ,XMITINFO	THERE IS 295.
00014A	3A3821			2115		LD A,XMITPUT+1	GET HIGH PART OF PUT 296.
00014D	BC			2117		CMP H	SEE IF EQUAL 297.
00014E	CA7B01			2119		JMP Z,INTEND2	NOTHING TO SEND SO LEAVE 298.
		00151		2121	XMITINFO EQU *		HERE TO SEND SOMETHING 299.
				2123		INC HL	MOVE POINTER UP 300.
000151	23			2125		LODI A,XMITEND,<	SEE IF WRAP AROUND 301.
000152	3E41			2127		CMP H	302.
000154	BC			2129		JMP NZ,XMITNOWP	303.
000155	C25B01			2131		LODI HL,XMITBUF	304.
000158	210040			2133	XMITNCWP EQU *		305.
		0015B		2135		ST HL,XMITGET	SAVE NEW POINTER 306.
00015B	223521			2137		LD A,(HL)	GET CHARACTER 307.
00015E	7E			2139		OUT URTXMT	SEND IT OFF TO HOST 308.
00015F	D301						

LOC	OBJECT CODE	ADDR1	ADDR2	STMT	SOURCE	STATEMENT	ASM H V 05 01.23 10/02/76
000161	FE04			2141		CMPI @EOT	SEE IF CNTL D ATTN 309.
000163	CA6B01			2143		JMP Z,SIMCR	YES THEN FAKE A <CR> FOR TYPE AHEAD 310.
				2145	*		THIS IS USEFUL WAY OF DELETING A 311.
				2146	*		LINE YOU JUST TYPED, BUT NOT CLRIN 312.
				2147	*		OUT ALL OF TYPEAHEAD BUFFER. 313.
				2148	*		314.
000166	FE0D			2149		CMPI @CR	315.
000168	C27201			2151		JMP NZ,XMITCALL	316.
00016B	213B21			2153	SIMCR	LODI HL,CRCNT	317.
00016E	35			2155		DEC M	DECREASE COUNT OF <CR> IN BUF 318.
00016F	23			2157		INC HL	319.
000170	3600			2159		LODI M,0	AND TURN OFF DC1 INDICATOR 320.
		00172		2161	XMITCALL	EQU *	321.
000172	57			2163		LOD D,A	SAVE THE CHARACTER 322.
000173	3A1921			2165		LD A,FLAGS	SEE IF IN FULLDUPLEX MODE 323.
000176	E680			2167		ANDI FULLDUPL	SEE IF IN FULLDUPLEX MODE 324.
000178	CCB002			2169		CALL Z,KEYSTOR2	PUT IN DISPLAY BUFFER 325.
				2171	*		326.
		0017B		2172	INTEND2	EQU *	HERE TO LEAVE FROM INTERUPT 327.
00017B	C1			2174		POP BC	328.
00017C	D1			2176		POP DE	329.
00017D	E1			2178		POP HL	330.
00017E	F1			2180		POP FA	331.
00017F	FB			2182		EI	332.
000180	C9			2184		RET ,	*** END OF INTERRUPT *** 333.
				2186	*		334.
				2187	*		335.
				2188	*		336.
				2189	@TOCR	LD A,CRCNT	337.
000181	3A3B21			2191		IOR A	COMPAR 338.
000184	B7			2193		JMP Z,INTEND2	NO CR-THEN NO XMIT 339.
000185	CA7B01			2195		JMP @TE1	ELSE XMIT 340.
000188	C34001			2197	@TODC1	EQU *	ROUTINE TO XMIT ON DC1 341.
00018B	3A3C21			2199		LD A,DC1CNT	342.
00018E	B7			2201		IOR A	343.
00018F	CA7B01			2203		JMP Z,INTEND2	NO DC1 NO XMIT 344.
000192	C34001			2205		JMP @TE1	ELSE GO XMIT 345.
				2208	*		347.
		00195		2209	RCVCLR	EQU *	ESET RECEIVER BUFFER 348.
000195	210041			2211		LODI HL,BUFFER	349.
000198	222021			2213		ST HL,PUTPTR	INITIALIZE BUFFER POINTERS 350.
00019B	222221			2215		ST HL,GETPTR	351.
00019E	3E01			2217		LODI A,1	352.
0001A0	321021			2219		ST A,SCRCTR	RESET SCROLL COUNTER 353.
0001A3	C9			2221		RET ,	AND GO BACK 354.
				2223	*		355.
				2224	*		356.
				2225	*		357.
				2226	*		358.
				2227	*		359.
				2228	*		360.
				2229	*		361.

0001A4 210040
0001A7 223721

001A4	2230	XMITRCLR	EQU	*
	2232		LODI	HL,XMITBUF
	2234		ST	HL,XMITPUT
	2236		ST	HL,XMITGET

362.
363.
364.
365.

LOC	OBJECT CODE	ADDR1	ADDR2	STMT	SOURCE STATEMENT	ASM H V 05 01.23 10/02/76
0001AA	223521			2237		
0001AD	223921			2238	ST HL,XMITSAV	366.
0001B0	97			2240	SUB A	367.
0001B1	323B21			2242	ST A,CRCNT	368.
0001B4	323C21			2244	ST A,DC1CNT	369.
0001B7	3D			2246	DEC A	370.
0001B8	321121			2248	ST A,BRKSTATE	371.
				2250	* NOTE THE FOLLOWING RELATED TO TYPE AHEAD. IF YOU GO INTO	372.
				2251	* TYPEAHEAD WHILE COMPUTER IS TYPEING AND THEN START TYPING	373.
				2252	* TYPEAHEAD WILL NOT WORK UNTIL FIRST CARRAIGE RETURN	374.
				2253	* HOWEVER IF YOU HIT ATTN OR BUF RESET IT TYPE AHEAD WILL STILL WORK	375.
				2254	*	376.
				2255	*	377.
0001BB	C9			2256	RET ,	378.
				2258	*	379.
				2259	*	380.
				2260	*	381.
		001BC		2261	DNTBREAK EQU *	382.
0001BC	211121			2263	LODI HL,BRKSTATE	383.
0001BF	34			2265	INC M	384.
0001C0	3600			2267	LODI M,0	385.
0001C2	C23501			2269	JMP NZ,INTEND	386.
0001C5	3E07			2271	LODI A,URTRSBRK	387.
0001C7	D341			2273	OUT URTCTL	388.
0001C9	C33501			2275	JMP INTEND	389.
				2277	*	390.
				2278	*	391.
				2279	USART INTERRUPT	392.
0001CC	D383			2280	INTURT OUT RSURTINT	393.
0001CE	DB41			2282	INP URTSTAT	394.
0001D0	6F			2284	LOD L,A	395.
0001D1	E602			2286	ANDI URTRXRDY	396.
0001D3	CA3501			2288	JMP Z,INTEND	397.
0001D6	7D			2290	LOD A,L	398.
0001D7	E638			2292	ANDI URTRERR	399.
0001D9	CAE401			2294	JMP Z,RCVROK	400.
0001DC	D386			2296	OUT BELL	401.
0001DE	3E17			2298	LODI A,URTRSERR	402.
0001E0	D341			2300	OUT URTCTL	403.
0001E2	3E7F			2302	LODI A,@DEL	404.
		001E4		2304	RCVROK EQU *	405.
0001E4	3A2421			2306	LD A,LCLMODE	406.
0001E7	E604			2308	ANDI NORCV	407.
0001E9	DB01			2310	INP URTRCV	408.
0001EB	C23501			2312	JMP NZ,INTEND	409.
0001EE	FE11			2314	CMPI @DC1	410.
0001F0	C20802			2316	JMP NZ,NOTDC1	411.
0001F3	3A3C21			2318	LD A,DC1CNT	412.
0001F6	B7			2320	IOR A	413.
0001F7	CA0002			2322	JMP Z,SETDC1	414.
0001FA	2A3921			2324	LD HL,XMITSAV	415.
0001FD	223521			2326	ST HL,XMITGET	416.
000200	3E11			2328	SETDC1 LODI A,@DC1	417.
000202	323C21			2330	ST A,DC1CNT	418.
000205	2A3521			2332	LD HL,XMITGET	419.

LOC	OBJECT CODE	ADDR1	ADDR2	STMT	SOURCE	STATEMENT	ASM H V 05 01.23 10/02/76
000208	223921			2334		ST HL,XMITSAV	AND SAVE FOR REPROMPTING 420.
00020B	B7			2336	NOTDC1	IOR A	421.
00020C	CA3501			2338		JMP Z,INTEND	IGNORE NULL 422.
00020F	CDB102			2340		CALL KEYSTORE	GO SAVE THE CHARACTER IN BUFFER 423.
000212	C33501			2342		JMP INTEND	424.
		00215		2344	KEYRCVED	EQU *	425.
000215	E67F			2346		ANDI X'7F'	REMOVE STROBE BIT 426.
000217	6F			2348		LOD L,A	427.
000218	DB85			2350		INP STATBITS	428.
00021A	E601			2352		ANDI KBRPT	SEE IF REPEAT PRESS 429.
00021C	0F			2354		ROT R	PUT IT IN RIGHT PLACE 430.
00021D	B5			2356		IOR L	431.
00021E	FA2802			2358		JMP S,KEYCMD	432.
000221	57			2360		LOD D,A	SAVE IT 433.
000222	CD3002			2362		CALL KEYRPTL	REG. CHAR. DO REPEAT 434.
000225	C33501			2364		JMP INTEND	AND GO HOME 435.
		00228		2366	KEYCMD	EQU *	ITS A COMMAND 436.
000228	CD5302			2368		CALL CMDDOIT	GO PROCESS <CMD> FUNCTION 437.
00022B	D396			2370		OUT KBRESET	RESET 'NEWKB STROBE' BIT 438.
00022D	C33501			2372		JMP INTEND	THEN GO HOME 439.
				2374	*		440.
				2375	*	AUTOMATIC KEY REPEAT LOGIC	441.
				2376	*		442.
000230	210821			2377	KEYRPTL	LODI HL,KEYCTR	LOOK AT REPEAT COUNTER 443.
000233	DB85			2379		INP STATBITS	GET THE 'NEW KB STROBE' STATUS BIT 444.
000235	E610			2381		ANDI KBNEWCHR	WHICH INDICATES A NEW CHARACTER 445.
000237	C24502			2383		JMP NZ,KEYNEWCH	YES, INDEED... 446.
00023A	34			2385		INC M	OLD CHAR - INCREMENT REPEAT WAIT CTR 447.
00023B	7E			2387		LOD A,M	448.
00023C	FE14			2389		CMPI KEYRPTD	TIME FOR REPEAT? 449.
00023E	D8			2391		RET C	NO - DO NOTHING 450.
00023F	D603			2393		SUBI KEYRPTR	YES - DECR. FOR NEXT REPEAT 451.
000241	77			2395		LOD M,A	452.
000242	C36502			2397		JMP KEYDOIT2	AND GO DOIT 453.
000245	D396			2399	KEYNEWCH	OUT KBRESET	RESET 'NEW KB STROBE' STATUS BIT 454.
000247	36FF			2401		LODI M,X'FF'	INITIALIZE REPEAT WAIT COUNTER 455.
000249	C36502			2403		JMP KEYDOIT2	AND PROCESS FIRST OCCURENCE OF CHAR 456.
				2405	*		457.
				2406	*		458.

*CMP R1 @DEL
JMP Z,INTEND*

TSICMD IOR A test scpm bit



LOC	OBJECT CODE	ADDR1	ADDR2	STMT	SOURCE STATEMENT	ASM H V 05 01.23 10/02/76
				2408	*****	460.
				2409	*	461.
				2410	* CHARPROC: CHARACTER PROCESSING ROUTINE	462.
				2411	*	463.
				2412	* THE CHARACTER IN 'A' IS EITHER (1) PROCESSED AS A <COMMAND>	464.
				2413	* IF THE HIGH-ORDER BIT IS ON, OR (2) PUT IN THE	465.
				2414	* 'RECEIVER BUFFER' FOR LATER BACKGROUND PROCESSING.	466.
				2415	*	467.
				2416	* CALLS FROM THE BACKGROUND SHOULD PROBABLY DISABLE INTERRUPTS.	468.
				2417	*	469.
				2418	* DESTROYS: A, BC, DE, HL	470.
				2419	*	471.
				2420	*****	472.
00024C	57		0024C	2421	CHARPROC EQU * HERE TO PROCESS CHARACTERS	473.
				2423	LCD D,A SAVE THE CHAR	474.
				2425	*	475.
				2426	*	476.
				2427	*	477.
				2428	*	478.
				2429	* CHECK FOR SPECIAL ESCAPE KEYS	479.
				2430	*	480.
00024D	E680			2431	ANDI CMDKEY 'CMD' KEY?	481.
00024F	CA6702			2433	JMP Z,KEYDOIT NO, NOT SPECIAL...	482.
000252	7A			2435	LCD A,D	483.
000253	E67F			2437	ANDI X'7F' GET RID OF CMD BIT	484.
000255	21E602			2439	LODI HL,CMDTAB	485.
000258	CD3E06			2441	CALL SEARCH SEARCH FOR CHAR IN TABLE	486.
00025B	D8			2443	RET C	487.
00025C	3A0721			2445	LD A,PRVCCHAR	488.
00025F	5F			2447	LCD E,A SAVE LAST CMD CHAR	489.
000260	7A			2449	LCD A,D LOAD BACK CHAR	490.
000261	320721			2451	ST A,PRVCCHAR SAVE NEW ONE AS OLD	491.
000264	E9			2453	JMP (HL) AND GO TO ROUTINE	492.

LOC	OBJECT CODE	ADDR1	ADDR2	STMT	SOURCE STATEMENT	ASM H V 05 01.23 10/02/76
				2456	*****	494.
				2457	*	495.
				2458	* KEYDOIT: PUT CHARACTER IN TRANSMIT OR RECEIVE BUFFER	496.
				2459	*	497.
				2460	* THE CHARACTER IN 'D' IS STORED IN TRANSMIT BUFFER (IF NOT	498.
				2461	* LOCAL OR SEMI-LOCAL MODE)	499.
				2462	* OTHERWISE, IT PASSES CONTROL TO 'KEYDOIT' (Q.V.)	500.
				2463	*	501.
				2464	* DESTROYS: A, HL	502.
				2465	*	503.
				2466	*****	504.
000265	D3AE			2467	KEYDOIT2 OUT KBCLICK SAME AS KEYDOIT, BUT CLICKS	505.
000267	7A			2469	KEYDOIT LOD A,D THE CHAR	506.
000268	320921			2471	ST A,KEYSAVED SAVE IT FOR NEXT TIME	507.
00026B	B7			2473	IOR A SEE IF SPECIAL CURS MOTION	508.
00026C	FAB102			2475	JMP S,KEYSTORE IF SIGNED IT IS SO DON'T SEND	509.
00026F	3A2421			2477	LD A,LCLMODE. SEE IF JUST LOCAL TXT ENTRY	510.
000272	E602			2479	ANDI NOSEND	511.
000274	C2B002			2481	JMP NZ,KEYSTOR2 IT IS.GO AROUND XMIT STUFF	512.
				2483	*	513.
				2484	* STUFXMIT: STORE (D) IN TRANSMIT BUFFER	514.
				2485	*	515.
000277	2A3721			2486	STUFXMIT LD HL,XMITPUT PLACE TO STORE CHAR IN BUF	516.
00027A	23			2488	INC HL BUMP IT	517.
00027B	97			2490	SUB A CLEAR A	518.
00027C	323321			2492	ST A,XMITSTAT SO WE CAN CLEAR XMITSTAT	519.
00027F	3E41			2494	LODI A,XMITEND,< CHECK FOR WRAP AROUND	520.
000281	BC			2496	CMP H	521.
000282	C28802			2498	JMP NZ,XMITCHK	522.
000285	210040			2500	LODI HL,XMITBUF WE WRAPED	523.
		00288		2502	XMITCHK EQU *	524.
000288	223721			2504	ST HL,XMITPUT AND SAVE FINAL ADDRESS	525.
00028B	3A3521			2506	LD A,XMITGET CHKECK FOR OVERRUN	526.
00028E	BD			2508	CMP L	527.
00028F	C2A102			2510	JMP NZ,XMITROOM NOT YET	528.
000292	3A3621			2512	LD A,XMITGET+1 CHECK HIGHT ORDER OART	529.
000295	BC			2514	CMP H	530.
000296	C2A102			2516	JMP NZ,XMITROOM OKAY, WE CAN STORE	531.
000299	D386			2518	OUT BELL TELL USER ABOUT OVERRUN	532.
00029B	3E01			2520	LODI A,1 PUT 1 IN A	533.
00029D	323321			2522	ST A,XMITSTAT AND STORE IN XMITSTAT	534.
0002A0	C9			2524	RET , AND LEAVE	535.
		002A1		2526	XMITROOM EQU *	536.
0002A1	7A			2528	LD A,D GET IT	537.
0002A2	77			2530	ST A,(HL) AND PUT IT IN BUFFER	538.
0002A3	FE04			2532	CMPI @ECT SEE IF CNTL-D	539.
0002A5	CAAB02			2534	JMP Z,SETSIMCR IF SO TREAT AS THOUGH <CR>	540.
				2536	*	541.
0002A8	FE0D			2537	CMPI @CR SEE COMMENTS IN TYPEAHEAD FOR REASON	542.
				2539	* NOTE THIS CHECK MUST STAY HERE, BECAUSE WE DON'T	543.
				2540	* WANT TO SCREW UP CARRIAGE RET COUNT IF BUFFER OVERFLOW	544.
0002AA	C0			2541	RET NZ NO,THEN LEVAE	545.
0002AB	213B21			2543	SETSIMCR LODI HL,CRCNT	546.
0002AE	34			2545	INC M UPDATE CRCNR	547.
0002AF	C9			2547	RET , AND GO CHK ABOUT DOING OUTPUT	548.

LCC	OBJECT CODE	ADDR1	ADDR2	STMT	SOURCE STATEMENT	ASM H V 05 01.23 10/02/76
				2550	*****	550.
				2551	*	551.
				2552	*	552.
				2553	KEYSTORE: PUT 'A' IN THE RECEIVER BUFFER.	553.
				2554	(EXCEPT IN CONTROL-UNDERSCORE (NON-ECHO) MODE)	554.
				2555	*	555.
				2556	DESTROYS: A, D	556.
				2557	*****	557.
0002B0	7A			2558	KEYSTOR2 LCD A,D	558.
				2560	*	559.
		002B1		2561	KEYSTORE EQU *	560.
0002B1	57			2563	LDD D,A	561.
0002B2	FE0D			2565	CMPI @CR	562.
0002B4	C2C102			2567	JMP NZ,NOTCRCHG	563.
0002B7	3A1921			2569	LD A,FLAGS	564.
0002BA	213121			2571	LODI HL,CNTLOMD	565.
				2573	*	566.
				2574	*	567.
				2575	*	568.
				2576	*	569.
0002BD	A6			2577	AND M	570.
0002BE	321921			2579	ST A,FLAGS	571.
0002C1	2A2021			2581	NOTCRCHG LD HL,PUTPTR	572.
0002C4	23			2583	INC HL	573.
0002C5	7C			2585	LCD A,H	574.
0002C6	FE44			2587	CMPI BUFEND,<	575.
0002C8	C2CE02			2589	JMP NZ,KEYST1	576.
0002CB	210041			2591	LODI HL,BUFFER	577.
0002CE	3A2221			2593	KEYST1 LD A,GETPTR	578.
0002D1	BD			2595	CMP L	579.
0002D2	C2DF02			2597	JMP NZ,KEYST2	580.
0002D5	3A2321			2599	LD A,GETPTR+1	581.
0002D8	BC			2601	CMP H	582.
0002D9	C2DF02			2603	JMP NZ,KEYST2	583.
0002DC	D386			2605	OUT BELL	584.
0002DE	C9			2607	RET ,	585.
0002DF	222021			2609	KEYST2 ST HL,PUTPTR	586.
0002E2	72			2611	LDD M,D	587.
0002E3	C9			2613	KEYDEF RET ,	588.
0002E4	2A2D21			2615	KEYEND LD HL,FOOADDR	589.
0002E7	E9			2617	JMP (HL)	590.

LOC	OBJECT CODE	ADDR1	ADDR2	STMT	SOURCE	STATEMENT	ASM H V 05 01.23 10/02/76
				2620	*****		592.
				2621	*****	LOCAL COMMAND TABLE	593.
				2622	*****		594.
		002E 6		2623	CMDTAB	CMDEF 20,LOCAL SPACE	595.
0002E8	20039C			2625			
0002EB	3A053D			2626	CMDEF 3A,PGDOWN	:-SCROLL PAGES DOWN	596.
0002EE	3B0528			2628	CMDEF 3B,PGUP	PAGE UP (;)	597.
0002F1	400424			2630	CMDEF 40,SCRUPQ	@ SCROLL UP	598.
0002F4	5B0512			2632	CMDEF 5B,UNHOME	LEFT BRACKET RETURN	599.
0002F7	5C040C			2634	CMDEF 5C,SCRDOWNQ	REV SLASH -SCROLL DWN QUICK	600.
0002FA	5D04CF			2636	CMDEF 5D,HOME	RIGHT BRACKET HOME DISPLAY TO CURSOR	601.
0002FD	6303A6			2638	CMDEF 63,CLRSCREEN	C-CLEAR SCREEN	602.
000300	7404A5			2640	CMDEF 74,SHOWTIME	T SHOWTIME IN UPPER RIGHT	603.
000303	08039C			2642	CMDEF 08,LOCAL	BS	604.
000306	09039C			2644	CMDEF 09,LOCAL	HT	605.
000309	0A039C			2646	CMDEF 0A,LOCAL	LF	606.
00030C	0B039C			2648	CMDEF 0B,LOCAL	VT	607.
00030F	0C03E0			2650	CMDEF 0C,ENTGRPC	FORMFEED ENTER GRAPH AND CLEAR	608.
000312	0D039C			2652	CMDEF 0D,LOCAL	CR	609.
000315	0E05AA			2654	CMDEF 0E,COPY	SD	610.
000318	0F05AA			2656	CMDEF 0F,COPY	SI	611.
00031B	1105AA			2658	CMDEF 11,COPY	OUTPUT STATUS	612.
00031E	1705B1			2660	CMDEF 17,PATCHST	COMMENCE PATCHING	613.
000321	1305AA			2662	CMDEF 13,COPY	CNTL S, CHANGE CHAR SET	614.
000324	210594			2664	CMDEF 21,STPRV	,CHANGE ESCAPE CHARACTER	615.
000327	310E7A			2666	CMDEF 31,SETTAB		616.
00032A	320E80			2668	CMDEF 32,CLRSTAB	2,CLEAR SINGLE TAB	617.
00032D	330E88			2670	CMDEF 33,CLRATAB	3,CLEAR ALLTABS AND ENABLE TABS	618.
000330	340E49			2672	CMDEF 34,RETTAB	4,TURN OFF TABS,SIMULATE 4013 TAB	619.
000333	3F038A			2674	CMDEF 3F,HELPCHK	?-GIVE USER HELP	620.
000336	4203A2			2676	CMDEF 42,STORSIZE	BIG GRAPHS	621.
000339	43057A			2678	CMDEF 43,S@TEL	SHIFT C-XMIT ON XR	622.
00033C	450F5B			2680	CMDEF 45,BRGEXT	SHIFT E = SET FOR EXTERNAL CLOCK	623.
00033F	460400			2682	CMDEF 46,SETFULL	F-FULLDUPLES(REMOTE ECHO)	624.
000342	47039C			2684	CMDEF 47,LOCAL	SPECIAL EXECUTION MODE	625.
000345	4803F8			2686	CMDEF 48,SETHALF	H-HALFDUPLEX	626.
000348	490F50			2688	CMDEF 49,BRGINT	SHIFT I = SET FOR INTERNAL CLOCK	627.
00034B	44056A			2690	CMDEF 44,S@TOCR	<CMD><SHIFT>D LINEMODE	628.
00034E	4D03A2			2692	CMDEF 4D,STORSIZE	<CMD><SHIFT>M NORMAL	629.
000351	4E0F47			2694	CMDEF 4E,NOGMODE	SHIFT N = DON'T ALLOW ENTER GMODE	630.
000354	510572			2696	CMDEF 51,S@TODC1	CMD Q TYPE AHEAD	631.
000357	72058D			2698	CMDEF 72,CLRBUFS	R, CLEAR XMIT,RCV BUFS	632.
00035A	5303A2			2700	CMDEF 53,STORSIZE	<CMD>SHIFT S SMALL GRAPH	633.
00035D	550594			2702	CMDEF 55,STPRV	SHIFT U = SET UART SPEED(CHGUART)	634.
000360	56059D			2704	CMDEF 56,VIDEOINV	SHIFT V = INVERT VIDEO	635.
000363	570F66			2706	CMDEF 57,SETURTX1	SHIFT W = X 1 CLOCK FOR UART	636.
000366	580F6F			2708	CMDEF 58,SETURTX6	SHIFT X = X 16 CLOCK FOR UART	637.
000369	590F4B			2710	CMDEF 59,YESGMODE	SHIFT Y = ALLOW ENTRY INTO GMODE	638.
00036C	6703EC			2712	CMDEF 67,ENTERGRP	G-ENTER GRAPHMODE	639.
00036F	6805DA			2714	CMDEF 68,HEXINIT	H-PROCESS HEX ADDR	640.
000372	6B03C6			2716	CMDEF 6B,LEAVGRP	K CLEAR SCREEN AND LEAVE GMODE	641.
000375	4C0552			2718	CMDEF 4C,SETLCL	SHIFT L-LOCAL MODE(NO XMIT TO HOST)	642.
000378	6D0519			2720	CMDEF 6D,MOVECURS	M MOVE CURSOR TO BOTTOM OF PAGE	643.
00037B	6E03D9			2722	CMDEF 6E,NORMODE	N LEAVE GRAPH MODE	644.
00037E	6F047D			2724	CMDEF 6F,SCRUPS	O SCROLL UP SLOWLY	645.

000381 70045E
000384 52055B
000387 0002E4

2726
2728
2730
2732 *

CMDEF 70,SCRDWNS
CMDEF 52,SETRMT
CMDEF 00,KEYEND

P SCROLL DOWN SLOWLY
SHIFT R-REMOTE MODE(XMIT TO HOST)
THAT'S ALL FOLKS

646.
647.
648.
649.
650.
651.

2733 * END OF COMMAND DEFINITIONS. NOTE THE ABOVE
2734 * TABLE MUST END WITH A "CMDEF 00,FOO" ELSE SEARCH WON'T WORK.

LOC	OBJECT CODE	ADDR1	ADDR2	STMT	SOURCE STATEMENT	ASM H V 05 01.23 10/02/76
				2735 *		652.
				2736 *		653.
				2737 *		654.
				2738 *	INTERRUPT-TIME COMMAND ROUTINES ARE ENTERED WITH THE PREVIOUS	655.
				2739 *	COMMAND CHARACTER IN 'E'. BY DOING A 'CMP E' AND 'RET Z',	656.
				2740 *	COMMANDS WILL ONLY BE EXECUTED THE FIRST TIME THE KEY	657.
				2741 *	IS PRESSED.	658.
				2742 *		659.
				2743 *		660.
				2744 *	LOCAL MODE WITH LAST COMMAND CHECK	661.
				2745 *		662.
		0038A		2746	HELPCHK EQU *	663.
00038A	BB			2748	CMP E	664.
00038B	C8			2750	RET Z	665.
00038C	3A1F21			2752	LD A,MODEBITS	666.
00038F	E610			2754	ANDI GRAPHMD	667.
000391	CA9703			2756	JMP Z,NOCLR	668.
000394	CDBC03			2758	CALL CLRGRPH	669.
000397	3EBF			2760	NOCLR LODI A,X'3F'+X'80'	670.
000399	C3B102			2762	JMP KEYSTORE	671.
				2764 *		672.
				2765 *	LOCAL CURSOR MOTION	673.
				2766 *		674.
00039C	F680			2767	LOCAL IORI X'80'	675.
				2769 *	HIGHORDER BIT INDICATES NO TRANSMIT TO HOST	676.
				2770 *		677.
00039E	57			2771	LOD D,A	678.
00039F	C33002			2773	JMP KEYRPTL	679.
				2775 *		680.
				2776 *	LOCAL CHANGE OF GRAPHIC SCALLING	681.
				2777 *		682.
0003A2	326021			2778	STORSIZE ST A,TYPE	683.
0003A5	C9			2780	RET ,	684.
				2782 *		685.
				2783 *		686.
				2784 *		687.
		003A6		2785	CLRSCREEN EQU *	688.
				2787 *		689.
0003A6	BB			2788	CMP E	690.
0003A7	C8			2790	RET Z	691.
0003A8	3A1F21			2792	LD A,MODEBITS	692.
0003AB	E610			2794	ANDI GRAPHMD	693.
0003AD	C2BC03			2796	JMP NZ,CLRGRPH	694.
0003B0	2A0C21			2798	LD HL,NXTDISA	695.
0003B3	220021			2800	ST HL,CURSLOC	696.
0003B6	3EA0			2802	LODI A,X'A0'	697.
0003B8	321821			2804	ST A,CLRFLG	698.
0003BB	C9			2806	RET ,	699.
				2808 *		700.
				2809 *		701.
		003BC		2810	CLRGRPH EQU *	702.
0003BC	3EC0			2812	LODI A,X'C0'	703.
0003BE	321821			2814	ST A,CLRFLG	704.
0003C1	3E8C			2816	LODI A,@FF+X'80'	705.
0003C3	C3B102			2818	JMP KEYSTORE	706.

	2820	*	THE ABOVE IS A LITTLE KLUGGY IN THAT FF MAY BE OUT		707.	
	2821	*	OF SYNC WITH REST OF TEXT.		708.	
	2822	*	EHF		709.	
	2823	*			710.	
	2824	*			711.	
	2825	*			712.	
0003C6	BB	003C6	2826 LEAVGRP	EQU *	CMD K- LEAVES GRPHMODE AND CLEAR	713.
0003C7	C8		2828	CMP E		714.
0003C8	3E80		2830	RET Z		715.
0003CA	321821		2832	LODI A,X'80'	FLAG TO CLEAR	716.
			2834	ST A,CLRFLG	NO MATTER WHAT	717.

LOC	OBJECT CODE	ADDR1	ADDR2	STMT	SOURCE	STATEMENT	ASM H V 05 01.23 10/02/76
0003CD	2A6721			2836	LD	HL,TEXT	718.
0003D0	220021			2838	ST	HL,CURSLOC	719.
0003D3	3A1F21			2840	LD	A,MODEBITS	720.
0003D6	E610			2842	ANDI	GRAPHMD	721.
0003D8	C8			2844	RET	Z	722.
				2846	*		723.
			003D9	2847	NORMODE	EQU *	724.
0003D9	BB			2849	CMP	E	725.
0003DA	C8			2851	RET	Z	726.
0003DB	3EFE			2853	LODI	A,X'FE'	727.
0003DD	C3B102			2855	JMP	KEYSTORE	728.
				2857	*		729.
				2858	*		730.
				2859	*		731.
			003E0	2860	ENTGRPC	EQU *	732.
0003E0	BB			2862	CMP	E	733.
0003E1	C8			2864	RET	Z	734.
				2866	*		735.
				2867	*		736.
				2868	LD	A,ALLOWGMD	737.
0003E2	3A3221			2870	IOR	A	738.
0003E5	B7			2872	RET	NZ	739.
0003E6	C0			2874	LODI	A,X'C0'	740.
0003E7	3EC0			2876	ST	A,CLRFLG	741.
0003E9	321821			2878	*		742.
				2879	*		743.
				2880	*		744.
			003EC	2881	ENTERGRP	EQU *	745.
0003EC	BB			2883	CMP	E	746.
0003ED	C8			2885	RET	Z	747.
0003EE	3A3221			2887	LD	A,ALLOWGMD	748.
0003F1	B7			2889	IOR	A	749.
0003F2	C0			2891	RET	NZ	750.
0003F3	3E8C			2893	LODI	A,@FF+X'80'	751.
0003F5	C3B102			2895	JMP	KEYSTORE	752.
				2897	*		753.
				2898	*		754.
				2900	*		756.
				2901	*	SET FULL/HALF DUPLEX	757.
				2902	*		758.
0003F8	3A1921			2903	SETHALF	LD A,FLAGS	759.
0003FB	E67F			2905	ANDI	255-FULLDUPL	760.
0003FD	C30504			2907	JMP	SETFULL2	761.
000400	3A1921			2909	SETFULL	LD A,FLAGS	762.
000403	F680			2911	IORI	FULLDUPL	763.
000405	321921			2913	SETFULL2	ST A,FLAGS	764.
000408	323121			2915	ST	A,CNTLOMD	765.
00040B	C9			2917	RET	,	766.
						SAVE COPY FOR CNTLSHIFT O FAKE	

LOC	OBJECT CODE	ADDR1	ADDR2	STMT	SOURCE STATEMENT	ASM H V 05 01.23 10/02/76
				2920 *		768.
				2921 *	SCROLL DOWN QUICKLY	769.
				2922 *		770.
00040C	211021			2923	SCRDWNQ LODI HL,SCRCTR	771.
00040F	35			2925	DEC M	772.
000410	C0			2927	RET NZ	773.
000411	3604			2929	LODI M,SCRQTIME	774.
000413	CD1604			2931	DSALFINE	775.
				2933 *		776.
000416	CD3C04			2934	SCRDWN CALL DOWNCHK	777.
000419	D0			2936	RET NC	778.
00041A	2A0C21			2938	LD HL,NXTDISA	779.
00041D	CD5B07			2940	CALL SUBWRP81	780.
000420	220C21			2942	ST HL,NXTDISA	781.
000423	C9			2944	RET ,	782.
				2946 *		783.
				2947 *	SCROLL UP QUICKLY	784.
				2948 *		785.
000424	211021			2949	SCRUPQ LODI HL,SCRCTR	786.
000427	35			2951	DEC M	787.
000428	C0			2953	RET NZ	788.
000429	3604			2955	LODI M,SCRQTIME	789.
00042B	CD2E04			2957	DSALFINE	790.
				2959 *		791.
00042E	CD4A04			2960	SCRUP CALL UPCHK	792.
000431	D0			2962	RET NC	793.
000432	2A0C21			2964	SCRUPDO LD HL,NXTDISA	794.
000435	CD5007			2966	CALL ADDWRP81	795.
000438	220C21			2968	ST HL,NXTDISA	796.
00043B	C9			2970	RET ,	797.

LOC	OBJECT CODE	ADDR1	ADDR2	STMT	SOURCE STATEMENT	ASM H V 05 01.23 10/02/76
00043C	2A0C21			2973	DOWNCHK LD HL,NXTDISA	CHECKS TO SEE IF WE CAN SCROLL DOWN 799.
00043F	545D			2975	LOD DE,HL	REMEMBER NXTDISA 800.
000441	2A6B21			2977	LD HL,TEXTTOP	GET UPPER BOUNDARY 801.
000444	19			2979	ADD HL,DE	FIND DIFFERENCE 802.
000445	11FFFF			2981	LODI DE,X'FFFF'	803.
000448	19			2983	ADD HL,DE	804.
000449	C9			2985	RET ,	RETURN WITH CARRY RESET IF IT IS 0 805.
00044A	2A0C21			2987	UPCHK LD HL,NXTDISA	GET DISPLAY ADDRESS 806.
00044D	11B50B			2989	LODI DE,SCRNSIZE	ADD ON SIZE OF SCREEN 807.
000450	CD5307			2991	CALL ADDWRP	808.
000453	545D			2993	LOD DE,HL	SAVE NEW NXTDISA 809.
000455	2A6D21			2995	LD HL,TEXTBOT	SEE IF ITS TOO FAR 810.
000458	19			2997	ADD HL,DE	811.
000459	11FFFF			2999	LODI DE,X'FFFF'	812.
00045C	19			3001	ADD HL,DE	813.
00045D	C9			3003	RET ,	RETURN WITH CARRY RESET IF SO 814.
				3005	*	815.
				3006	*	816.
				3007	*	817.
				3008	SCRDWNS LODI HL,SCRCTR	818.
000461	35			3010	DEC M	TIME TO ACT? 819.
000462	C0			3012	RET NZ	NO... 820.
000463	3606			3014	LODI M,SCRSTIME	YES - RESET FOR NEXT TIME 821.
000465	CD6804			3016	DSALFINE	SCROLL TWO RASTER LINES 822.
000468	210A21			3018	SCRDWNSD LODI HL,NXTCHL1	POINT TO PTR TO CHL1TAB ENTRY 823.
00046B	35			3020	DEC M	DECREMENT LOW BYTE OF PTR 824.
00046C	7E			3022	LOD A,M	(TABLE IS ALL ON 1 PAGE) 825.
00046D	FE98			3024	CMPI CHL1TAB,>	UP BEFORE START OF TABLE? 826.
00046F	D0			3026	RET NC	NO - LEAVE IT 827.
000470	36A4			3028	LODI M,CHL1TAB+12,>	YES - MOVE TO BOTTOM OF TABLE 828.
000472	E5			3030	PUSH HL	SAV NXTCHL1 829.
000473	CD3C04			3032	CALL DOWNCHK	SHOULD WE SCROLL A FULL LINE? 830.
000476	E1			3034	POP HL	831.
000477	DA1604			3036	JMP C,SCRDOWN	IT'S OK, GO DO IT 832.
00047A	3698			3038	LODI M,CHL1TAB,>	NO SO MOVE POINTER BACK... 833.
00047C	C9			3040	RET ,	834.
				3042	*	835.
				3043	*	836.
				3044	*	837.
				3045	SCRUPS LODI HL,SCRCTR	838.
000480	35			3047	DEC M	839.
000481	C0			3049	RET NZ	840.
000482	3606			3051	LODI M,SCRSTIME	841.
000484	CD8704			3053	DSALFINE	SCROLL TWO RASTER LINES 842.
000487	CD4A04			3055	SCRUPS DO CALL UPCHK	DO WE MOVE AT ALL? 843.
00048A	D0			3057	RET NC	NO, JUST RETURN 844.
00048B	210A21			3059	LODI HL,NXTCHL1	POINT TO PTR TO CHL1TAB ENTRY 845.
00048E	34			3061	INC M	INCREMENT LOW BYTE OF PTR 846.
00048F	7E			3063	LOD A,M	847.
000490	FEA5			3065	CMPI CHL1TAB+13,>	PAST THE END? 848.
000492	D8			3067	RET C	NO - LEAVE IT THERE 849.
000493	3698			3069	LODI M,CHL1TAB,>	YES - POINT TO TOP OF TABLE 850.

LOC	OBJECT CODE	ADDR1	ADDR2	STMT	SOURCE STATEMENT	ASM H V 05 01.23 10/02/76
000495	C32E04			3071	JMP SCRUP	851.
				3073 *		852.
				3074 *	THE FOLLOWING TABLE GIVES THE VALUES WHICH MUST BE OUTPUT	853.
				3075 *	TO THE CHARACTER LINE COUNTER ('OUT CHLINE1') TO DETERMINE	854.
				3076 *	WHICH LINE OF THE FIRST CHARACTER ROW IS THE FIRST TO BE	855.
				3077 *	DISPLAYED AT THE TOP OF THE SCREEN.	856.
				3078 *	THERE ARE 13 ENTRIES, ONE FOR EACH POSSIBLE CHARACTER LINE,	857.
				3079 *	AND THE FIRST IS THE 'NORMAL' ONE, IE THE ONE WHICH DISPLAYS	858.
				3080 *	A FULL ROW AT THE TOP SUCH THAT THE BOTTOM ROW OF 13 LINES	859.
				3081 *	IS FOLLOWED BY THE HALF RASTER FROM THE TOP LINE OF THE	860.
				3082 *	NEXT CHARACTER ROW. THAT IS DONE BY DISPLAYING 14 LINES	861.
				3083 *	FROM THE FIRST ROW, SINCE $1+37*13+.5=482.5$ DISPLAYED LINES.	862.
				3084 *	THE FIRST 4 BITS OF EACH ENTRY IS THE NUMBER TO BE OUTPUT	863.
				3085 *	FOR EVEN (FRAMECNT=0) FRAMES, AND THE SECOND 4 BITS IS	864.
				3086 *	FOR THE ODD (FRAMECNT=1) FRAMES.	865.
				3087 *	YOU CAN UNDERSTAND THESE NUMBERS ONLY BY LOOKING AT THE SMALL	866.
				3088 *	PORTION OF THE SCHEMATIC WHICH SHOWS THE CHARACTER LINE COUNTER	867.
				3089 *	AND CPU LOAD MULTIPLEXER FOR IT. REMEMBER THAT 74158'S INVERT	868.
				3090 *	GOOD LUCK. YOU SHOULD WORK AS HARD FOR THIS AS I DID. (LJS)	869.
				3091 *		870.
000498	8009911AA22BB3			3092	CHL1TAB DC X'8009911AA22BB3'	871.
00049F	3CC44DD55EE6			3093	DC X'3CC44DD55EE6'	872.

LOC	OBJECT CODE	ADDR1	ADDR2	STMT	SOURCE STATEMENT	ASM H V 05 01.23 10/02/76
				3095 *		874.
				3096 *	DISPLAY CLOCK TIME IN UPPER RIGHT CORNER	875.
				3097 *		876.
0004A5	2A0C21			3098 SHOWTIME	LD HL,NXTDISA	877.
0004A8	119100			3100	LODI DE,80+65	878.
0004AB	19			3102	ADD HL,DE	879.
0004AC	221621			3104	ST HL,TIMELOC	880.
0004AF	111221			3106	LODI DE,TIME	881.
0004B2	1A			3108 SHOWTIM1	LD A,(DE)	882.
0004B3	0F0F0F0F			3110	ROT R,4	HIGH-ORDER DECIMAL DIGIT
0004B7	E60F			3112	ANDI X'0F'	884.
0004B9	C630			3114	ADDI '0'	885.
0004BB	77			3116	ST A,(HL)	886.
0004BC	23			3118	INC HL	887.
0004BD	1A			3120	LD A,(DE)	888.
0004BE	E60F			3122	ANDI X'0F'	LOW-ORDER DECIMAL DIGIT
0004C0	C630			3124	ADDI '0'	889.
0004C2	77			3126	ST A,(HL)	890.
0004C3	23			3128	INC HL	891.
0004C4	13			3130	INC DE	NEXT PAIR
0004C5	7B			3132	LOD A,E	893.
0004C6	FE16			3134	CMPI TIME+4,>	DONE?
0004C8	C8			3136	RET Z	YES...
0004C9	363A			3138	LODI M,'::'	NO - INSERT '::'
0004CB	23			3140	INC HL	898.
0004CC	C3B204			3142	JMP SHOWTIM1	899.
				3145 *		901.
				3146 *	HOME SCREEN TO PUT CURSOR AT BOTTOM	902.
				3147 *		903.
0004CF	BB	004CF		3148 HOME	EQU *	904.
0004D0	C8			3150	CMP E	WAS LAST CMD HOME?
0004D1	2A0C21			3152	RET Z	YES, IGNORE THIS ONE
0004D4	E5			3154	LD HL,NXTDISA	SAVE CURRENT DISPLAY ADDRESS
0004D5	2A0021			3156	PUSH HL	BY PUSHING IT
0004D8	119DF4			3158	LD HL,CURSLOC	CURSOR LOCATION
0004DB	19			3160	LODI DE,-(SCRNSIZE-LINESIZE)+1	
0004DC	3A0221			3162	ADD HL,DE	-(SCREENSIZE-LINESIZE)+1
0004DF	2F			3164	LD A,CURSX	
0004E0	85			3166	CMA	
0004E1	6F7C			3168	ADD L	-XLOC-1
0004E3	CEFF			3170	LOD LA,AH	
0004E5	67			3172	ADCI X'FF'	
0004E6	220C21			3174	LOD H,A	
0004E9	EB			3176	ST HL,NXTDISA	THAT RESULT IN NEW START OF SCREEN
0004EA	2A6B21			3178	XCH HL,DE	SAVE THE NEW NXTDISA
0004ED	424B			3180	LD HL,TEXTTOP	SEE IF THAT PUTS TEXTTOP ON SCREEN
0004EF	11B50B			3182	LOD BC,DE	
0004F2	CD7C07			3184	LODI DE,SCRNSIZE	BY USING CLEVER CIRCLEFN
0004F5	D20105			3186	CALL CIRCLEFN	
0004F8	2A6B21			3188	JMP NC,HOMEDONE	IT DOESN'T SO GO STORE IT
0004FB	CD6B07			3190	LD HL,TEXTTOP	IT DOES SO MAKE NXTDISA=TEXTTOP
				3192	CALL MINUSHL	926.

LOC	OBJECT	CODE	ADDR1	ADDR2	STMT	SOURCE	STATEMENT	ASM H V	05 01.23	10/02/76
0004FE	220C21				3194		ST HL,NXTDISA			927.
000501	E1				3196	HOMEDONE	POP HL	REMEMBER WHERE IT WAS		928.
000502	3A0C21				3198		LD A,NXTDISA	SEE IF IT IS THE SAME		929.
000505	BD				3200		CMP L			930.
000506	C20E05				3202		JMP NZ,NOTSAME	IF NOT, STORE OLD ADDRESS		931.
000509	3A0D21				3204		LD A,NXTDISA+1			932.
00050C	BC				3206		CMP H			933.
00050D	C8				3208		RET Z	IF SAME, DO NOT STORE		934.
00050E	220E21				3210	NOTSAME	ST HL,SAVDISA			935.
000511	C9				3212		RET U			936.
					3215	*				938.
					3216	*	UNHOME: RETURN TO WHERE WE WERE WHEN "HOME" WAS DONE			939.
					3217	*				940.
000512	2A0E21				3218	UNHOME	LD HL,SAVDISA			941.
000515	220C21				3220		ST HL,NXTDISA			942.
000518	C9				3222		RET ,			943.
			00519		3224	MOVECURS	EQU *	MOVE CURSOR TO BOTTOM OF DISPLAY		944.
000519	2A0C21				3226		LD HL,NXTDISA			945.
00051C	11640B				3228		LODI DE,SCRNSIZE-LINESIZE			946.
00051F	19				3230		ADD HL,DE			947.
000520	220021				3232		ST HL,CURSLOC			948.
000523	97				3234		SUB A			949.
000524	320221				3236		ST A,CURSX	RESET X POSITION		950.
000527	C9				3238		RET ,			951.
					3240	*				952.
					3241	*	PAGE UP.			953.
					3242	*				954.
			00528		3243	PGUP	EQU *			955.
000528	BB				3245		CMP E	WAS LAST PAGE UP		956.
000529	211021				3247		LODI HL,SCRCTR	ADD OF SCROLLING TIME		957.
00052C	C23105				3249		JMP NZ,PGUDOIT	ITS FIRST TIME		958.
					3251	*		ELSE SEE IF TIME TO SCROLL		959.
00052F	35				3252		DEC M	DECREMENT SCROLL COUNT		960.
000530	C0				3254		RET NZ	IF NOT ZERO-->THEN NOT TIME		961.
000531	3630				3256	PGUDOIT	LODI M,SCRPTIME	RESET		962.
000533	061F				3258		LODI B,NLINES-6	DO 31 LINES		963.
000535	CD2E04				3260	KEEPUP	CALL SCRUP	DO ONE LINE		964.
000538	05				3262		DEC B	ALTER COUNT		965.
000539	C23505				3264		JMP NZ,KEEPUP	AND KEEP IT UP		966.
00053C	C9				3266		RET ,			967.
					3268	*				968.
					3269	*				969.
					3270	*	PAGEDOWN.			970.
					3271	*				971.
			0053D		3272	PGDOWN	EQU *			972.
00053D	BB				3274		CMP E	WAS LAST PAGE		973.
00053E	211021				3276		LODI HL,SCRCTR	ADD OF SCROLLING TIME		974.
000541	C24605				3278		JMP NZ,PGDDOIT	ITS FIRST TIME		975.
					3280	*		ELSE SEE IF TIME TO SCROLL		976.
000544	35				3281		DEC M	DECREMENT SCROLL COUNT		977.
000545	C0				3283		RET NZ	IF NOT ZERO-->THEN NOT TIME		978.
000546	3630				3285	PGDDOIT	LODI M,SCRPTIME	RESET		979.
000548	061F				3287		LODI B,NLINES-6			980.
00054A	CD1604				3289	KEEPUP2	CALL SCRDOWN			981.

LOC	OBJECT CODE	ADDR1	ADDR2	STMT	SOURCE	STATEMENT	ASM H V 05 01.23 10/02/76
00054D	05			3291	DEC	B	982.
00054E	C24A05			3293	JMP	NZ,KEEPUP2	983.
000551	C9			3295	RET	,	984.
				3297	*		985.
				3298	*		986.
				3299	* SET LOCAL		987.
				3300	*		988.
		00552		3301	SE TLCL	EQU *	989.
000552	3A2421			3303	LD	A,LCLMODE	990.
000555	F606			3305	ORAI	NORCV+NSEND	991.
000557	322421			3307	ST	A,LCLMODE	992.
00055A	C9			3309	RET	,	993.
				3311	*		994.
				3312	* SETREMOTE - SEND STUFF TO COMPUTER		995.
				3313	*		996.
		0055B		3314	SETRMT	EQU *	997.
00055B	3A2421			3316	LD	A,LCLMODE	998.
00055E	E620			3318	ANDI	PATFLAG	999.
000560	3E22			3320	LODI	A,PATFLAG+NSEND	1000.
000562	C26605			3322	JMP	NZ,REMTDOIT	1001.
000565	97			3324	SUB	A	1002.
000566	322421			3326	REMTDOIT	ST A,LCLMODE	1003.
000569	C9			3328	RET	,	1004.
				3330	*		1005.
				3331	*		1006.
				3332	* SETTING XMIT MODES		1007.
				3333	*		1008.
		0056A		3334	S@TOCR	EQU *	1009.
00056A	218101			3336	LODI	HL,@TOCR	1010.
00056D	3E01			3338	LODI	A,1	1011.
00056F	C37E05			3340	JMP	STXMITTP	1012.
				3342	*		1013.
		00572		3343	S@TODC1	EQU *	1014.
000572	218B01			3345	LODI	HL,@TODC1	1015.
000575	3E02			3347	LODI	A,2	1016.
000577	C37E05			3349	JMP	STXMITTP	1017.
				3351	*		1018.
		0057A		3352	S@TE1	EQU *	1019.
				3354	* THIS IS NORMAL XMIT MODE		1020.
00057A	214001			3355	LODI	HL,@TE1	1021.
00057D	97			3357	SUB	A	1022.
		0057E		3359	STXMITTP	EQU *	1023.
00057E	223E21			3361	ST	HL,XMITHAN	1024.
000581	323421			3363	ST	A,XMITTYPE	1025.
000584	CDA401			3365	CALL	XMITRCLR	1026.
000587	3E11			3367	LODI	A,@DC1	1027.
000589	323C21			3369	ST	A,DC1CNT	1028.
00058C	C9			3371	RET	,	1029.
				3373	*		1030.
				3374	* CLEAR BUFFERS		1031.
				3375	*		1032.
		0058D		3376	CLRBUFS	EQU *	1033.
00058D	CDA401			3378	CALL	XMITRCLR	1034.
000590	CD9501			3380	CALL	RCVCLR	1035.
000593	C9			3382	RET	,	1036.
				3384	*		1037.

3385 *
3386 STPRV EQU *

SAVES CMD AS PREVIOUS TEXT CHAR

1038.
1039.

LCC	OBJECT CODE	ADDR1	ADDR2	STMT	SOURCE STATEMENT	ASM H V 05 01.23 10/02/76
			00594	3387		
000594	7A			3388	LOD A,D	GET COMMAND 1040.
000595	F680			3390	IORI X'80'	SHOW THAT IT WAS COMD 1041.
000597	320521			3392	ST A,PREVCHAR	SAVE IT 1042.
00059A	C3FB05			3394	JMP SEMILCL	SET SEMILOCAL MODE 1043.
				3396	*	1044.
				3397	* INVERT VIDEO POLARITY	1045.
				3398	*	1046.
		0059D		3399	VIDEOINV EQU *	INVERT VIDEO 1047.
00059D	BB			3401	CMP E	SEE IF SAME AS LAST TIME 1048.
00059E	C8			3403	RET Z	YES, THEN DON'T INVERT 1049.
00059F	3A1F21			3405	LD A,MODEBITS	1050.
0005A2	EE80			3407	XORI REVRVID	INVERT IT 1051.
0005A4	D380			3409	OUT MODESET	1052.
0005A6	321F21			3411	ST A,MODEBITS	AND SAVE IT 1053.
0005A9	C9			3413	RET ,	1054.
0005AA	BB			3415	CMP E	THIS JUST PUTS THE CHAR IN BUFFER 1055.
0005AB	C8			3417	RET Z	RETURN IF LAST WAS THE SAME 1056.
0005AC	F680			3419	IORI X'80'	SHOW IT'S COMMAND 1057.
0005AE	C3B102			3421	JMP KEYSTORE	GO AND STORE 1058.
0005B1	BB			3423	CMP E	1059.
0005B2	C8			3425	RET Z	1060.
0005B3	3E42			3427	CCBSET LOADER,EP=LOADST,CP=DISPL,CS=GETCHR	1061.
0005B5	322420			3429		
0005B8	213109			3430		
0005BB	224620			3431		
0005BE	3E6C			3432		
0005C0	322620			3433		
0005C3	3E00			3434		
0005C5	322720			3435		
0005C8	3E24			3436	CCBSET GETCHR,CP=LOADER	INITIALIZE COROUTINE 1062.
0005CA	326F20			3438		
0005CD	3E24			3439	CCBSET DISPL,CS=LOADER	CONTROL BLOCKS 1063.
0005CF	320220			3441		
0005D2	CDFB05			3442	CALL SEMILCL	SET SEMILOCAL MODE 1064.
0005D5	3E1A			3444	LODI A,@SUB	DISPLAY A RIGHT ARROW 1065.
0005D7	C3B102			3446	JMP U,KEYSTORE	AND GO HOME 1066.
		005DA		3448	HEXINIT EQU *	1067.
0005DA	BB			3450	CMP E	1068.
0005DB	C8			3452	RET Z	1069.
0005DC	3E66			3454	CCBSET HEXER,EP=HEXEP,CP=DISPL,CS=GETCHR	1070.
0005DE	324820			3456		
0005E1	21510A			3457		
0005E4	226A20			3458		
0005E7	3E6C			3459		
0005E9	324A20			3460		
0005EC	3E00			3461		
0005EE	324B20			3462		
0005F1	3E48			3463	CCBSET GETCHR,CP=HEXER	1071.
0005F3	326F20			3465		
0005F6	3E48			3466	CCBSET DISPL,CS=HEXER	1072.
0005F8	320220			3468		
0005FB	3A2421			3469	SEMILCL LD A,LCLMODE	SET SEMILOCAL MODE (NOSEND) 1073.
0005FE	E604			3471	ANDI NORCV	SEE IF WE IS REALLY IN LOCAL MODE 1074.
000600	3E22			3473	LODI A,PATFLAG+NOSEND	ASSUME NOT AND LOAD USUAL FLAGS 1075.

000602 CA0706

3475
3477

JMP Z,STOREIT
LODI A,PATFLAG+NORCV+NOSEND

GO STORE IF OK.
WE WAS ALREADY LOCAL, SO

1076.
1077.

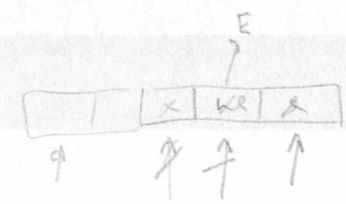
LOC	OBJECT CODE	ADDR1	ADDR2	STMT	SOURCE	STATEMENT	ASM H V 05 01.23 10/02/76
000605	3E26			3478			
000607	322421			3479	STOREIT	ST A,LCLMODE	1078.
00060A	C9			3481	RET	U	1079.

LOC	OBJECT CODE	ADDR1	ADDR2	STMT	SOURCE STATEMENT	ASM H V 05 01.23 10/02/76
				3484	*****	1081.
				3485	*	1082.
				3486	* HERE ARE THE GLOBAL SYSTEM ROUTINES	1083.
				3487	*	1084.
				3488	*****	1085.
00060B	E9			3489	JUMPER JMP (HL)	1086.
00060C	C5D5			3491	RESUMER PUSH BC,DE SAVE REGS	1087.
00060E	6E			3493	LOD L,M HL NOW POINTS TO STACK POINTER	1088.
00060F	6E			3495	LOD L,M HL NOW IS STACKPTR OF NEW TASK	1089.
000610	F9			3497	LOD SP,HL	1090.
000611	D1C1			3499	POP DE,BC RESTORE REGS	1091.
000613	C9			3501	RET U REACTIVATE NEW TASK	1092.
				3503	*****	1093.
				3504	*	1094.
				3505	* HEXIN: CONVERT ASCII CHARACTER IN 'A' TO EQUIVALENT HEX.	1095.
				3506	* IF NOT HEX, RETURN C ELSE NC.	1096.
				3507	*	1097.
				3508	* DESTROYS: A (ONLY IF HEX)	1098.
				3509	*	1099.
000614	D5			3510	HEXIN PUSH DE	1100.
000615	57			3512	LOD D,A SAVE THE CHAR	1101.
000616	FE30			3514	CMPI '0'	1102.
000618	FA3A06			3516	JMP S,BADHEX SMALLER THEN 0	1103.
00061B	FE3A			3518	CMPI ':' :=9+1	1104.
00061D	FA3606			3520	JMP S,HEXOK	1105.
000620	FE41			3522	CMPI 'A' IS IT UPPERCASE A-F?	1106.
000622	FA3A06			3524	JMP S,BADHEX TOO SMALL, NO GOOD	1107.
000625	FE47			3526	CMPI 'G'	1108.
000627	FA3406			3528	JMP S,LETTER IT IS, OK...	1109.
00062A	FE61			3530	CMPI 'A' HOW ABOUT LOWERCASE??	1110.
00062C	FA3A06			3532	JMP S,BADHEX TOO SMALL	1111.
00062F	FE67			3534	CMPI 'G'	1112.
000631	F23A06			3536	JMP NS,BADHEX TOO BIG	1113.
000634	C609			3538	LETTER ADDI 9 IT'S OK, MAKE IT RIGHT VALUE	1114.
000636	E60F			3540	HEXOK ANDI X'0F' AND CHOP OFF HIGH-ORDER BITS	1115.
000638	D1			3542	POP DE	1116.
000639	C9			3544	RET	1117.
00063A	37			3546	BADHEX STC SET CARRY TO INDICATE NOT HEX	1118.
00063B	7A			3548	LOD A,D REMEMBER CHAR	1119.
00063C	D1			3550	POP DE AND RETRIEVE DE	1120.
00063D	C9			3552	RET U AND RETURN	1121.
				3554	*****	1122.
				3555	*	1123.
				3556	* SEARCH: SEARCH THE COMMAND TABLE WHOSE ADDRESS IS IN HL	1124.
				3557	* FOR THE CHARACTER IN 'A'.	1125.
				3558	* (N.B.: THE TABLE ADDRESS IS THE FIRSTENTRY-2)	1126.
				3559	* EACH TABLE ENTRY IS OF THE FORM:	1127.
				3560	* DC 'C'CHARACTER',AL2(UNREVERSED ADDRESS)	1128.
				3561	* THE LAST ENTRY IN THE TABLE MUST BE X'00' FOLLOWED	1129.
				3562	* BY THE ADDRESS OF THE ROUTINE TO BE EXECUTED IF	1130.
				3563	* THE CHARACTER IS NOT IN THE TABLE.	1131.
				3564	*	1132.
00063E	57			3565	SEARCH LOD D,A SAVE THE CHAR	1133.
00063F	23			3567	SEARCHLP INC HL INC TO RIGHT PLACE	1134.
000640	23			3569	INC HL	1135.

000641	7E	3571		L0D	A,M	GET CONTENTS	1136.
000642	23	3573		INC	HL		1137.
000643	BA	3575		CMP	D	SEE IF ITS THE CHAR	1138.
000644	CA4B06	3577		JMP	Z,GOTCMD	IF SO GET ADDRESS	1139.
000647	B7	3579		ORA	A	NO SEE IF ZERO	1140.
000648	C23F06	3581		JMP	NZ,SEARCHLP	ITS NOT LOOP AROUND AND TRY AGAIN	1141.
00064B	5E	3583	GOTCMD	L0D	E,M	GET HIGH ORDER ADDRESS	1142.
00064C	23	3585		INC	HL		1143.
00064D	6E	3587		L0D	L,M		1144.
00064E	63	3589		L0D	H,E	AND LOW ORDER	1145.

L,M | H,M
 H,E | L,E

for reversed byte table



LOC	OBJECT CODE	ADDR1	ADDR2	STMT	SOURCE STATEMENT	ASM H V 05 01.23 10/02/76
00064F	7A			3591	LOD A,D	RESTORE THE CHAR 1146.
000650	C9			3593	RET	1147.
				3595	*****	1148.
000651	F5			3596	SEND BYTE PUSH AF	SAVE THE CHAR 1149.
000652	E6F0			3598	ANDI X'F0'	COMPUTE FIRST DIGIT 1150.
000654	07070707			3600	ROT L,4	1151.
000658	CD7906			3602	CALL HEXCUT	1152.
00065B	S7			3604	LOD D,A	1153.
00065C	F3			3606	DI ,	DISABLE TO DO CHAR 1154.
00065D	CD6702			3608	CALL KEYDOIT	OUTPUT FIRST DIGIT 1155.
000660	FB			3610	EI ,	RENABLE 1156.
000661	F1			3612	POP AF	RETRIEVE CHAR 1157.
000662	E60F			3614	ANDI X'0F'	GET SECOND DIGIT 1158.
000664	CD7906			3616	CALL HEXCUT	1159.
000667	S7			3618	LOD D,A	1160.
000668	F3			3620	DI ,	DISABLE TO DO CHAR 1161.
000669	CD6702			3622	CALL KEYDOIT	OUTPUT 1162.
00066C	FB			3624	EI ,	RENABLE 1163.
00066D	C9			3626	RET ,	GO HOME 1164.
				3628	*****	1165.
				3629	*	1166.
				3630	*	1167.
				3631	*	1168.
				3632	*	1169.
				3633	*	1170.
00066E	CD1406			3634	HEXSTR CALL HEXIN	CONVERT TO HEX 1171.
000671	D8			3636	RET C	1172.
000672	29			3638	ADD HL,HL	ROTATE FOUR BITS LEFT 1173.
000673	29			3640	ADD HL,HL	1174.
000674	29			3642	ADD HL,HL	1175.
000675	29			3644	ADD HL,HL	1176.
000676	85			3646	ADD L	BUT ADD NEW DIGIT (AND CLEAR C) 1177.
000677	6F			3648	LOD L,A	GET ADDRESS IN HL 1178.
000678	C9			3650	RET U	AND RETURN 1179.
				3652	*****	1180.
000679	FE0A			3653	HEXCUT CMPI 10	1181.
00067B	3F			3655	CMC	1182.
00067C	CE30			3657	ADCI X'30'	1183.
00067E	27			3659	DAA	1184.
00067F	C9			3661	RET ,	1185.

LOC	OBJECT CODE	ADDR1	ADDR2	STMT	SOURCE STATEMENT	ASM H V 05 01.23 10/02/76
3664	*				*****	1187.
3665	*					1188.
3666	*				CLEAR: ROUTINE TO CLEAR THE SCREEN (IT RHYMES)	1189.
3667	*					1190.
3668	*				THIS ROUTINE WILL CLEAR THE SCREEN	1191.
3669	*				DEPENDING HOW THE A REG SEEMS	1192.
3670	*				IF THE 40 BIT IS SET	1193.
3671	*				THEN ALL ZEROS WILL IT GET	1194.
3672	*				OTHERWISE A BLANK IS USED	1195.
3673	*				SO TEXT MODE WON'T BE CONFUSED	1196.
3674	*					1197.
3675	*				THE 20 BIT IS QUITE A FELLOW	1198.
3676	*				AND IN ITS CLEAR IS VERY MELLOW	1199.
3677	*				FOR ONLY ONE PAGE DOTH IT ZAP	1200.
3678	*				TO RID THE SCREEN OF CRUFTY CRAP	1201.
3679	*				BUT DON'T DESPAIR, WE ARE NOT DONE	1202.
3680	*				FOR WHEN WE FIND THE LOW BIT 1	1203.
3681	*				A BANNER ON THE SCREEN WILL GLOW	1204.
3682	*				TO SHOW THE CHARACTERS WE KNOW.	1205.
3683	*					1206.
3684	*				<EHF> AND <LJS> TAKE FULL RESPONSIBILITY FOR THE ABOVE.	1207.
3685	*					1208.
3686	*					1209.
3687	*				EQUATES FOR CLEAR TYPES	1210.
00080	3688	CLRFCLR	EQU	X'80'	INDICATES A CLEAR	1211.
00040	3690	CLRFZERO	EQU	X'40'	CLEAR MEMORY TO X'00'	1212.
00020	3692	CLRFPAGE	EQU	X'20'	ONLY CLEAR SCRNSIZE NUMBER OF CHAR.	1213.
00001	3694	CLRBANR	EQU	X'01'	PUT BANNER ON SCREEN	1214.
	3696	*				1215.
00680	3697	CLEAR	EQU	*		1216.
	3699	*				1217.
	3700	*			CLEAR THE SCREEN	1218.
	3701	*				1219.
000680	1717	3702	ROT	LC,2	BIT X'40' MEANS ZERO IT	1220.
000682	0620	3704	LODI	B,' '	ELSE BLANK IT	1221.
000684	D28906	3706	JMP	NC,CLEARIT		1222.
000687	0600	3708	LODI	B,0		1223.
		3710	*			1224.
	00689	3711	CLEAR IT	EQU	*	1225.
		3713	ROT	LC	SEE IF JUST PAGE CLEAR	1226.
00068A	D2A106	3715	JMP	NC,CLEARIT2	NO, FULL MEMORY CLEAR	1227.
00068D	2A0C21	3717	LD	HL,NXTDISA	PLACE TO CLEAR FROM	1228.
000690	EB	3719	XCH	HL,DE		1229.
000691	21B50B	3721	LODI	HL,SCRNSIZE	AMOUNT TO CLEAR	1230.
000694	78	3723	LOD	A,B	THING TO CLEAR TO.	1231.
000695	01FFFF	3725	LODI	BC,-1	THING TO DECREMTN BY	1232.
	00698	3727	CLRPAG	EQU	*	1233.
		3729	ST	A,(DE)		1234.
000699	13	3731	INC	DE	UPDATE THAT	1235.
00069A	09	3733	ADD	HL,BC	UPDATE COUNT	1236.
00069B	DA9806	3735	JMP	C,CLRPAG	NOT DONE YET	1237.
00069E	C3F406	3737	JMP	LVCLR	GO AND LEAVE	1238.
0006A1	2A6721	3739	CLEARIT2	LD	HL,TEXT	1239.
0006A4	220C21	3741	ST	HL,NXTDISA		1240.
		3743	*		THIS IS THE SUPER FAST CLEAR LOOP. IT DOES DOUBLE STORES	1241.

	3744	* USING THE STACK		1242.	
	3745	*		1243.	
0006A7	3A1F21	LD	A,MODEBITS	SET QUICK-MODE ACCESS TO RAM	1244.
0006AA	F642	IORI	QUICKMD+SCRNBLNK	SET QUICK MODE,AN TURN OFF SCREEN	1245.
0006AC	D380	OUT	MODESET		1246.
0006AE	78	LOD	A,B	SEE IF WE ARE DOING GRAPHIC CLEAR	1247.
0006AF	B7	ORA	A	BY CHECKING IF B IS 0	1248.
0006B0	2A6F21	LD	HL,WASTED	INITIALIZE TEXTBOT AND TOP	1249.
0006B3	226D21	ST	HL,TEXTBOT		1250.
0006B6	2A6921	LD	HL,MTEXT	AN JUST CLEAR TO TEXT ADDR	1251.
0006B9	226B21	ST	HL,TEXTTOP		1252.
	3764	JMP	NZ,GOCLR		1253.

LOC	OBJECT CODE	ADDR1	ADDR2	STMT	SOURCE	STATEMENT	ASM H V 05 01.23 10/02/76
0006BC	C2C206			3765			
0006BF	2110A7			3766			
0006C2	E5			3768	GOCLR	LODI HL,--GRAPHEVN	1254.
0006C3	210000			3770		PUSH HL	1255.
0006C6	39			3772		LODI HL,0	1256.
0006C7	222521			3774		ADD HL,SP	1257.
				3776	*	ST HL,SAVSTK	1258.
				3777			1259.
0006CA	48			3777		LOD C,B	1260.
0006CB	11ECFF			3779		LODI DE,-20	1261.
0006CE	E1			3781		POP HL	1262.
0006CF	E5			3783		PUSH HL	1263.
0006D0	19			3785		ADD HL,DE	1264.
0006D1	3F			3787		CMC ,	1265.
0006D2	7C			3789		LOD A,H	1266.
0006D3	1F			3791		ROT RC	1267.
0006D4	67			3793		LOD H,A	1268.
0006D5	7D			3795		LOD A,L	1269.
0006D6	1F			3797		ROT RC	1270.
0006D7	6F			3799		LOD L,A	1271.
0006D8	11FFFF			3801		LODI DE,-1	1272.
0006DB	310000			3803		LODI SP,0	1273.
				3805	*		1274.
		006DE		3806	CLEARLP	EQU *	1275.
0006DE	C5			3808		PUSH BC	1276.
0006DF	19			3810		ADD HL,DE	1277.
0006E0	DADE06			3812		JMP C,CLEARLP	1278.
				3814	*		1279.
0006E3	2A2521			3815		LD HL,SAVSTK	1280.
0006E6	F9			3817		LOD SP,HL	1281.
				3819	*		1282.
				3820	*		1283.
0006E7	E1			3821		POP HL	1284.
0006E8	CD6B07			3823		CALL MINUSHL	1285.
0006EB	78			3825		LOD A,B	1286.
0006EC	1E14			3827		LODI E,20	1287.
		006EE		3829	CLEARLP2	EQU *	1288.
0006EE	77			3831		ST A,(HL)	1289.
0006EF	1D			3833		DEC E	1290.
0006F0	23			3835		INC HL	1291.
0006F1	C2EE06			3837		JMP NZ,CLEARLP2	1292.
				3839	*		1293.
		006F4		3840	LVCLR	EQU *	1294.
0006F4	3A1F21			3842		LD A,MODEBITS	1295.
0006F7	D380			3844		OUT MODESET	1296.
0006F9	3E20			3846		LODI A,' '	1297.
0006FB	320421			3848		ST A,CURCHAR	1298.
0006FE	97			3850		SUB A	1299.
0006FF	320221			3852		ST A,CURSX	1300.
000702	211821			3854		LODI HL,CLRFLG	1301.
000705	7E			3856		LOD A,M	1302.
000706	3600			3858		LODI M,0	1303.
000708	1F			3860		ROT RC	1304.
000709	D0			3862		RET NC	1305.
00070A	114401			3864		LODI DE,4*LINE SIZE	1306.
00070D	2A6721			3866		LD HL,TEXT	1307.

LOC	OBJECT CODE	ADDR1	ADDR2	STMT	SOURCE	STATEMENT	ASM H V 05 01.23	10/02/76
000710	19			3868		ADD HL,DE		1308.
				3870	* PRINT	CHARACTER SET		1309.
000711	97			3871		SUB A		1310.
000712	77			3873	RESETCL	LOD M,A		1311.
000713	23			3875		INC HL		1312.
000714	3C			3877		INC A		1313.
000715	F21207			3879		JMP NS,RESETCL		1314.
000718	110A00			3881		LODI DE,10		1315.
00071B	19			3883		ADD HL,DE		1316.
00071C	113E07			3885		LODI DE,DATE	PRINT OUT HELPFUL MESSAGE	1317.
00071F	0612			3887		LODI B,18	FOR FXR	1318.
000721	1A			3889	DODATE	LD A,(DE)	GET A CHAR	1319.
000722	77			3891		ST A,(HL)	PUT IT ON SCREEN	1320.
000723	13			3893		INC DE	MOVE POINTER	1321.
000724	23			3895		INC HL	AND MOVE OTHER POINTER	1322.
000725	05			3897		DEC B	DECREMENT COUNT	1323.
000726	C22107			3899		JMP NZ,DODATE	CONTINUE TIL DONE	1324.
000729	F3			3901		DI	DISABLE TO CLEAR BUFFERS	1325.
00072A	CD8D05			3903		CALL CLRBUFS	GO EMPTY RECIEVE AND XMIT BUFS	1326.
00072D	11E601			3905		LODI DE,6*LINE SIZE		1327.
000730	2A6721			3907		LD HL,TEXT		1328.
000733	19			3909		ADD HL,DE		1329.
000734	220021			3911		ST HL,CURSLOC		1330.
000737	3EBF			3913		LODI A,X'BF'	GET HELP='?'+X'80'	1331.
000739	CD4C02			3915		CALL CHARPROC		1332.
00073C	FB			3917		EI		1333.
00073D	C9			3919		RET		1334.
00073E	417373656D626C65			3921	DATE	CHAR 'ASSEMBLED'		1335.
000746	6420			3923				
000748	31302F30322F3736			3924		CHAR 'ESYS DATE'		1336.
				3926	*****	*****		1337.
000750	115100			3927	ADDWRP81	LODI DE,LINESIZE	ADDS 81 TO HL WITH WRAP AROUND	1338.
000753	19			3929	ADDWRP	ADD HL,DE	ACTUALLY YOU CAN ADD ANYTHING	1339.
000754	D0			3931		RET NC	RETURN IF IT'S UNDER X'10000'	1340.
000755	3A6821			3933		LD A,TEXT+1	IF NOT WRAP TO TEXT	1341.
000758	84			3935		ADD H	ADD ON LEFTOVER	1342.
000759	67			3937		LOD H,A		1343.
00075A	C9			3939		RET	AND GO BACK	1344.
				3941	*****	*****		1345.
00075B	11AFFF			3942	SUBWRP81	LODI DE,-LINESIZE	VERY SIMILAR TO ABOVE ROUTINE	1346.
00075E	19			3944	SUBWRP	ADD HL,DE	IN FACT THERE ARE ONLY TRIVIAL	1347.
00075F	3A6821			3946		LD A,TEXT+1	DIFFERENCES	1348.
000762	BC			3948		CMP H	WHICH I'M SURE YOU CAN FIGURE OUT	1349.
000763	C8			3950		RET Z	IN A RELATIVELY SHORT PERIOD	1350.
000764	D8			3952		RET C	OF TIME...	1351.
000765	EB			3954		XCH HL,DE	SAY FIVE OR SIX HOURS.	1352.
000766	2A6921			3956		LD HL,MTEXT	JUST KIDDING REALLY.	1353.
000769	19			3958		ADD HL,DE	DON'T YOU HATE---	1354.
00076A	C9			3960		RET	"FUNNY" DOCUMENTATION.	1355.
				3962	*****	*****		1356.
00076B	7C			3963	MINUSHL	LOD A,H	THIS ROUTINE COMPLEMENTS HL	1357.
00076C	2F			3965		CMA	IN THE 2'S FASHION	1358.
00076D	67			3967		LOD H,A	WE JUST COMPLEMENTED H	1359.

LOC	OBJECT CODE	ADDR1	ADDR2	STMT	SOURCE STATEMENT	ASM H V 05 01.23 10/02/76
00076E	7D			3969	LOD A,L	NOW DO L 1360.
00076F	2F			3971	CMA ,	1361.
000770	6F			3973	LOD L,A	1362.
000771	23			3975	INC HL	AND INCREMENT THE RESULT 1363.
000772	C9			3977	RET ,	1364.
				3979	*****	1365.
000773	CD6B07			3980	NEGADDWR CALL MINUSHL	THIS ADDS 81 TO A NUMBER 1366.
000776	CD5007			3982	CALL ADDWRP81	WHICH IS STORED IN MINUS FORMAT 1367.
000779	C36B07			3984	JMP MINUSHL	LIKE TEXTTOP AND TEXTBOT 1368.
				3986	*****	1369.
00077C	C5			3987	CIRCLEFN PUSH BC	SNEAKY JEZ ROUTINE TO FIGURE OUT 1370.
00077D	E3			3989	XCH HL,(SP)	IF X<Y<=X+Z (X+Z IS WRAP SUM) 1371.
				3991	*	OR IF Y<X+Z<X OR IF X+Z<X<Y 1372.
				3992	*	THIS IS THE WAY TO CHECK FOR 1373.
				3993	*	WRAPAROUND CONDITIONS LIKE 1374.
				3994	*	WHETHER TEXTOP IS ON THE CURRENT 1375.
				3995	*	PAGE (FOR EXAMPLE). 1376.
00077E	CD5307			3996	CALL ADCWRP	UPON CALLING, BC=X,DE=Z,HL=-Y 1377.
000781	D1			3998	POP DE	AT THIS POINT, BC=X,DE=-Y,HL=X+Z 1378.
000782	7C			4000	LOD A,H	NOW WE COMPARE X AND X+Z 1379.
000783	90			4002	SUB B	ASSUMING THAT Z<X'F000'-TEXT 1380.
000784	F5			4004	PUSH AF	(WHICH IS TRUE WHENEVER I CALL IT) 1381.
000785	19			4006	ADD HL,DE	SAVE FLAG AND COMPUTE (X+Z)-Y 1382.
000786	F5			4008	PUSH AF	SAVE THAT FLAG TOO. 1383.
000787	EB			4010	XCH HL,DE	NOW WE ADD X AND -Y 1384.
000788	09			4012	ADD HL,BC	1385.
000789	9F			4014	SBB A	COPY CARRY TO ALL BITS IN A 1386.
00078A	D1			4016	POP DE	PUT OLD FLAGS INTO E 1387.
00078B	AB			4018	XOR E	AND XOR WITH RECENTEST FLAG 1388.
00078C	D1			4020	POP DE	THERE ARE THREE RESULTS TO BE 1389.
00078D	AB			4022	XOR E	COMBINED 1390.
00078E	0F			4024	ROT R	PUT RESULT IN CARRY FLAG... 1391.
00078F	C9			4026	RET ,	AND GO BACK HOME. 1392.

LOC	OBJECT CODE	ADDR1	ADDR2	STMT	SOURCE STATEMENT	ASM H V 05 01.23 10/02/76
				4029	*****	1394.
				4030	* RESET, PART II	1395.
				4031	*	1396.
000790	97			4032	RESET II SUB A	1397.
000791	321921			4034	ST A,FLAGS	1398.
000794	323121			4036	ST A,CNTLOMD INITIAL NOT FULLDUPLEX	1399.
000797	323221			4038	ST A,ALLOWGMD INITIALLY ALLOW ENTRY INTO GRAPHMD	1400.
00079A	323D21			4040	ST A,SHIFTMD	1401.
00079D	322421			4042	ST A,LCLMODE SET REMOTE MODE	1402.
0007A0	326021			4044	ST A,TYPE	1403.
0007A3	D3A6			4046	OUT MODESET2	1404.
0007A5	3E66			4048	LODI A,B'01100110' INITIAL UART SPEED OF 9600(RCV&XMIT)	1405.
0007A7	D38E			4050	OUT URTSPEED	1406.
0007A9	DB83			4052	INP SCRWHEEL GET WHEEL CONTENTS	1407.
0007AB	E6FE			4054	ANDI X'FE' KILL LOW BIT	1408.
0007AD	322F21			4056	ST A,PREVSCRL AND STORE IT	1409.
0007B0	210E28			4058	LODI HL,CHGENROM+13+1	1410.
0007B3	226521			4060	ST HL,GRPCSET	1411.
0007B6	21E302			4062	LODI HL,KEYDEF SET DEFAULT FOOADDR	1412.
0007B9	222D21			4064	ST HL,FOOADDR	1413.
0007BC	213E00			4066	LODI HL,ROMINT SET INT. HANDLER TO ROM	1414.
0007BF	222B21			4068	ST HL,INTROUT	1415.
0007C2	219804			4070	LODI HL,CHL1TAB SAVE STARTING NXTCHL1	1416.
0007C5	220A21			4072	ST HL,NXTCHL1	1417.
0007C8	210044			4074	LODI HL,ITEXT SAVE INITIAL WRAP VALUE	1418.
0007CB	226721			4076	ST HL,TEXT	1419.
0007CE	2100BC			4078	LODI HL,-ITEXT AND MINUS IT TOO, FOR CONVENIENCE	1420.
0007D1	226921			4080	ST HL,MTEXT	1421.
0007D4	226B21			4082	ST HL,TEXTTOP CAUSE IT'S EASIER THAT WAY.	1422.
0007D7	210E00			4084	LODI HL,IWASTED	1423.
0007DA	226D21			4086	ST HL,TEXTBOT	1424.
0007DD	226F21			4088	ST HL,WASTED	1425.
0007E0	3EC9			4090	LODI A,X'C9' SAVE A RETURN AT GRPDRAW+2	1426.
0007E2	327321			4092	ST A,GRPDRAW+2 SO WE WON'T HAVE TO DO IT LATER.	1427.
0007E5	3E8A			4094	CCBSET GETCHR,CP=DISPL,EP=GETCHAR	1428.
0007E7	326C20			4096		
0007EA	210F08			4097		
0007ED	228E20			4098		
0007F0	3E00			4099		
0007F2	326F20			4100		
0007F5	3E1E			4101	CCBSET DISPL,CS=GETCHR,EP=TXTMOD	1429.
0007F7	320020			4103		
0007FA	213E0C			4104		
0007FD	222220			4105		
000800	3E6C			4106		
000802	320220			4107		
000805	216608			4108	LODI HL,GETBUF INITIALIZE TEXT SOURCE	1430.
				4110	ST HL,SOURCE TO GETBUF	1431.

for 300 baud
55

LOC	OBJECT CODE	ADDR1	ADDR2	STMT	SOURCE STATEMENT	ASM H V 05 01.23 10/02/76
000808	221A21			4111		
000808	319020			4112	LODI SP,GETCHRST	1432.
00080E	FB			4114	EI .	1433.

50
75
150
300

LOC	OBJECT CODE	ADDR1	ADDR2	STMT	SOURCE STATEMENT	ASM H V 05 01.23 10/02/76
				4117	*****	1435.
				4118	*	1436.
				4119	* GETCHR---THE COROUTINE WHICH SUPPLIES CHARACTERS	1437.
				4120	*	1438.
				4121	* GET CHARACTER FROM CURRENT SOURCE. ROUTINE IS INVOKED	1439.
				4122	* USING THE RESUME MACRO. THE CHARACTER IS RETURNED	1440.
				4123	* IN THE A REGISTER. IT RETURNS (BY DOING A RESUME OF	1441.
				4124	* ITS CHARACTER PROCESSOR (I.E. CP) .	1442.
				4125	* IT DOES A CALL INDIRECT OF 'SOURCE' TO GET	1443.
				4126	* THE CHARACTER	1444.
				4127	*	1445.
				4128	*	1446.
				4129	*	1447.
				4130	*****	1448.
		0080F		4131	GETCHAR EQU *	1449.
00080F	2A1A21			4133	CALL @ SOURCE GET A CHAR FROM SOMEBODY	1450.
000812	CD0B06			4135		
000815	57			4136	LOD D,A	1451.
000816	210521			4138	LODI HL,PREVCHAR	1452.
000819	4E			4140	LOD C,M GET PREVIOUS CHAR	1453.
00081A	3A0621			4142	LD A,ESCCHAR	1454.
00081D	B9			4144	CMP C	1455.
00081E	CAB408			4146	JMP Z,DOHOSTCD	1456.
000821	BA			4148	CMP D	1457.
000822	CA5D08			4150	JMP Z,SAVEPREV	1458.
000825	79			4152	LOD A,C	1459.
000826	FEA1			4154	CMPI X'21'+X'80'	1460.
000828	CA8F08			4156	JMP Z,CHGESC	1461.
00082B	FED5			4158	CMPI X'55'+X'80'	1462.
00082D	CA9908			4160	JMP Z,CHGUART	1463.
000830	7A			4162	LOD A,D	1464.
000831	B7			4164	ORA A	1465.
000832	F25108			4166	JMP NS,SENDCHAR	1466.
000835	213A08			4168	LODI HL,GTCHRTAB	1467.
000838	CD3E06			4170	CALL SEARCH	1468.
00083B	E9			4172	JMP (HL)	1469.
		0083A		4174	GTCHRTAB CMDEF 91,STATOUT	1470.
00083C	9108C3			4176		
00083F	BF08D9			4177	CMDEF BF,HLPINIT	1471.
000842	FF08E8			4179	CMDEF FF,BUFINIT	1472.
000845	8E090E			4181	CMDEF 8E,SHIFTOUT	1473.
000848	8F0919			4183	CMDEF 8F,SHIFTIN	1474.
00084B	9308FA			4185	CMDEF 93,CHARSET	1475.
00084E	000851			4187	CMDEF 00,SENDCHAR	1476.
				4189	****	1477.
				4190	****	1478.
				4191	****	1479.
		00851		4192	SENDCHAR EQU *	1480.
000851	21FAFF			4194	RESUME GETCHR,CP	1481.
000854	39			4196		
000855	226C20			4197		
000858	2E6F			4198		
00085A	CD0C06			4199		
00085D	7A			4200	SAVEPREV LOD A,D	1482.
00085E	320521			4202	ST A,PREVCHAR	1483.

000861 C30F08

4204

JMP U,GETCHAR

AND LOOP

1484.

LOC	OBJECT CODE	ADDR1	ADDR2	STMT	SOURCE STATEMENT	ASM H V 05 01.23 10/02/76
				4207	*****	1486.
				4208	*	1487.
				4209	* GETBUF -- THE ENABLED WAIT LOOP -- DEFAULT CHARACTER SOURCE	1488.
				4210	*	1489.
				4211	*****	1490.
				4212	*	1491.
000864	FB			4213	WAIT EI	1492.
000865	76			4215	HLT ,	1493.
000866	3A1821			4217	GETBUF LD A,CLRFLG	1494.
000869	B7			4219	IOR A	1495.
00086A	FC8006			4221	CALL S,CLEAR	1496.
00086D	F3			4223	DI ,	1497.
				4225	* DISABLE WHILE CHECKING POINTERS	1498.
				4226	* CHECK FOR BUFFERED CHARACTERS	1499.
				4227	*	1500.
00086E	2A2221			4228	LD HL,GETPTR	1501.
000871	3A2021			4230	LD A,PUTPTR	1502.
000874	BD			4232	CMP L	1503.
000875	C27F08			4234	JMP NZ,BUFGET1	1504.
000878	3A2121			4236	LD A,PUTPTR+1	1505.
00087B	BC			4238	CMP H	1506.
00087C	CA6408			4240	JMP Z,WAIT	1507.
00087F	23			4242	BUFGET1 INC HL	1508.
000880	7C			4244	LOD A,H	1509.
000881	FE44			4246	CMPI BUFEND,<	1510.
000883	C28908			4248	JMP NZ,BUFGET2	1511.
000886	210041			4250	LODI HL,BUFFER	1512.
000889	222221			4252	BUFGET2 ST HL,GETPTR	1513.
00088C	7E			4254	LD A,(HL)	1514.
00088D	FB			4256	EI ,	1515.
00088E	C9			4258	RET U	1516.
				4260	*****	1517.
				4261	*	1518.
				4262	* VARICUS ROUTINES JUMPED TO BY THE GETCHAR ESCAPE PROCESSOR.	1519.
				4263	*	1520.
				4264	*****	1521.
				4265	*	1522.
00088F	7A	0088F		4266	CHGESC EQU * CHANGE ESCAPE	1523.
000890	320621			4268	LOD A,D	1524.
000893	CD420A			4270	ST A,ESCCHAR	1525.
000896	C35D08			4272	CALL UNDOSEMI	1526.
				4274	JMP SAVEPREV	1527.
				4276	*	1528.
				4277	*	1529.
				4278	* CHANGE USART SPEED	1530.
				4279	*	1531.
000899	7A	00899		4280	CHGUART EQU * CHG UART SPEED	1532.
00089A	CD1406			4282	LOD A,D	1533.
00089D	DAAF08			4284	CALL HEXIN	1534.
0008A0	57			4286	JMP C,SPDERR	1535.
0008A1	07070707			4288	LOD D,A	1536.
0008A5	B2			4290	ROT L,4	1537.
0008A6	D38E			4292	IOR D	1538.
0008A8	97			4294	OUT URTSPEED	1539.
				4296	CUSRET SUB A CLR IT OUT	1540.

0008A9 CD420A
0008AC C35D08
0008AF D386

4298
4300
4302 SPDERR
4304

CALL UNDOSEMI
JMP SAVEPREV
OUT BELL
JMP CUSRET

UNDO SEMILOCAL MODE
AND GO HOME
TELL THE USER

1541.
1542.
1543.
1544.

LOC	OBJECT CODE	ADDR1	ADDR2	STMT	SOURCE STATEMENT	ASM H V 05 01.23 10/02/76
0008B1	C3A808			4305		
		008B4		4306	DOHOSTCD EQU *	1545.
				4308	* THIS LETS THE HOST ISSUE LOCAL FUNCTIONS	1546.
				4309	* THIS IS DONE BY DOING AN <ESC><FN>	1547.
				4310	*	1548.
				4311	* WE DO IT BY SIMULATING A INTERRUPT.	1549.
0008B4	97			4312	SUB A CLEAR OT PREVCHAR	1550.
0008B5	320521			4314	ST A,PREVCHAR AND WE ARE REALLY DOING IT	1551.
				4316	*	1552.
0008B8	7A			4317	LOD A,D GET THE CHARACTER	1553.
0008B9	F680			4319	IORI X'80' AND MAKE IT COMMAND	1554.
0008BB	F3			4321	DI , TURN OFF INTERRUPTS	1555.
0008BC	CD4C02			4323	CALL CHARPROC AND GO TO IT	1556.
0008BF	FB			4325	EI , CHARPROC RUNS DISABLED	1557.
0008C0	C30F08			4327	JMP GETCHAR AND GO DO SOMEMORE	1558.
0008C3	3A1C21			4329	STATOUT LD A,PATSTAT GET PATCHSTAT	1559.
0008C6	CD5106			4331	CALL SENDBYTE SEND THE INFO	1560.
0008C9	3E00			4333	LODI A,0	1561.
0008CB	321C21			4335	ST A,PATSTAT CLEAR PATCHSTAT	1562.
0008CE	3E0D			4337	LODI A,@CR	1563.
0008D0	57			4339	LOD D,A NOW SEND A @CR	1564.
0008D1	F3			4341	DI , DISABLE TO DO CHAR	1565.
0008D2	CD6702			4343	CALL KEYDOIT	1566.
0008D5	FB			4345	EI , RENABLE	1567.
0008D6	C30F08			4347	JMP GETCHAR AND GO HOME	1568.
0008D9	21F108			4349	HLPINIT LODI HL,PUTTER MAKE HELPST THE CHARACTER SOURCE	1569.
0008DC	221A21			4351	ST HL,SOURCE	1570.
0008DF	21001C			4353	LODI HL,HELPIFNO INITIALIZE THE "PUTIT" POINTER	1571.
0008E2	222721			4355	ST HL,POINTER	1572.
0008E5	C30F08			4357	JMP U,GETCHAR AND GO HOME	1573.
0008E8	216608			4359	BUFINIT LODI HL,GETBUF	1574.
0008EB	221A21			4361	ST HL,SOURCE MAKE GETBUF THE SOURCE	1575.
0008EE	C30F08			4363	JMP U,GETCHAR	1576.
0008F1	2A2721			4365	PUTTER LD HL,POINTER	1577.
0008F4	7E			4367	LOD A,M	1578.
0008F5	23			4369	INC HL	1579.
0008F6	222721			4371	ST HL,POINTER	1580.
0008F9	C9			4373	RET U	1581.
0008FA	3A1F21			4375	CHARSET LD A,MODEBITS	1582.
0008FD	EE20			4377	XCRI NOROMCHR	1583.
0008FF	D380			4379	OUT MODESET	1584.
000901	321F21			4381	ST A,MODEBITS	1585.
000904	3A3D21			4383	LD A,SHIFTMD	1586.
000907	B7			4385	ORA A	1587.
000908	CA1D09			4387	JMP Z,SHIFIN2	1588.
00090B	C30F08			4389	JMP GETCHAR	1589.
00090E	3E80			4391	SHIFTOUT LODI A,X'80'	1590.
000910	323D21			4393	ST A,SHIFTMD	1591.

LCC	OBJECT CODE	ADDR1	ADDR2	STMT	SOURCE	STATEMENT	ASM H V 05 01.23 10/02/76
000913	210E38			4395	SHFT OUT2	LODI HL,CHGENRAM+X'800'+13+1	BUT IT DO STORE NEW CHAR SET 1592.
000916	C32B09			4397		JMP GCHARSET	ADDRESS FOR GRAPH MODE 1593.
000919	97			4399	SHIFT IN	SUB A	HERE WE JUST CLEAR SHIFTMD 1594.
00091A	323D21			4401		ST A,SHIFTMD	1595.
00091D	3A1F21			4403	SHIFTIN2	LD A,MODEBITS	SEE IF WE ARE USING NO ROM CHR SETS 1596.
000920	E620			4405		ANDI NOROMCHR	1597.
000922	210E28			4407		LODI HL,CHGENROM+13+1	GET DEFAULT CHAR SET OFFSET 1598.
000925	CA2B09			4409		JMP Z,GCHARSET	AND DO IT IF IT'S RIGHT 1599.
000928	210E30			4411		LODI HL,CHGENRAM+13+1	GET THE OTHER ONE INSTEAD 1600.
00092B	226521			4413	GCHAR SET	ST HL,GRPCSET	AND STORE IT. 1601.
00092E	C30F08			4415		JMP GETCHAR	1602.

LOC	OBJECT CODE	ADDR1	ADDR2	STMT	SOURCE STATEMENT	ASM H V 05 01.23 10/02/76
				4418	*****	1604.
				4419	*	1605.
				4420	* THE ONE THE ONLY THE ORIGINAL	1606.
				4421	* LOADER BY YOURS TRULY,WHAT'S HIS NAME.	1607.
				4422	*	1608.
				4423	*****	1609.
000931	4F			4424	LOADST LOD C,A SAVE THE CHAR	1610.
000932	3A1C21			4426	LD A,PATSTAT MODIFY STATUS TO SHOW PATCHING	1611.
000935	F601			4428	ORAI PATCHING	1612.
000937	321C21			4430	ST A,PATSTAT	1613.
00093A	CDD209			4432	CALL BYTEIN GET A HEX BYTE,OUTPUT UPARROW	1614.
00093D	B7			4434	ORA A SEE IF LENGTH 0	1615.
00093E	CAAB09			4436	JMP Z,LOADEND IF SO, QUIT	1616.
000941	321E21			4438	ST A,PATCNT FIRST BYTE IS THE COUNT	1617.
000944	321D21			4440	ST A,PATCKSM START CHECKSUM GOING	1618.
000947	CDD209			4442	CALL BYTEIN	1619.
00094A	57			4444	LOD D,A HIGH ORDER ADDRESS	1620.
00094B	3A1D21			4446	LD A,PATCKSM	1621.
00094E	82			4448	ADD D MODIFY CHECKSUM	1622.
00094F	321D21			4450	ST A,PATCKSM AND STORE IT BACK	1623.
000952	CDD209			4452	CALL BYTEIN GET LOW ORDER ADDRESS	1624.
000955	5F			4454	LOD E,A	1625.
000956	3A1D21			4456	LD A,PATCKSM	1626.
000959	83			4458	ADD E MODIFY CKECKSUM	1627.
00095A	321D21			4460	ST A,PATCKSM	1628.
00095D	CDD209			4462	CALL BYTEIN GET TYPE	1629.
000960	B7			4464	ORA A SHOULD BE 0	1630.
000961	C2350A			4466	JMP NZ,EADTYP	1631.
000964	79			4468	LOD A,C	1632.
000965	21FAFF			4470	RESUME LOADER,CP PUT OUT SYNC CHAR	1633.
000968	39			4472		
000969	222420			4473		
00096C	2E27			4474		
00096E	CD0C06			4475		
000971	0E16			4476	LODI C,@SYN	1634.
000973	CDD209			4478	LOADING CALL BYTEIN THE MAIN LOOP	1635.
000976	12			4480	ST A,(DE)	1636.
000977	47			4482	LOD B,A SEE IF IT STUCK	1637.
000978	1A			4484	LD A,(DE)	1638.
000979	B8			4486	CMP B	1639.
00097A	C42A0A			4488	CALL NZ,NCSTICK IF NOT, TELL USER	1640.
00097D	3A1D21			4490	LD A,PATCKSM	1641.
000980	80			4492	ADD B	1642.
000981	321D21			4494	ST A,PATCKSM MODIFY CHECKSUM	1643.
000984	210100			4496	LODI HL,1	1644.
000987	19			4498	ADD HL,DE	1645.
000988	545D			4500	LOD DE,HL	1646.
00098A	3A1E21			4502	LD A,PATCNT	1647.
00098D	3D			4504	DEC A	1648.
00098E	321E21			4506	ST A,PATCNT MODIFY COUNT	1649.
000991	C27309			4508	JMP NZ,LOADING ANC CONTINUE IF NOT DONE	1650.

LOC	OBJECT CODE	ADDR1	ADDR2	STMT	SOURCE STATEMENT	ASM H V 05 01.23 10/02/76
000994	CDD209			4510	CALL BYTEIN	GET CHECK BYTE 1651.
000997	47			4512	LOD B,A	1652.
000998	3A1D21			4514	LD A,PATCKSM	1653.
00099B	80			4516	ADD B	1654.
00099C	CAA909			4518	JMP Z,LOADOK	QUIT IF CHECKSUM IS ZERO 1655.
00099F	3A1C21			4520	LD A,PATSTAT	IT AINT SO MODIFY STATUS 1656.
0009A2	F610			4522	ORAI CKSMERR	1657.
0009A4	D386			4524	OUT BELL	RING BELL 1658.
0009A6	321C21			4526	ST A,PATSTAT	1659.
0009A9	161B			4528	LOADOK LODI D,@ESC	DISPLAY ESCAPE AFTER PATCH 1660.
0009AB	79			4530	LOADEND LOD A,C	1661.
0009AC	21FAFF			4532	RESUME L0ADER,CP	1662.
0009AF	39			4534		
0009B0	222420			4535		
0009B3	2E27			4536		
0009B5	CD0C06			4537		
0009B8	3E00			4538	CCBSET GETCHR,CP=DISPL	WE IS DONE SO REMOVE THE PATCHER 1663.
0009BA	326F20			4540		
0009BD	3E6C			4541	CCBSET DISPL,CS=GETCHR	FROM THE PROCESSING CHAIN 1664.
0009BF	320220			4543		
0009C2	CD420A			4544	CALL UNDOSEMI	UNDO SEMILOCAL MODE 1665.
0009C5	7A			4546	LOD A,D	DISPLAY A LEFT ARROW 1666.
0009C6	21FAFF			4548	RESUME LOADER,CP	AND LEAVE 1667.
0009C9	39			4550		
0009CA	222420			4551		
0009CD	2E27			4552		
0009CF	CD0C06			4553		
0009D2	79			4554	BYTEIN LOD A,C	GET BACK CHAR 1668.
0009D3	21FAFF			4556	RESUME LOADER,CP	PROCESS IT 1669.
0009D6	39			4558		
0009D7	222420			4559		
0009DA	2E27			4560		
0009DC	CD0C06			4561		
0009DF	21FAFF			4562	RESUME LOADER,CS	AND GET NEXT ONE 1670.
0009E2	39			4564		
0009E3	222420			4565		
0009E6	2E26			4566		
0009E8	CD0C06			4567		
0009EB	4F			4568	LOD C,A	SAVE IT 1671.
0009EC	CD1406			4570	CALL HEXIN	CONVERT TO HEX 1672.
0009EF	DA190A			4572	JMP C,ABEND	IF CARRY IS SET ITS NOT HEX 1673.
0009F2	07070707			4574	ROT L,4	MOVE IT OVER 1674.
0009F6	47			4576	LOD B,A	AND SAVE FIRST DIGIT 1675.
0009F7	79			4578	LOD A,C	RETRIEVE CHAR 1676.
0009F8	21FAFF			4580	RESUME LOADER,CP	AND PROCESS 1677.
0009FB	39			4582		
0009FC	222420			4583		
0009FF	2E27			4584		
000A01	CD0C06			4585		
000A04	21FAFF			4586	RESUME LOADER,CS	CONTINUE 1678.
000A07	39			4588		
000A08	222420			4589		
000A0B	2E26			4590		
000A0D	CD0C06			4591		
000A10	4F			4592	LOD C,A	SAVE NEXT CHAR 1679.

000A11 CD1406
000A14 DA190A
000A17 80

4594
4596
4598
4600

CALL HEXIN
JMP C. ABEND
ADD B
RET

RETURN IF NOT HEX
NOW WE HAVE THE FULL BYTE

1680.
1681.
1682.
1683.

LOC	OBJECT CODE	ADDR1	ADDR2	STMT	SOURCE STATEMENT	ASM H V 05 01.23 10/02/76
000A18	C9			4601		
000A19	E1			4602	ABEND POP HL	POP RETURN ADDRESS 1684.
000A1A	3A1C21			4604	NOGOOD LD A,PATSTAT	NOT HEX CHAR, MODIFY STATUS 1685.
000A1D	F640			4606		
000A1F	321C21			4608	ORAI NOTHEX	1686.
000A22	D386			4610	ST A,PATSTAT	1687.
000A24	51			4612	OUT BELL	AND SOUND HORN 1688.
000A25	0E1B			4614	LOD D,C	DISPLAY CHAR AFTER ESCAPE 1689.
000A27	C3AB09			4616	LODI C,@ESC	1690.
000A2A	3A1C21			4618	JMP LOADEND	TERMINATE PATCHING 1691.
000A2D	F608			4620	NOSTICK LD A,PATSTAT	BYTE DID NOT STAY 1692.
000A2F	321C21			4622	ORAI PATFAIL	1693.
000A32	D386			4624	ST A,PATSTAT	1694.
000A34	C9			4626	OUT BELL	INFORM USER 1695.
000A35	3A1C21			4628	RET U	AND RETURN 1696.
000A38	F620			4630	BADTYP LD A,PATSTAT	TYPE NOT ZERO 1697.
000A3A	321C21			4632	ORAI TYPERR	1698.
000A3D	D386			4634	ST A,PATSTAT	1699.
000A3F	C3A909			4636	OUT BELL	1700.
000A42	3A2421			4638	JMP LOADOK	1701.
000A45	E604			4640	UNDOSEMI LD A,LCLMODE	THIS SHIFTS OUT OF SEMILOCAL MODE 1702.
000A47	3E06			4642	ANDI NORCV	SEE IF REAL LOCAL MODE SET 1703.
000A49	C24D0A			4644	LODI A,NORCV+NOSEND	1704.
000A4C	97			4646	JMP NZ,UNDOIT	IF SO STORE THOSE FLAGS BACK 1705.
000A4D	322421			4648	SUB A	IF NOT JUST CLEAR IT 1706.
000A50	C9			4650	UNDOIT ST A,LCLMODE	SO STORE WHATEVER IT IS. 1707.
					RET U	AND RETURN TO PAPA 1708.

LOC	OBJECT CODE	ADDR1	ADDR2	STMT	SOURCE STATEMENT	ASM H V 05 01.23 10/02/76
				4653	*****	1710.
				4654	*	1711.
				4655	* HEXER--- A COROUTINE TO PROCESS A HEX LOCATION	1712.
				4656	*	1713.
				4657	*****	1714.
000A51	010022			4658	HEXEP LODI BC,X'2200' INITIALIZE BC	1715.
000A54	57			4660	HEXLOOP LOD D,A SAVE THE CHAR	1716.
000A55	6069			4662	LOD HL,BC NOW SHIFT FOUR BITS LEFT	1717.
000A57	CD6E06			4664	CALL HEXSTR CALL GLOBAL ROUTINE	1718.
000A5A	7A			4666	LOD A,D GET BACK CHAR	1719.
000A5B	DA7B0A			4668	JMP C,READY ITS NOT HEX SO DO JUMP	1720.
000A5E	444D			4670	LOD BC,HL SAVE ADDRESS	1721.
000A60	21FAFF			4672	RESUME HEXER,CP AND DISPLAY IT	1722.
000A63	39			4674		
000A64	224820			4675		
000A67	2E4B			4676		
000A69	CD0C06			4677		
000A6C	21FAFF			4678	RESUME HEXER,CS GET NEXT ONE	1723.
000A6F	39			4680		
000A70	224820			4681		
000A73	2E4A			4682		
000A75	CD0C06			4683		
000A78	C3540A			4684	JMP HEXLOOP AND LOOP	1724.
000A7B	222921	00A7B		4686	READY EQU * WE HAVE THE WHOLE ADDRESS NOW	1725.
000A7E	21FAFF			4688	ST HL,HEXADDR SO STORE IT	1726.
000A81	39			4690	RESUME HEXER,CP	1727.
000A82	224820			4692		
000A85	2E4B			4693		
000A87	CD0C06			4694		
000A8A	7A			4695		
000A8B	FE73			4696	LOD A,D GET THE CHAR	1728.
000A8D	CAB50A			4698	CMPI 'S' IS IT A DISPLAY COMMAND	1729.
000A90	FE70			4700	JMP Z,GETNUM YES GO GET ANOTHER CHAR	1730.
000A92	CADC0A			4702	CMPI 'P' NO SEE IF PATCH COMMAND	1731.
000A95	FE6A			4704	JMP Z,SHOWIT YES GO DO THAT	1732.
000A97	CA3C0B			4706	CMPI 'J' SEE IF WE JUMP THERE.	1733.
000A9A	D386			4708	JMP Z,GCTHERE	1734.
000A9C	CD420A			4710	HEXABEND OUT BELL EXIT FROM HEXER AND RING BELL	1735.
000A9F	3E6C	00A9F		4712	HEXEXIT CALL UNDOSEMI UNDO SEMILOCAL MODE	1736.
000AA1	320220			4714	UNDUN CCBSET DISPL,CS=GETCHR COME HERE IF SEMILOCAL IS UNDUN	1737.
000AA4	3E00			4716		
000AA6	326F20			4717		
000AA9	21FAFF			4718	CCBSET GETCHR,CP=DISPL	1738.
000AAC	39			4720		
000AAD	224820			4721	RESUME HEXER,CS	1739.
000AB0	2E4A			4723		
000AB2	CD0C06			4724		
000AB5	21FAFF			4725		
000AB8	39			4726		
000AB9	224820			4727	GETNUM RESUME HEXER,CS GET ANOTHER CHAR	1740.
000ABC	2E4A			4729		
000ABE	CD0C06			4730		
000AC1	57			4731		
				4732		
				4733	LOD D,A SAVE IT AWAY	1741.

000AC2 21FAFF
000AC5 39
000AC6 224820
000AC9 2E4B
000ACB CD0C06
000ACE 7A
000ACF CD1406

4735
4737
4738
4739
4740
4741
4743

RESUME HEXER.CP

LDD A,D
CALL HEXIN

AND DISPLAY IT

GET CHAR BACK
MAKE IT INTO HEX DIGIT

1742.

1743.
1744.

LCC	OBJECT CODE	ADDR1	ADDR2	STMT	SOURCE	STATEMENT	ASM H V 05 01.23	10/02/76
000AD2	DA9A0A			4745		JMP C,HEXABEND	IT ISN'T ONE, GO COMPLAIN	1745.
000AD5	B7			4747		ORA A	SEE IF IT'S ZERO	1746.
000AD6	C2DB0A			4749		JMP NZ,NOTZERO	NO GO START DISPLAY	1747.
000AD9	3E10			4751		LODI A,16	IT IS, MAKE IT 16	1748.
000ADB	57			4753	NOTZERO	LOD D,A	SAVE THIS NUMBER	1749.
000ADC	CD1C0B			4755	SHOWIT	CALL HEXCRLF	PUT OUT A CRLF	1750.
000ADF	78			4757		LOD A,B	DISPLAY THE ADDRESS	1751.
000AE0	CD8C0B			4759		CALL HEXDISP	ONE BYTE AT A TIME	1752.
000AE3	79			4761		LOD A,C	GET THE OTHER HALF	1753.
000AE4	CD8C0B			4763		CALL HEXDISP	DISPLAY THAT TOO	1754.
000AE7	CD850B			4765		CALL SPACE	LEAVE SOME ROOM FOR VISIBILITY	1755.
000AEA	1E10			4767		LODI E,16	AND SET UP DISPLAY LOOP	1756.
000AEC	3E20			4769	DISLOOP	LODI A,' '	1 SPACE BETWEEN BYTES	1757.
000AEE	21FAFF			4771		RESUME HEXER,CP	SHOW THAT	1758.
000AF1	39			4773				
000AF2	224820			4774				
000AF5	2E4B			4775				
000AF7	CD0C06			4776				
000AFA	0A			4777		LD A,(BC)	GET THE BYTE	1759.
000AFB	CD8C0B			4779		CALL HEXDISP	DISPLAY IT	1760.
000AFE	3E70			4781		LODI A,'P'	SEE IF WE'RE PATCHING	1761.
000B00	BA			4783		CMP D	{D HAS 'P' IF WE ARE}	1762.
000B01	CC480B			4785		CALL Z,PATCHIT	IF SO GO DO PATCHING	1763.
000B04	DA160B			4787		JMP C,HEXDONE	IF CARRY GOT SET WE END DISPLAY	1764.
000B07	03			4789		INC BC	INCREMENT ADDRESS	1765.
000B08	1D			4791		DEC E	DECREMENT COUNT	1766.
000B09	C2EC0A			4793		JMP NZ,DISLOOP	AND CONTINUE	1767.
000B0C	3E70			4795		LODI A,'P'	IF PATCHING WE DO ANOTHER LINE	1768.
000B0E	BA			4797		CMP D		1769.
000B0F	CADC0A			4799		JMP Z,SHOWIT		1770.
000B12	15			4801		DEC D	IF NOT DECREMENT LINE COUNT	1771.
000B13	C2DC0A			4803		JMP NZ,SHOWIT	AND DO ANOTHER ANY WAY	1772.
000B16	CD1C0B			4805	HEXDONE	CALL HEXCRLF	COUNT WAS ZERO, WE END HEXIT	1773.
000B19	C39C0A			4807		JMP HEXEXIT		1774.
000B1C	3E0D			4809	HEXCRLF	LODI A,@CR	DISPLAY A CR	1775.
000B1E	320521			4811		ST A,PREVCHAR		1776.
000B21	21FAFF			4813		RESUME HEXER,CP		1777.
000B24	39			4815				
000B25	224820			4816				
000B28	2E4B			4817				
000B2A	CD0C06			4818				
000B2D	3E0A			4819		LODI A,@LF	AND A LF	1778.
000B2F	21FAFF			4821		RESUME HEXER,CP		1779.
000B32	39			4823				
000B33	224820			4824				
000B36	2E4B			4825				
000B38	CD0C06			4826				
000B3B	C9			4827		RET ,	RETURN	1780.
000B3C	CD420A			4829	GO THERE	CALL UNDOSEMI	COME HERE IF 'J' WAS TYPED	1781.
000B3F	2A2921			4831		CALL@ HEXADDR	GO TO THAT ADDRESS	1782.
000B42	CD0B06			4833				
				4834		JMP UNDUN	SEMILOCAL ALREADY UNDONE, GO HOME	1783.

LOC	OBJECT CODE	ADDR1	ADDR2	STMT	SOURCE STATEMENT	ASM H V 05 01.23 10/02/76
000B45	C39F0A			4835		
000B48	D5			4836	PATCHIT PUSH DE	THIS IS THE PATCHING ROUTINE 1784.
000B49	3E88			4838	LODI A,@BS+X*80'	FIRST WE MOVE THE CURSOR BACK 1785.
000B4B	1E02			4840	LODI E,2	TWO SPACES 1786.
000B4D	CD00C			4842	CALL PUTREP	BY CALLING PUTREP 1787.
000B50	21FAFF			4844	RESUME HEXER,CS	THEN WE GET A CHARACTER 1788.
000B53	39			4846		
000B54	224820			4847		
000B57	2E4A			4848		
000B59	CD0C06			4849		
000B5C	CDB60B			4850	CALL HEXERCHK	SEE IF IT IS HEX 1789.
000B5F	DA800B			4852	JMP C,BLOOEY	IT WAS NOT, GO COMPLAIN 1790.
000B62	07070707			4854	ROT L,4	IT WAS , SAVE IT AWAY 1791.
000B66	57			4856	LOD D,A	IN D 1792.
000B67	21FAFF			4858	RESUME HEXER,CS	AND GET ANOTHER ONE 1793.
000B6A	39			4860		
000B6B	224820			4861		
000B6E	2E4A			4862		
000B70	CD0C06			4863		
000B73	CDB60B			4864	CALL HEXERCHK	DO THE SAME RIGAMAROLE 1794.
000B76	DA800B			4866	JMP C,BLOOEY	AS BEFORE 1795.
000B79	82			4868	ADD D	AND ADD IN THE OTHER DIGIT 1796.
000B7A	D1			4870	POP DE	GET BACK THE OLD DE 1797.
000B7B	02			4872	ST A,(BC)	STORE THE NEW VALUE 1798.
000B7C	0A			4874	LD A,(BC)	AND GET BACK WHATEVER IS THERE 1799.
000B7D	C38C0B			4876	JMP HEXDISP	GO DISPLAY THAT 1800.
000B80	D386			4878	BLOOEY OUT BELL	COMPLAINT DEPT. 1801.
000B82	D1			4880	POP DE	WE MUST POP DE 1802.
000B83	37			4882	STC ,	SET THE CARRY 1803.
000B84	C9			4884	RET ,	AN RETURN 1804.
000B85	1E04			4886	SPACE LODI E,4	DISPLAY 4 SPACES 1805.
000B87	3E20			4888	LODI A,' '	
000B89	C300C			4890	JMP PUTREP	
000B8C	F5			4892	HEXDISP PUSH AF	SAVE THE CHAR 1808.
000B8D	E6F0			4894	ANDI X'F0'	GET FIRST DIGIT 1809.
000B8F	07070707			4896	ROT L,4	
000B93	CD7906			4898	CALL HEXOUT	CONVERT TO DISPLAY CODE 1811.
000B96	21FAFF			4900	RESUME HEXER,CP	
000B99	39			4902		
000B9A	224820			4903		
000B9D	2E4B			4904		
000B9F	CD0C06			4905		
000BA2	F1			4906	POP AF	GET CHAR AGAIN 1813.
000BA3	E60F			4908	ANDI X'0F'	SECOND DIGIT 1814.
000BA5	CD7906			4910	CALL HEXOUT	
000BA8	21FAFF			4912	RESUME HEXER,CP	AND SHOW IT 1815.
000BAB	39			4914		
000BAC	224820			4915		
000BAF	2E4B			4916		
000BB1	CD0C06			4917		
000BB4	B7			4918	ORA A	RESET CARRY FOR RETURN TO DISLOOP 1817.
000BB5	C9			4920	RET ,	
000BB6	FE20			4922	HEXERCHK CMPI ' '	PATCH CHAR PROCESSOR.. WAS IT SPACE 1819.
000BB8	CACF0B			4924	JMP Z,ONESPACE	GO DO SPACE ITF SUCH 1820.
000BBB	D21406			4926	JMP NC,HEXIN	IF CARRY SET IT'S NOT CTRL CHAR 1821.

000BBE FE08

4928

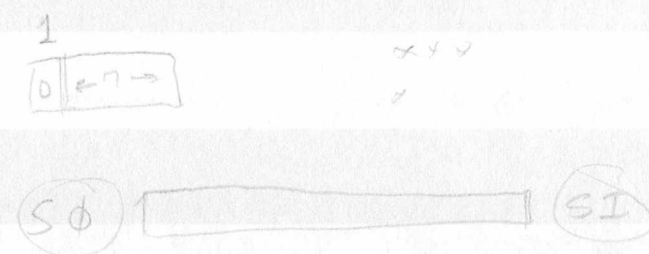
CMPI @BS

IT WARN'T NO SPACE SEE IF BS

1822.

LOC	OBJECT CODE	ADDR1	ADDR2	STMT	SOURCE STATEMENT	ASM H V 05 01.23 10/02/76
000BC0	CAE20B			4930	JMP Z,EACKUP	YES GO DO IT. 1823.
000BC3	FE0A			4932	CMPI @LF	SEE IF LINE FEED ETC... 1824.
000BC5	CA150C			4934	JMP Z,DOWNLINE	1825.
000BC8	FE0B			4936	CMPI @VT	1826.
000BCA	CA290C			4938	JMP Z,UPLINE	1827.
000BCD	37			4940	STC ,	NOT VALID CHAR 1828.
000BCE	C9			4942	RET ,	1829.
000BCF	E1			4944	ONESPACE POP HL	IT'S A SPACE POP RETURN ADDRESS 1830.
000BD0	1E02			4946	LODI E,2	PUT OUT A COUPLE SPACES 1831.
000BD2	F680			4948	ORAI X'80'	(COMMAND SPACES) 1832.
000BD4	CD000C			4950	CALL PUTREP	1833.
000BD7	D1			4952	POP DE	POP THE OLD DE 1834.
000BD8	7B			4954	LOD A,E	AND SEE IF WE WRAP AROUND 1835.
000BD9	FE01			4956	CMPI 1	1836.
000BDB	C0			4958	RET NZ	NO WE DON'T SO RETURN TO DISLOOP 1837.
000BDC	E1			4960	POP HL	YES SO POP ANOTHER RETURN ADDRESS 1838.
000BDD	03			4962	INC BC	INCREMENT BC "BY HAND" 1839.
000BDE	C3DC0A			4964	JMP SHOWIT	AND START ALL OVER 1840.
000BE1	C9			4966	RET ,	GO TELL USER HE BLEW IT 1841.
000BE2	1E04			4968	BACKUP LODI E,4	DO SOME BACKSPACES 1842.
000BE4	F680			4970	ORAI X'80'	1843.
000BE6	CD000C			4972	CALL PUTREP	1844.
000BE9	E1			4974	POP HL	POP THE RETURN ADDRESS 1845.
000BEA	D1			4976	POP DE	AND THE OLD DE 1846.
000BEB	1C			4978	INC E	INCREMENT COUNT 1847.
000BEC	1C			4980	INC E	TWICE SINCE IT GETS DECREMENTED 1848.
000BED	3E12			4982	LODI A,18	CHECK FOR WRAPAROUND 1849.
000BEF	BB			4984	CMP E	1850.
000BF0	C2FC0B			4986	JMP NZ,NOTOOFAR	NO WRAP 1851.
000BF3	1E21			4988	LODI E,33	WE DID SO PUT OUT A BUNCH OF BS'S 1852.
000BF5	3E88			4990	LODI A,@BS+X'80'	1853.
000BF7	CD000C			4992	CALL PUTREP	1854.
000BFA	1E02			4994	LODI E,2	AND SETUP COUNT FOR END OF LINE 1855.
000BFC	0B			4996	NOTOOFAR DEC BC	NOW DEC ADDRESS 1856.
000BFD	0B			4998	DEC BC	TWICE SINCE IT'S INCREMENTED ALSO 1857.
000BFE	B7			5000	ORA A	RESET CARRY FLAG 1858.
000BFF	C9			5002	RET ,	AND RETURN 1859.
000C00	D5			5004	PUTREP PUSH DE	SAVE DE 1860.
000C01	57			5006	LOD D,A	SAVE THE CHAR WHICH WE REPEAT 1861.
000C02	7A			5008	PUTREP1 LOD A,D	GET IT 1862.
000C03	21FAFF			5010	RESUME HEXER,CP	SHOW IT 1863.
000C06	39			5012		
000C07	224820			5013		
000C0A	2E4B			5014		
000C0C	CD0C06			5015		
000C0F	1D			5016	DEC E	DECREMENT COUNT 1864.
000C10	C2020C			5018	JMP NZ,PUTREP1	AND KEEP GOINT 1865.
000C13	D1			5020	POP DE	POP OLD DE 1866.
000C14	C9			5022	RET ,	AND GO HOME 1867.
000C15	1E01			5024	DOWNLINE LODI E,1	GO DOWN A LINE, PUT OUT LF 1868.
000C17	CD000C			5026	CALL PUTREP	1869.
000C1A	3E88			5028	LODI A,@BS+X'80'	AND A BS 1870.
000C1C	CD000C			5030	CALL PUTREP	1871.

LOC	OBJECT CODE	ADDR1	ADDR2	STMT	SOURCE STATEMENT	ASM H V 05 01.23 10/02/76
000C1F	E1			5032	POP HL	POP RETURN ADDRESS 1872.
000C20	D1			5034	POP DE	AND OLD DE 1873.
000C21	1C			5036	INC E	MODIFY COUNT 1874.
000C22	210F00			5038	LODI HL,15	AND ADDRESS 1875.
000C25	09			5040	ADD HL,BC	1876.
000C26	444D			5042	LOD BC,HL	ASAVE IT BACK 1877.
000C28	C9			5044	RET	1878.
000C29	1E01			5046	LODI E,1	GO UP A LINE (JUST LIKE ABOVE) 1879.
000C2B	CD000C			5048	CALL PUTREP	1880.
000C2E	3E88			5050	LODI A,@ES+X*80'	1881.
000C30	CD000C			5052	CALL PUTREP	1882.
000C33	E1			5054	POP HL	1883.
000C34	D1			5056	POP DE	1884.
000C35	1C			5058	INC E	1885.
000C36	21EFFF			5060	LODI HL,-17	1886.
000C39	09			5062	ADD HL,BC	1887.
000C3A	444D			5064	LOD BC,HL	1888.
000C3C	3F			5066	CMC	1889.
000C3D	C9			5068	RET	1890.



LOC	OBJECT CODE	ADDR1	ADDR2	STMT	SOURCE STATEMENT	ASM H V 05 01.23 10/02/76
				5071	*****	1892.
				5072	*	1893.
				5073	* THE DISPLAY COROUTINE...	1894.
				5074	*	1895.
				5075	*****	1896.
		00C3E		5076	TXTMOD EQU *	1897.
000C3E	210421			5078	LODI HL,CURSCAR	1898.
000C41	56			5080	LOD D,M	1899.
000C42	2A0021			5082	LD HL,CURSLOC	1900.
000C45	72			5084	LOD M,D	1901.
000C46	E5			5086	PUSH HL	1902.
000C47	B7			5088	IOR A	1903.
000C48	F26B0C			5090	JMP NS,TXTCTL	1904.
000C4B	21510C			5092	LODI HL,CMDTAB2	1905.
000C4E	CD3E06			5094	CALL SEARCH	1906.
000C51	E3			5096	XCH HL,(SP)	1907.
000C52	C9			5098	RET U	1908.
		00C51		5100	CMDEF 8C,GOGRAPH	1909.
000C53	8C0F28			5102		
000C56	C70C59			5103	CMDEF C7,GODRAG	1910.
				5105	EQU VGT+X'IFFD'	1911.
	IEV044	***	ERROR ***	UNDEFINED	SYMBOL	
000C59	880DB2			5107	CMDEF 88,KEY@B SND	1912.
000C5C	8D0D81			5109	CMDEF 8D,KEY@CR	1913.
000C5F	8A0DA7			5111	CMDEF 8A,KEY@LF	1914.
000C62	8B0DFC			5113	CMDEF 8B,KEY@VT	1915.
000C65	A00CAF			5115	CMDEF A0,CURMOTSP	1916.
000C68	000F80			5117	CMDEF 00,TXTEND	1917.
000C6B	FE0D			5119	TXTCTL CMPI @CR	1918.
000C6D	CA800D			5121	JMP Z,KEY@CR2	1919.
000C70	57			5123	LOD D,A	1920.
000C71	3A3021			5125	LD A,CRLF0VR	1921.
000C74	47			5127	LOD B,A	1922.
000C75	97			5129	SUB A	1923.
000C76	323021			5131	ST A,CRLF0VR	1924.
000C79	7A			5133	LOD A,D	1925.
000C7A	FE20			5135	CMPI X'20'	1926.
000C7C	F2A90C			5137	JMP NS,TXTCHAR	1927.
000C7F	21850C			5139	LODI HL,CMDTAB3	1928.
000C82	CD3E06			5141	CALL SEARCH	1929.
000C85	E3			5143	XCH HL,(SP)	1930.
000C86	C9			5145	RET U	1931.
		00C85		5147	CMDEF 1D,GOGRAPHC	1932.
000C87	1D0F17			5149		
000C8A	0A0D93			5150	CMDEF 0A,KEY@LF2	1933.
000C8D	1C0E03			5152	CMDEF 1C,SIM4023	1934.
000C90	080DAE			5154	CMDEF 08,KEY@BS	1935.
000C93	0B0DFC			5156	CMDEF 0B,KEY@VT	1936.
000C96	090D6B			5158	CMDEF 09,KEYHTAB	1937.
000C99	140EF7			5160	CMDEF 14,KEYHYPER	1938.
000C9C	1F0F78			5162	CMDEF 1F,SETNECHD	1939.
000C9F	110F80			5164	CMDEF 11,TXTEND	1940.
000CA2	070F12			5166	CMDEF 07,DOBELL	1941.
000CA5	000CA8			5168	CMDEF 00,TXTCHAR2	1942.

RESTORE CHAR AT CURSOR LOCATION
(MUST LEAVE CURSLOC IN HL)

SEE IF IT'S A CR
IF SO DO NOT RESET CRLF0VR
SAVE THE CHAR
GET FLAG
AND REMEMBER IT
CLEAR IT

GET CHAR AGAIN

ENTER GRAPH MODE?

ALIGNS CURSOR TO COL 41 OF LINE
(CNTL SHIFT 0)
DO NOT DISPLAY @DC1
RING BELL
NOT FOUND

LOC	OBJECT CODE	ADDR1	ADDR2	STMT	SOURCE	STATEMENT	ASM H V 05 01.23 10/02/76
				5171 *			1944.
				5172 *			1945.
				5173 *			1946.
				5174	TXTCHAR2	PUSH HL	1947.
000CA8	E5			5176	TXTCHAR	LODI HL,SHIFTMD	1948.
000CA9	213D21			5178		IOR M	1949.
000CAC	B6			5180		PCP HL	1950.
000CAD	E1			5182		LOD M,A	1951.
000CAE	77			5184	CURMOTSP	INC HL	1952.
000CAF	23			5186		LD A,CURSX	1953.
000CB0	3A0221			5188		INC A	1954.
000CB3	3C			5190		ST A,CURSX	1955.
000CB4	320221			5192		CMP I L,INESIZE	1956.
000CB7	FE51			5194		JMP S,KEYCROK	1957.
000CB9	FA280D			5196	TABWRAP2	EQU *	1958.
		00CBC		5198		SUB A	1959.
000CBC	97			5200		ST A,CURSX	1960.
000CBD	320221			5202		DEC A	1961.
000CC0	3D			5204		ST A,CRLF0VR	1962.
000CC1	323021			5206		ST A,PREVCHAR	1963.
000CC4	320521			5208	KEYINCY	LOD A,H	1964.
000CC7	7C			5210		ORA A	1965.
000CC8	B7			5212		JMP NZ,NOUPDATE	1966.
000CC9	C2D00C			5214		LD A,TEXT+1	1967.
000CCC	3A6821			5216		LOD H,A	1968.
000CCF	67			5218	NOUPDATE	ST HL,CURSLOC	1969.
000CD0	220021			5220		CALL SUBWRP81	1970.
000CD3	CD5B07			5222		LOD BC,HL	1971.
000CD6	444D			5224		LD HL,TEXTBOT	1972.
000CD8	2A6D21			5226		LODI DE,LINESIZE	1973.
000CDB	115100			5228		CALL CIRCLEFN	1974.
000CDE	CD7C07			5230		JMP NC,NOMOVE	1975.
000CE1	D20C0D			5232		LD HL,TEXTBOT	1976.
000CE4	2A6D21			5234		CALL NEGADDWR	1977.
000CE7	CD7307			5236		ST HL,TEXTBOT	1978.
000CEA	226D21			5238		LD HL,TEXTTOP	1979.
000CED	2A6B21			5240		PUSH BC	1980.
000CF0	C5			5242		LOD BC,HL	1981.
000CF1	444D			5244		LD HL,NXTDISA	1982.
000CF3	2A0C21			5246		ADD HL,BC	1983.
000CF6	09			5248		LODI DE,X'FFFF'	1984.
000CF7	11FFFF			5250		ADD HL,DE	1985.
000CFA	19			5252		CALL NC,SCRUPDO	1986.
000CFB	D43204			5254		LOD HL,BC	1987.
000CFE	6069			5256		POP BC	1988.
000D00	C1			5258		CALL NEGADDWR	1989.
000D01	CD7307			5260		ST HL,TEXTTOP	1990.
000D04	226B21						

LOC	OBJECT CODE	ADDR1	ADDR2	STMT	SOURCE	STATEMENT	ASM H V 05 01.23 10/02/76
000D07	3EFF			5262		LODI A,X'FF'	SET FLAG TO BLANK LINE 1991.
000D09	320521			5264		ST A,PREVCHAR	1992.
000D0C	2A0C21			5266	NOMOVE	LD HL,NXTDISA	GET NXTDISA 1993.
000D0F	11B50B			5268		LODI DE,SCRNSIZE	SEE IF WE MUST SCROLL UP 1994.
000D12	CD5307			5270		CALL ADDWRP	DDDDDDDDDDDDDDDDDD 1995.
000D15	CD6B07			5272		CALL MINUSHL	TO KEEP UP WITH THE CURSOR 1996.
000D18	115100			5274		LODI DE,LINESIZE	1997.
000D1B	CD7C07			5276		CALL CIRCLEFN	CALL ROUTINE 1998.
000D1E	DC2E04			5278		CALL C,SCRUP	AND DO IT IF NECESSARY 1999.
000D21	6069			5280		LOD HL,BC	GET THE MODIFIED CURSLOC 2000.
000D23	CD5007			5282		CALL ADDWRP81	AND RESTORE IT TO THE RIGHT VALUE 2001.
000D26	444D			5284		LOD BC,HL	SAVE IN BC FOR LATER USE 2002.
000D28	220021			5286	KEYCUROK	ST HL,CURSLOC	SAVE AS THE NEW CURSLOC 2003.
000D2B	3A0521			5288		LD A,PREVCHAR	SEE IF WE BLANK A LINE 2004.
000D2E	3C			5290		INC A	2005.
000D2F	C2560D			5292		JMP NZ,KEYCURST	NO SO GO STORE CURSCHAR 2006.
000D32	1600			5294		LODI D,0	YES WE DO GET READY 2007.
000D34	3A0221			5296		LD A,CURSX	2008.
000D37	D651			5298		SUBI LINESIZE	START BLANKING AT CURSLOC+(81-XOFF) 2009.
000D39	2F			5300		CMA	2010.
000D3A	3C			5302		INC A	(WE NOW HAVE THE RIGHT NUMBER) 2011.
000D3B	5F			5304		LOD E,A	NOW ADD THIS ON TO CURSLOC 2012.
000D3C	CD5307			5306		CALL ADDWRP	2013.
000D3F	1E01			5308		LODI E,1	USE DE TO INCREMENT HL 2014.
000D41	3E51			5310		LODI A,LINESIZE	USE A FOR COUNTER 2015.
000D43	3620			5312	KEYCRLF	LODI M,' '	BLANK IT 2016.
000D45	19			5314		ADD HL,DE	MOVE POINTER 2017.
000D46	D24C0D			5316		JMP NC,NOOVER	WE DIDN'T WRAP? 2018.
000D49	2A6721			5318		LD HL,TEXT	WE DID START AT TEXT 2019.
000D4C	3D			5320	NOOVER	DEC A	DECREMENT COUNT 2020.
000D4D	C2430D			5322		JMP NZ,KEYCRLF	AND KEEP GOING 2021.
000D50	97			5324		SUB A	CLEAR OUT PREVCHAR 2022.
000D51	320521			5326		ST A,PREVCHAR	2023.
000D54	6069			5328		LOD HL,BC	GET CURSLOC BACK 2024.
000D56	7E			5330	KEYCURST	LOD A,M	SAV CHAR FM WHERE CURSOR WILL BE PUT 2025.
000D57	320421			5332		ST A,CURSCHAR	2026.
000D5A	367F			5334		LODI M,@DEL	START CURSUR AS BLOB 2027.
000D5C	FE7F			5336		CMPI @DEL	UNLESS CHAR IS ALSO BLOB 2028.
000D5E	C2630D			5338		JMP NZ,KEYCURTM	2029.
000D61	3620			5340		LODI M,' '	IN WHICH CASE CURSOR STARTS AS BLOB 2030.
000D63	3E08			5342	KEYCURTM	LODI A,CURSTIME	RESTART ITS COUNTER 2031.
000D65	320321			5344		ST A,CURSCTR	2032.
000D68	C3800F			5346		JMP TXTEND	2033.

LOC	OBJECT CODE	ADDR1	ADDR2	STMT	SOURCE STATEMENT	ASM H V 05 01.23 10/02/76
				5349 *		2035.
				5350 *		2036.
		00D6B		5351 KEYHTAB EQU *	HORIZONTAL TAB	2037.
				5353 *		2038.
000D6B	CDA00E			5354	CALL HTAB GO TAB	2039.
000D6E	1600			5356	LODI D,0	2040.
000D70	2A0021			5358	LD HL,CURSLOC	2041.
000D73	19			5360	ADD HL,DE	2042.
000D74	3D			5362	DEC A	2043.
000D75	CABC0C			5364	JMP Z,TABWRAP2	2044.
000D78	FA280D			5366	JMP S,KEYCUROK	2045.
000D7B	3620			5368	LODI M,' '	2046.
000D7D	C3AF0C			5370	JMP CURMOTSP	2047.
				5372 *		2048.
				5373 *	CARRIAGE RETURN	2049.
				5374 *		2050.
000D80	E1			5375 KEY@CR2 POP HL	GET CURSLOC	2051.
		00D81		5377 KEY@CR EQU *		2052.
000D81	547D			5379	LOD DA,HL	2053.
000D83	210221			5381	LODI HL,CURSX	2054.
000D86	96			5383	SUB M	2055.
000D87	6F			5385	LOD L,A	2056.
000D88	7A			5387	LOD A,D	2057.
000D89	DE00			5389	SBBI 0	2058.
000D8B	67			5391	LOD H,A	2059.
000D8C	97			5393	SUB A	2060.
000D8D	320221			5395	ST A,CURSX	2061.
000D90	C3280D			5397	JMP KEYCUROK	2062.
				5399 *		2063.
				5400 *	LINEFEED	2064.
				5401 *		2065.
000D93	545D			5402 KEY@LF2 LOD DE,HL	REMEMBER CURSLOC	2066.
000D95	210521			5404	LODI HL,PREVCHAR	2067.
000D98	7E			5406	LOD A,M	2068.
000D99	FE0D			5408	CMPI @CR	2069.
000D9B	C2A50D			5410	JMP NZ,NOTCRLF	2070.
000D9E	78			5412	LOD A,E	2071.
000D9F	B7			5414	IOR A	2072.
000DA0	FA800F			5416	JMP S,XTEND	2073.
000DA3	36FF			5418	LODI M,X'FF'	2074.
000DA5	626B			5420	LOD HL,DE	2075.
000DA7	115100			5422 KEY@LF	LODI DE,LINESIZE	2076.
000DAA	19			5424	ADD HL,DE	2077.
000DAB	C3C70C			5426	JMP KEYINCY	2078.

LOC	OBJECT CODE	ADDR1	ADDR2	STMT	SOURCE STATEMENT	ASM H V 05 01.23 10/02/76
				5429 *		2080.
				5430 *	BACKSPACE	2081.
				5431 *		2082.
000DAE	2B			5432	KEY@BS DEC HL BACK UP A CHARACTER	2083.
000DAF	3620			5434	LODI M,' ' *** ENTRY FOR DESTRUCTIVE BS	2084.
000DB1	23			5436	INC HL INC, SO NEXT DEC IS REGHT	2085.
				5438 *	(SAVES A JMP) NOTE FOLLOWING PROBLEM:	2086.
				5439 *	(IF AT FIRST CHAR IN DISPLAY MEMORY-	2087.
				5440 *	(THEN DEST. BACKSPACE DOES A STORE INTO	2088.
				5441 *	(THE WRITABLE CHARACTER GENERATOR.	2089.
000DE2	2B			5442	KEY@B SND DEC HL *** ENTRY FOR NON-DESTRUCTIVE BS	2090.
000DB3	3A0221			5444	LD A,CURSX X=X-1	2091.
000DB6	D601			5446	SUBI 1	2092.
000DB8	320221			5448	ST A,CURSX	2093.
000DBB	F2280D			5450	JMP NS,KEYCUROK NO X UNDFL	2094.
000DBE	3E50			5452	LODI A,LINESIZE-1	2095.
000DC0	320221			5454	ST A,CURSX X UNDFL: X=LINESIZE-1	2096.
000DC3	220021			5456	KEYDECY ST HL,CURSLOC STORE NEW CURSLOC	2097.
000DC6	444D			5458	LOD BC,HL SAVE IN BC FOR POSTERITY	2098.
000DC8	2A6921			5460	LD HL,MTEXT SEE IF IT IS BENEATH TEXT	2099.
000DCE	09			5462	ADD HL,BC	2100.
000DCC	DAD40D			5464	JMP C,NOUPDAT2 NO SO DON'T WORRY	2101.
000DCF	444D			5466	LOD BC,HL IT IS, WE GOTTA WRAP	2102.
000DD1	220021			5468	ST HL,CURSLOC NOW STORE THAT	2103.
000DD4	2A6B21			5470	NOUPDAT2 LD HL,TEXTTOP IN ANY CASE, SEE IF WE ARE AT TOP	2104.
000DD7	115100			5472	LODI DE,LINESIZE USING CLEVER JEZ ROUTINE	2105.
000DDA	CD7C07			5474	CALL CIRCLEFN	2106.
000DDD	D2E80D			5476	JMP NC,NOABORT NO, CONTINUE	2107.
000DE0	6069			5478	LOD HL,BC WE WERE SO ABORT ATTEMPT TO	2108.
000DE2	CD5007			5480	CALL ADDWRP81 MOVE PAST BOUNDARY	2109.
000DE5	C3280D			5482	JMP KEYCUROK AND GO TO NORMAL EXIT	2110.
000DE8	2A0C21			5484	NOABORT LD HL,NXTDISA WE DIDN'T ABORT, SEE IF WE SCROLL	2111.
000DEB	CD6B07			5486	CALL MINUSHL	2112.
000DEE	115100			5488	LODI DE,LINESIZE USE ROUTINE AGAIN	2113.
000DF1	CD7C07			5490	CALL CIRCLEFN	2114.
000DF4	DC1604			5492	CALL C,SCRDOWN	2115.
000DF7	6069			5494	LOD HL,BC GET CURSLOC BACK IN ANY CASE	2116.
000DF9	C3280D			5496	JMP KEYCUROK AND GO HOME	2117.
				5498 *		2118.
				5499 *	REVERSE LINEFEED	2119.
				5500 *		2120.
000DFC	11AFFF			5501	KEY@VT LODI DE,-LINESIZE LOC=LOC-LINESIZE	2121.
000DFF	19			5503	ADD HL,DE	2122.
000E00	C3C30D			5505	JMP KEYDECY GO DECREMENT Y	2123.
				5507 *		2124.
				5508 *		2125.
				5509 *		2126.
				5510 *		2127.
000E03	21FAFF	00E03		5511	SIM4023 EQU * 4023 CURSOR SIMULATOR	2128.
000E06	39			5513	RESUME DISPL,CS	2129.
000E07	220020			5516		
000E0A	2E02			5517		
000E0C	CD0C06			5518		
				5519	ADDI -32 GET RID OF GARBAGE	2130.

LOC	OBJECT CODE	ADDR1	ADDR2	STMT	SOURCE STATEMENT	ASM H V 05 01.23 10/02/76
000E0F	C6E0			5520		
000E11	320221			5521	ST A,CURSX	SAVE AS X POSITION 2131.
000E14	4F			5523	LOD C,A	SAVE THAT 2132.
000E15	0600			5525	LODI B,0	FOR DOUBLE ADD LATER 2133.
000E17	21FAFF			5527	RESUME DISPL,CS	GET THE NEXT CHARACTER 2134.
000E1A	39			5529		
000E1B	220020			5530		
000E1E	2E02			5531		
000E20	CD0C06			5532		
000E23	C6E0			5533	ADDI -32	GET RID OF GRABAGE 2135.
000E25	6F			5535	LOD L,A	SAVE THAT 2136.
000E26	2600			5537	LODI H,0	CLEAR OUT H 2137.
000E28				5539	MULT 81	MULTIPLY HL BY LINESIZE 2138.
000E28	545D			5541		
000E2A	29			5542		
000E2B	29			5543		
000E2C	29			5544		
000E2D	29			5545		
000E2E	EB			5546		
000E2F	19			5547		
000E30	EB			5548		
000E31	29			5549		
000E32	29			5550		
000E33	19			5551		
000E34	09			5552	ADD HL,BC	ADD IN CURSX 2139.
000E35	EB			5554	XCH HL,DE	SAVE IT IN DE 2140.
000E36	2A0021			5556	LD HL,CURSLOC	GET WHERE CURSOR IS A PRESENT 2141.
000E39	3A0421			5558	LD A,CURSHAR	GET CHARACTER WE DISPLACE 2142.
000E3C	77			5560	ST A,(HL)	AND PUT IT BACK 2143.
000E3D	2A0C21			5562	LD HL,NXTDISA	GET WHERE TOP OF SCREEN IS 2144.
000E40	CD5307			5564	CALL ADDWRP	ADD IN DISPLACEMENT 2145.
000E43	220021			5566	ST HL,CURSLOC	AND SAVE IT IN CURSOR PLACE 2146.
000E46	C3800F			5568	JMP TXTEND	2147.

LOC	OBJECT CODE	ADDR1	ADDR2	STMT	SOURCE STATEMENT	ASM H V 05 01.23 10/02/76
				5571	*****	2149.
				5572	* HORIZONTAL TAB SUPPORT	2150.
				5573	*****	2151.
				5574	*	2152.
				5575	*	2153.
		00E49		5576	RE SETTAB EQU *	2154.
				5578	*	2155.
000E49	3A1921			5579	LD A,FLAGS	2156.
000E4C	E6FE			5581	ANDI 255-TABREF	2157.
000E4E	321921			5583	ST A,FLAGS	2158.
000E51	323121			5585	ST A,CNTLOMD	2159.
000E54	C9			5587	RET ,	2160.
				5589	*	2161.
				5590	*	2162.
		00E55		5591	GETTBYTE EQU *	2163.
000E55	3A0221			5593	LD A,CURSX	2164.
000E58	112D13			5595	LODI DE,DODAX	2165.
000E5B	47			5597	LOD B,A	2166.
000E5C	E607			5599	ANDI X'07'	2167.
000E5E	83			5601	ADD E	2168.
000E5F	5F			5603	LOD E,A	2169.
000E60	3E00			5605	LODI A,0	2170.
000E62	8A			5607	ADC D	2171.
000E63	57			5609	LCD D,A	2172.
000E64	CD690E			5611	CALL GETTAB	2173.
000E67	1A			5613	LD A,(DE)	2174.
000E68	C9			5615	RET ,	2175.
				5617	*	2176.
				5618	*	2177.
		00E69		5619	GETTAB EQU *	2178.
000E69	214021			5621	LODI HL,HTABTAB	2179.
000E6C	78			5623	LOD A,E	2180.
000E6D	E6F8			5625	ANDI X'F8'	2181.
000E6F	0F0F0F			5627	ROT R,3	2182.
000E72	4F			5629	LOD C,A	2183.
000E73	85			5631	ADD L	2184.
000E74	6F			5633	LOD L,A	2185.
000E75	3E00			5635	LODI A,0	2186.
000E77	8C			5637	ADC H	2187.
000E78	67			5639	LOD H,A	2188.
000E79	C9			5641	RET ,	2189.
				5643	*	2190.
				5644	*	2191.
		00E7A		5645	SETTAB EQU *	2192.
000E7A	CD550E			5647	CALL GETTBYTE	2193.
000E7D	B6			5649	IOR M	2194.
000E7E	77			5651	ST A,(HL)	2195.
000E7F	C9			5653	RET ,	2196.
				5655	*	2197.
				5656	*	2198.
		00E80		5657	CLRSTAB EQU *	2199.
000E80	CD550E			5659	CALL GETTBYTE	2200.
000E83	EEFF			5661	XORI X'FF'	2201.
000E85	A6			5663	AND M	2202.
000E86	77			5665	ST A,(HL)	2203.

000E87 C9

5667

RET ,

AND GO HOME

2204.

5669 *

2205.

5670 *

2206.

000E88 214021

00E88

5671 CLRATAB

EQU *

CLEARs ALL THE TABS

2207.

5673

LODI HL,HTABTAB

GET ADD OF TAB TABLE

2208.

5675

LODI E,10

NUMBER OF BYTE S IN TABLE

2209.

LOC	OBJECT CODE	ADDR1	ADDR2	STMT	SOURCE	STATEMENT	ASM H V 05 01.23 10/02/76
000E8B	1E0A			5676			
000E8D	97			5677	SUB	A	2210.
		00E8E		5679	CLRTABLP	EQU *	2211.
000E8E	77			5681	ST	A,(HL)	2212.
000E8F	1D			5683	DEC	E	2213.
000E90	23			5685	INC	HL	2214.
000E91	C28E0E			5687	JMP	NZ,CLRTABLP	2215.
000E94	3A1921			5689	LD	A,FLAGS	2216.
000E97	F601			5691	IORI	TABREF	2217.
000E99	321921			5693	ST	A,FLAGS	2218.
000E9C	323121			5695	ST	A,CNTLOMD	2219.
000E9F	C9			5697	RET	.	2220.
				5699	*		2221.
		00EA0		5700	HTAB	EQU *	2222.
				5702	*		2223.
				5703	*		2224.
000EA0	3A1921			5704	LD	A,FLAGS	2225.
000EA3	E601			5706	ANDI	TABREF	2226.
000EA5	3E02			5708	LODI	A,2	2227.
000EA7	1E00			5710	LODI	E,0	2228.
000EA9	C8			5712	RET	Z	2229.
000EAA	3A0221			5714	LD	A,CURSX	2230.
000EAD	FE50			5716	CMP	I LINESIZE-1	2231.
000EAF	F2E30E			5718	JMP	NS,TABWRAP	2232.
000EB2	3C			5720	INC	A	2233.
000EB3	47			5722	LOD	B,A	2234.
000EB4	CD690E			5724	CALL	GETTAB	2235.
000EB7	79			5726	LOD	A,C	2236.
000EB8	D60A			5728	SUB	I 10	2237.
000EBA	57			5730	LOD	D,A	2238.
000EBB	78			5732	LOD	A,B	2239.
000EBC	E607			5734	ANDI	X'07'	2240.
000EBE	47			5736	LOD	B,A	2241.
000EBF	D608			5738	SUB	I 8	2242.
000EC1	5F			5740	LOD	E,A	2243.
000EC2	0E00			5742	LODI	C,0	2244.
000EC4	78			5744	LOD	A,B	2245.
000EC5	87			5746	IOR	A	2246.
000EC6	CAD20E			5748	JMP	Z,OUTTAB	2247.
000EC9	7E			5750	LD	A,(HL)	2248.
		00ECA		5752	TABSHIFT	EQU *	2249.
000ECA	07			5754	ROT	L	2250.
000ECB	05			5756	DEC	B	2251.
000ECC	C2CA0E			5758	JMP	NZ,TABSHIFT	2252.
000ECF	C3D30E			5760	JMP	TABLOOP	2253.
		00ED2		5762	OUTTAB	EQU *	2254.
000ED2	7E			5764	LD	A,(HL)	2255.
		00ED3		5766	TABLOOP	EQU *	2256.
000ED3	0C			5768	INC	C	2257.
000ED4	07			5770	ROT	L	2258.
000ED5	DAED0E			5772	JMP	C,FOUNDTAB	2259.
000ED8	1C			5774	INC	E	2260.
000ED9	FAD30E			5776	JMP	S,TABLOOP	2261.

LOC	OBJECT CODE	ADDR1	ADDR2	STMT	SOURCE	STATEMENT	ASM H V 05 01.23 10/02/76
000EDC	1EF8			5778		LODI E,-8	RESTORE POSITION COINT
000EDE	14			5780		INC D	CHECK FOR END OF TAB TABLE
000EDF	23			5782		INC HL	BUT BUMP POINTER ANYWAY
000EE0	FAD20E			5784		JMP S,OUTTAB	NOT AT END OF TABLE
000EE3	3E51			5786	TABWRAP	LODI A,LINESIZE	
000EE5	210221			5788		LODI HL,CURSX	
000EE8	96			5790		SUB M	GET DISTANCE FROM END
000EE9	5F			5792		LOD E,A	
000EEA	3E01			5794		LODI A,X'01'	INDICATES WRAPAROUND
000EEC	C9			5796		RET ,	GO HOME
		00EED		5798	FOUNDTAB	EQU *	WE GO A TAB
000EED	3A0221			5800		LD A,CURSX	GET OLD X POSITION
000EF0	81			5802		ADD C	ADD IN AMOUNT MOVED
000EF1	320221			5804		ST A,CURSX	SAVE THAT
000EF4	59			5806		LOD E,C	
000EF5	97			5808		SUB A	
000EF6	C9			5810		RET ,	AND GO BACK TO CALLER
				5812	*		
				5813	*		
				5814	*		
				5815	*	HYPERTAB- MOVES CURSOR TO COL 41 NO MATTER WHERE IT IS	
				5816	*	ON THE LINE. IS USEFUL FOR TWO COLUMN TEXT- MAYBE.	
				5817	*		
000EF7	CDFD0E			5818	KEYHYPER	CALL HYPER SUB	
000EFA	C3280D			5820		JMP KEYCUROK	
		00EFD		5822	HYPER SUB	EQU *	
000EFD	EB			5824		XCH HL,DE	SAVE CURSOR POSITION
000EFE	210221			5826		LODI HL,CURSX	X OFFSET
000F01	97			5828		SUB A	
000F02	96			5830		SUB M	
000F03	3D			5832		DEC A	
000F04	3629			5834		LODI M,LINESIZE/2+1	CENTER OF SCREEN
000F06	212900			5836		LODI HL,LINESIZE/2+1	
000F09	85			5838		ADD L	GET OFFSET FROM CENTER
000F0A	6F			5840		LOD L,A	
000F0B	3EFF			5842		LODI A,X'FF'	
000F0D	8C			5844		ADC H	
000F0E	67			5846		LOD H,A	
000F0F	19			5848		ADD HL,DE	ADD OFFSET TO CURSLOC
000F10	23			5850		INC HL	
000F11	C9			5852		RET	
				5854	*		
000F12	D386			5855	DOBELL	OUT BELL	SOUND BELL
000F14	C3800F			5857		JMP TXTEND	AND GET ANOTHER
				5859	*		
				5860	*		
		00F17		5861	GOGRAPHIC	EQU *	ENTER GMODE AND DOES A CLEAR
000F17	3A3221			5863		LD A,ALLOWGMD	FIRST SEE IF WE IS ALLOWED TO ENTER
000F1A	B7			5865		IOR A	GMODE BOSS. IF NOT WE MUST NOT
000F1B	C2800F			5867		JMP NZ,TXTEND	DO IT LIKE THE MAN ASKED.
000F1E	3EC0			5869		LODI A,X'CO'	
000F20	321821			5871		ST A,CLRFLG	
000F23	061D			5873		LODI B,@GS	INDICATES TO GO DO GRAPH
000F25	C32A0F			5875		JMP GOGRPH2	
				5877	*		

00F28 5878 GOGRAPH EQU *

HERE TO ENTER GMODE

2317.

LOC	OBJECT CODE	ADDR1	ADDR2	STMT	SOURCE	STATEMENT	ASM H V 05 01.23 10/02/76	
000F28	0600			5880		LODI B,0	INDICATES DON'T GO DO GRAPHICS	2318.
			00F2A	5882	G0GRPH2	EQU *		2319.
000F2A	3A1F21			5884		LD A,MODEBITS	GET OLD MODEBITS	2320.
000F2D	F610			5886		IORI GRAPHMD	MAKE SURE IT IS ON	2321.
000F2F	321F21			5888		ST A,MODEBITS	SAVE IT	2322.
000F32	D380			5890		OUT MODESET	AND PU IT OUT	2323.
000F34	C38F0F			5892		JMP U,ALPHA	AND GO DO GRAPHICS	2324.
000F37	3A1F21			5894	G0TEXT	LD A,MODEBITS	GET CURRENT MODE	2325.
000F3A	E6EF			5896		ANDI X'EF'	GET RID OF GRAPH BIT	2326.
000F3C	321F21			5898		ST A,MODEBITS	AND STORE IT BACK	2327.
000F3F	D380			5900		OUT MODESET	PUT IT OUT	2328.
000F41	321F21			5902		ST A,MODEBITS	SAVE IT ALSO	2329.
000F44	C3800F			5904		JMP U,XTEND	AND GO TO TEXT MODE	2330.
				5906	*			2331.
				5907	*			2332.
				5908	*			2333.
			00F47	5909	NOGMODE	EQU *	DON'T ALLOW ENTRY INTO GRAPHMODE	2334.
000F47	323221			5911		ST A,ALLOWGMD		2335.
000F4A	C9			5913		RET ,	NZ=DON'T ALLOW GRAPH MOE	2336.
				5915	*			2337.
				5916	*			2338.
			00F4B	5917	YESGMODE	EQU *	ALLOW ENTRY INTO GRAPHMODE	2339.
000F4B	97			5919		SUB A	Z= ALLOW	2340.
000F4C	323221			5921		ST A,ALLOWGMD		2341.
000F4F	C9			5923		RET ,		2342.
				5925	*			2343.
				5926	*			2344.
				5927	*			2345.
			00F50	5928	BRGINT	EQU *	SET BAUD RATE INTERNAL	2346.
000F50	3A1F21			5930		LD A,MODEBITS		2347.
000F53	E6F3			5932		ANDI 255-URTCLOCK	SET THOSE BITS TO ZERO	2348.
000F55	D380			5934		OUT MODESET	PUT THEM OUT	2349.
000F57	321F21			5936		ST A,MODEBITS	AND SAVE THEM	2350.
000F5A	C9			5938		RET ,	AND LEAVE	2351.
				5940	*			2352.
				5941	*			2353.
				5942	*			2354.
			00F5B	5943	BRGEXT	EQU *	SET BAUD RATE FOR EXTERNAL CLOCK	2355.
000F5B	3A1F21			5945		LD A,MODEBITS	GET THE MODEBITS	2356.
000F5E	F60C			5947		IORI URTCLOCK	OR IN THE BITS	2357.
000F60	D380			5949		OUT MODESET	PUT THEM OUT TO HARDWARE	2358.
000F62	321F21			5951		ST A,MODEBITS	SAVE THEM	2359.
000F65	C9			5953		RET ,	AND GO HOME	2360.
				5955	*			2361.
				5956	*			2362.
				5957	*			2363.
			00F66	5958	SETURTX1	EQU *	X 1 CLOCK FOR UART	2364.
000F66	3E57			5960		LODI A,URTINTRS	UART INTERNAL RESET	2365.
000F68	D341			5962		OUT URTCTL		2366.
000F6A	3E79			5964		LODI A,URTX1MD		2367.
000F6C	D341			5966		OUT URTCTL	AND SAVE THEM	2368.
000F6E	C9			5968		RET ,	AND GO HOME	2369.
				5970	*			2370.
				5971	*			2371.
				5972	*			2372.

000F6F 3E57
000F71 D341
000F73 3E7A

00F6F 5973 *
5974 SETURTX6 EQU *
5976 LODI A,URTINRS
5978 OUT URTCTL
5980 LCDI A,URTMODE

X 16 CLOCK FOR UART (NORMAL MODE)
UART INTERNAL RESET

2373.
2374.
2375.
2376.
2377.

LOC	OBJECT CODE	ADDR1	ADDR2	STMT	SOURCE STATEMENT	ASM H V 05 01.23 10/02/76
000F75	D341			5982	OUT URTCTL	2378.
000F77	C9			5984	RET ,	2379.
				5986	*	2380.
				5987	*	2381.
				5988	*	2382.
				5989	*	2383.
		00F78		5990	SETNECHO EQU *	2384.
000F78	3A1921			5992	LD A,FLAGS	2385.
000F7B	F680			5994	IORI FULLDUPL	2386.
000F7D	321921			5996	ST A,FLAGS	2387.
000F80	21FAFF			5998	TXTEND RESUME DISPL,CS	2388.
000F83	39			6000		
000F84	220020			6001		
000F87	2E02			6002		
000F89	CD0C06			6003		
000F8C	C33E0C			6004	JMP U,TXTMOD	2389.
				6006	*	2390.
				6007	*	2391.
				6008	*	2392.

LOC	OBJECT CODE	ADDR1	ADDR2	STMT	SOURCE STATEMENT	ASM H V 05 01.23 10/02/76
				6010	*****	2394.
				6011	*****	2395.
				6012	* THIS IS IT	2396.
				6013	* WHAT YOU ASK?	2397.
				6014	* WHY, THE TEKTRONIX GRAPH SIMULATOR	2398.
				6015	*	2399.
				6016	*****	2400.
000F8F				6017	ALPHA DS 0X	2401.
000F8F	210000			6019	LODI HL,X'00'	2402.
000F92	7D			6021	LOD A,L	2403.
000F93	326421			6023	ST A,MARGIN	2404.
000F96	320221			6025	ST A,XOFF	2405.
000F99	320421			6027	ST A,CURSCHAR	2406.
000F9C	21F606			6029	LODI HL,TOPOFF+(7*LINESIZE) PLACE TO PUT CHARACTERS AT	2407.
000F9F	225C21			6031	ST HL,CHARPOS	2408.
000FA2	78			6033	LOD A,B	2409.
000FA3	FE1D			6035	CMPI @GS	2410.
000FA5	CAA911			6037	JMP Z,GRAPH	2411.
				6039	*	2412.
000FA8				6040	ALPHA2 DS 0X	2413.
000FA8	2A0021			6042	LD HL,CURSLOC	2414.
000FAB	EB			6044	XCH HL,DE	2415.
000FAC	2A5C21			6046	LD HL,CHARPOS	2416.
000FAF	01F058			6048	LODI BC,GRAPHEVN	2417.
000FB2	09			6050	ADD HL,BC	2418.
000FB3	7E			6052	LD A,(HL)	2419.
000FB4	47			6054	LOD B,A	2420.
000FB5	220021			6056	ST HL,CURSLOC	2421.
000FB8	3A0421			6058	LD A,GRPOLD	2422.
000FBB	12			6060	ST A,(DE)	2423.
000FBC	78			6062	LOD A,B	2424.
000FBD	320421			6064	ST A,GRPOLD	2425.
000FC0	21FAFF			6066	RESUME DISPL,CS	2426.
000FC3	39			6068		
000FC4	220020			6069		
000FC7	2E02			6070		
000FC9	CD0C06			6071		
000FCC	FEFE			6072	ALPHACHK CMPI X'FE'	2427.
000FCE	CA370F			6074	JMP Z,GOTEXT	2428.
000FD1	21B6C9			6076	LODI HL,X'C9B6'	2429.
000FD4	227121			6078	ST HL,GRPDRAW	2430.
000FD7	E67F			6080	ANDI X'7F'	2431.
000FD9	FE20			6082	CMPI X'20'	2432.
000FDB	F25B10			6084	JMP NS,GRPNCTRL	2433.
000FDE	21E30F			6086	LODI HL,CMDTAB4	2434.
000FE1	CD3E06			6088	CALL SEARCH	2435.
000FE4	E9			6090	JMP (HL)	2436.
		00FE3		6092	CMDTAB4 CMDEF 1D,GRAPH	2437.
000FE5	1D11A9			6094		
000FE8	0C0F8F			6095	CMDEF 0C,ALPHA	2438.
000FEB	0A10C1			6097	CMDEF 0A,LINEFEED	2439.
000FEE	09100C			6099	CMDEF 09,GRPTAB	2440.
000FF1	1F0FA8			6101	CMDEF 1F,ALPHA2	2441.
				6103	CMDEF 11,ALPHA2	2442.
					DON'T DISPLAY @DC1	

LOC	OBJECT CODE	ADDR1	ADDR2	STMT	SOURCE	STATEMENT	ASM H V 05 01.23 10/02/76
000FF4	110FA8			6104			
000FF7	010FA8			6105			
000FFA	1410B0			6107	CMDEF	01,ALPHA2	2443.
000FFD	0D1027			6109	CMDEF	14,GRPDC4	2444.
001000	0710BC			6111	CMDEF	0D,GRAPHCAR	2445.
001003	08102D			6113	CMDEF	07,GRPBELL	2446.
001006	0B1049			6115	CMDEF	08,GRPBS	2447.
001009	00105B			6117	CMDEF	0B,GRPRL	2448.
00100C	CDA00E			6119	CMDEF	00,GRPNCTRL	2449.
00100F	3D			6121	GRPTAB	CALL HTAB	2450.
001010	CA9510			6123		DEC A	2451.
001013	1600			6125		JMP Z,GRPARND	2452.
001015	2A5C21			6127		LODI D,0	2453.
001018	19			6129		LD HL,CHARPOS	2454.
001019	FAB610			6131		ADD HL,DE	2455.
00101C	3E20			6133		JMP S,GRPSTCUR	2456.
00101E	C35B10			6135		LODI A,' '	2457.
001021	CD9B10			6137		JMP GRPNCTRL	2458.
001024	C3A80F			6139		CALL GRPCR	2459.
001027	CD9B10			6141	GRAPHCAR	JMP ALPHA2	2460.
00102A	C3A80F			6143		CALL GRPCR	2461.
00102D	11FFFF			6145	GRPBS	JMP ALPHA2	2462.
001030	3A0221			6147		LODI DE,-1	2463.
001033	3D			6149		LD A,XOFF	2464.
001034	F23C10			6151		DEC A	2465.
001037	1119FE			6153		JMP NS,GRPBSOK	2466.
00103A	3E50			6155		LODI DE,-(6*LINESIZE)-1	2467.
00103C				6157	GRPBSOK	LODI A,(LINESIZE-1)	2468.
00103C	320221			6159		DS 0X	2469.
00103F	2A5C21			6161		ST A,XOFF	2470.
001042	19			6163		LD HL,CHARPOS	2471.
001043	225C21			6165		ADD HL,DE	2472.
001046	C3A80F			6167		ST HL,CHARPOS	2473.
001049	2A5C21			6169	GRPRL	JMP ALPHA2	2474.
00104C	11C9FD			6171		LD HL,CHARPOS	2475.
00104F	19			6173		LODI DE,-(7*LINESIZE)	2476.
001050	225C21			6175		ADD HL,DE	2477.
001053	7C			6177		ST HL,CHARPOS	2478.
001054	B7			6179		LOD A,H	2479.
001055	F2A80F			6181		IOR A	2480.
001058	C38F0F			6183		JMP NS,ALPHA2	2481.
00105B				6185	GRPNCTRL	JMP ALPHA	2482.
00105B	2A6521			6187		DS 0X	2483.
00105E	07070707			6189		LD HL,GRPCSET	2484.
001062	57			6191		ROT L,4	2485.
001063	E6F0			6193		LOD D,A	2486.
001065	B5			6195		ANDI X'F0'	2487.
						IOR L	2488.

LOC	OBJECT CODE	ADDR1	ADDR2	STMT	SOURCE	STATEMENT	ASM H V 05 01.23 10/02/76
001066	4F			6197		LOD C,A	2489.
001067	7A			6199		LCD A,D	2490.
001068	E607			6201		ANDI X'07'	2491.
00106A	B4			6203		IOR H	2492.
00106B	47			6205		LOD B,A	2493.
00106C	2A5C21			6207		LD HL,CHARPOS	2494.
00106F	E5			6209		PUSH HL	2495.
001070	97			6211		SUB A	2496.
001071	326221			6213		ST A,EVENODD	2497.
001074				6215	GENLOOP	DS 0X	2498.
001074	0A			6217		LD A,(BC)	2499.
001075	2F			6219		CMA	2500.
001076	6F			6221		LOD L,A	2501.
001077	3E02			6223		LODI A,X'02'	2502.
001079	CD4F11			6225		CALL XYPTINC	2503.
00107C	0B			6227		DEC BC	2504.
00107D	79			6229		LOD A,C	2505.
00107E	E60F			6231		ANDI X'0F'	2506.
001080	C27410			6233		JMP NZ,GENLOOP	2507.
001083	E1			6235		POP HL	2508.
001084	23			6237		INC HL	2509.
001085	225C21			6239		ST HL,CHARPOS	2510.
001088	3A6421			6241		LD A,MARGIN	2511.
00108B	210221			6243		LODI HL,XOFF	2512.
00108E	34			6245		INC M	2513.
00108F	86			6247		ADD M	2514.
001090	FE51			6249		CMP I LINESIZE	2515.
001092	C2A80F			6251		JMP NZ,ALPHA2	2516.
001095	CD9B10			6253	GRPARND	CAL GRPCR	2517.
001098	C3C110			6255		JMP LINEFEED	2518.
00109B				6257	GRPCR DS	DS 0X	2519.
00109B	2A5C21			6259		LD HL,CHARPOS	2520.
00109E	EB			6261		XCH HL,DE	2521.
00109F	3A0221			6263		LD A,XOFF	2522.
0010A2	2F			6265		CMA	2523.
0010A3	6F			6267		LOD L,A	2524.
0010A4	26FF			6269		LODI H,X'FF'	2525.
0010A6	23			6271		INC HL	2526.
0010A7	19			6273		ADD HL,DE	2527.
0010A8	225C21			6275		ST HL,CHARPOS	2528.
0010AB	97			6277		SUB A	2529.
0010AC	320221			6279		ST A,XOFF	2530.
0010AF	C9			6281		RET	2531.
0010B0				6283	GRPDC4	DS 0X	2532.
0010B0	2A5C21			6285		LD HL,CHARPOS	2533.
0010B3	CDFD0E			6287		CALL HYPERSUB	2534.
0010B6	225C21			6289	GRPSTCUR	ST HL,CHARPOS	2535.
0010B9	C3A80F			6291		JMP ALPHA2	2536.
				6293	GRPBELL	OUT BELL	2537.

LOC	OBJECT CODE	ADDR1	ADDR2	STMT	SOURCE STATEMENT
0010BC	D386			6294	
0010BE	C3A80F			6295	JMP ALPHA2
				6297 *	
				6298 *	

ASM H V 05 01.23 10/02/76

2538.
2539.
2540.

LOC	OBJECT CODE	ADDR1	ADDR2	STMT	SOURCE STATEMENT	ASM H V 05 01.23 10/02/76
0010C1				6300	LINEFEED DS 0X	2542.
0010C1	2A5C21			6302	LD HL,CHARPOS	2543.
0010C4	EB			6304	XCH HL,DE	2544.
0010C5	21D9B1			6306	LODI HL,-(247*LINESIZE)	2545.
0010C8	19			6308	ADD HL,DE	2546.
0010C9	7C			6310	LOD A,H	2547.
0010CA	B7			6312	IOR A	2548.
0010CB	F2D810			6314	JMP NS,SWITCH	2549.
0010CE	213702			6316	LODI HL,(7*LINESIZE)	2550.
0010D1	19			6318	ADD HL,DE	2551.
0010D2	225C21			6320	ST HL,CHARPOS	2552.
0010D5	C3A80F			6322	JMP ALPHA2	2553.
0010D8				6324	SWITCH DS 0X	2554.
0010D8	3A6421			6326	LD A,MARGIN	2555.
0010DB	B7			6328	IOR A	2556.
0010DC	C28F0F			6330	JMP NZ,ALPHA	2557.
0010DF	211E07			6332	LODI HL,LINESIZE/2+TOPOFF+7*LINESIZE IS MARGIN SIZE	2558.
0010E2	225C21			6334	ST HL,CHARPOS	2559.
0010E5	3E28			6336	LODI A,LINESIZE/2	2560.
0010E7	326421			6338	ST A,MARGIN	2561.
0010EA	97			6340	SUB A	2562.
0010EB	320221			6342	ST A,XOFF	2563.
0010EE	C3A80F			6344	JMP ALPHA2	2564.
6346	*****					2565.
6347	*****					2566.
6348	*****					2567.
0010F1				6349	SHIFT DS 0X	2568.
0010F1	97			6351	SUB A	2569.
0010F2	7A			6353	LOD A,D	2570.
0010F3	1F			6355	ROT RC	2571.
0010F4	57			6357	LCD D,A	2572.
0010F5	7B			6359	LOD A,E	2573.
0010F6	1F			6361	ROT RC	2574.
0010F7	5F			6363	LOD E,A	2575.
0010F8	C9			6365	RET	2576.
6367	*****					2577.
6368	*****					2578.
6369	*****					2579.
0010F9				6370	BTOELOAD DS 0X	2580.
0010F9	46			6372	LOD B,M	2581.
0010FA	23			6374	INC HL	2582.
0010FB	4E			6376	LCD C,M	2583.
0010FC	23			6378	INC HL	2584.
0010FD	56			6380	LOD D,M	2585.
0010FE	23			6382	INC HL	2586.
0010FF	5E			6384	LOD E,M	2587.
001100	C9			6386	RET	2588.
6388	*****					2589.
6389	*****					2590.
6390	*****					2591.
001101				6391	BTOESAVE DS 0X	2592.
001101	70			6393	LOD M,B	2593.
001102	23			6395	INC HL	2594.
001103	71			6397	LOD M,C	2595.
001104	23			6399	INC HL	2596.

LOC	OBJECT CODE	ADDR1	ADDR2	STMT	SOURCE STATEMENT	ASM H V 05 01.23 10/02/76
001105	72			6401	LOD M,D	2597.
001106	23			6403	INC HL	2598.
001107	73			6405	LOD M,E	2599.
001108	C9			6407	RET ,	2600.
				6409	*****	2601.
				6410	*****	2602.
				6411	*****	2603.
				6412	*	2604.
				6413	* SCOPLOAD - ROUTINE POSITIONS GRAPHMODE 'POINT' AT	2605.
				6414	* THE POSITION INDICATED BY BC AND DE.	2606.
				6415	* BC HAS X COORIDINATE (RANGE = 0 TO 647)	2607.
				6416	* DE HAS Y CORRIDINATE (RANGE = 0 TO 480)	2608.
				6417	* ORIGIN IS LOWER LEFT HAND CORNER.	2609.
				6418	* (N.B. NO SCISSORING IS PERFORMED, SO THAT	2610.
				6419	* COORIDINATES > 647 OR 480 WILL DO WIERD	2611.
				6420	* THINGS, EXCEPT THAT Y CAN GO TO 512.)	2612.
				6421	*	2613.
				6422	*****	2614.
001109				6423	SCOPLOAD DS 0X	2615.
001109	7B			6425	LOD A,E	2616.
00110A	0F			6427	ROT R	2617.
00110B	E680			6429	ANDI X'80'	2618.
00110D	326221			6431	ST A,EVENODD	2619.
001110	CDF110			6433	CALL SHIFT	2620.
001113	3EFF			6435	LODI A,255	2621.
001115	93			6437	SUB E	2622.
001116	6F			6439	LOD L,A	2623.
001117	2600			6441	LODI H,0	2624.
001119				6443	MULT 81	2625.
001119	545D			6445		
00111B	29			6446		
00111C	29			6447		
00111D	29			6448		
00111E	29			6449		
00111F	EB			6450		
001120	19			6451		
001121	EB			6452		
001122	29			6453		
001123	29			6454		
001124	19			6455		
001125	225C21			6456	ST HL,CHARPOS	2626.
				6458	*	2627.
				6459	* NOW WE DO THE X PART	2628.
				6460	*	2629.
				6461	*	2630.
001128	50			6462	LOD D,B	2631.
001129	59			6464	LOD E,C	2632.
00112A	7B			6466	LOD A,E	2633.
00112B	E607			6468	ANDI X'07'	2634.
00112D	212D13			6470	LODI HL,DODAX	2635.
001130	85			6472	ADD L	2636.
001131	6F			6474	LOD L,A	2637.
001132	3E00			6476	LODI A,0	2638.
001134	8C			6478	ADC H	2639.
001135	67			6480	LOD H,A	2640.

001136	7E	6482	LOD	A,M	GET THE MASK	2641.
001137	326321	6484	ST	A,XMASK	AND SAVE IT	2642.
00113A	CDF110	6486	CALL	SHIFT	GET RID OF BIT OFFSET	2643.
00113D	CDF110	6488	CALL	SHIFT	WE ONLY WANT CHARACTER OOF	2644.
001140	CDF110	6490	CALL	SHIFT	WERE DONE	2645.
001143	7B	6492	LOD	A,E		2646.
001144	320221	6494	ST	A,XOFF		2647.
001147	2A5C21	6496	LD	HL,CHARPOS	GET SUM OF X AND Y	2648.
00114A	19	6498	ADD	HL,DE	NOTE D SHOULD BE ZERO SO ADD IN X	2649.
00114B	225C21	6500	ST	HL,CHARPOS	AND SAVE IT	2650.
00114E	C9	6502	RET	.	AND LEAVE	2651.
		6504	*****			2652.

LOC	OBJECT CODE	ADDR1	ADDR2	STMT	SOURCE STATEMENT	ASM H V 05 01.23 10/02/76
				6505	*****	2653.
				6506	*****	2654.
00114F				6507	XYPTINC DS OX	2655.
				6509	*****THIS IS THE REALLY IMPORTANT ROUTINE	2656.
				6510	***IT INCREMENTS AND DECREMENTS X AND Y	2657.
				6511	***** DEPENDING ON THE CONTENTS OF A	2658.
				6512	** 02 DEC Y (THAT MEANS UP)	2659.
				6513	** 03 INC Y (THAT OBVIOUSLY MEANS DOWN ALSO)	2660.
				6514	** 08 INC X	2661.
				6515	** 0C DEC X	2662.
				6516	** YOU CAN DO BOTH X AND Y AT THE SAME TIME	2663.
				6517	*	2664.
00114F	C5			6518	PUSH BC	2665.
001150	E5			6520	PUSH HL	2666.
001151	D5			6522	PUSH DE	2667.
001152	4F			6524	LOD C,A	2668.
001153	45			6526	LOD B,L	2669.
		02162		6528	DRAGEVOD EQU EVENODD	2670.
		0215C		6530	DRAGXY EQU CHARPOS	2671.
		02163		6532	DRAGBIT EQU XMASK	2672.
				6534	LD HL,DRAGXY	2673.
				6536	ANDI X'0C'	2674.
				6538	JMP Z,DRAGDOY	2675.
				6540	LOD A,B	2676.
				6542	JMP PE,DRAG1	2677.
				6544	ROT R	2678.
				6546	JMP NC,DRAG2	2679.
				6548	INC HL	2680.
				6550	DRAG2 LOD B,A	2681.
				6552	DRAGDOY LCD A,C	2682.
				6554	ANDI X'03'	2683.
				6556	LD A,DRAGEVOD	2684.
				6558	JMP Z,DRAGSPOT	2685.
				6560	JMP PO,DRAG3	2686.
				6562	XORI X'80'	2687.
				6564	JMP NS,DRAG5	2688.
				6566	LODI DE,LINESIZE	2689.
				6568	JMP DRAG6	2690.
				6570	DRAG3 XORI X'80'	2691.
				6572	JMP S,DRAG5	2692.
				6574	LODI DE,-LINESIZE	2693.
				6576	DRAG6 ADD HL,DE	2694.
				6578	DRAG5 ST A,DRAGEVOD	2695.
				6580	DRAGSPOT ST HL,DRAGXY	2696.
				6582	LODI DE,GRAPHVFN	2697.
				6584	ICR A	2698.
				6586	JMP NS,DRAG4	2699.
				6588	LODI DE,GRAPHODD	2700.
				6590	DRAG4 ADD HL,DE	2701.
				6592	LOD A,B	2702.
				6594	CALL GRPDRAW	2703.
				6596	LOD M,A	2704.
				6598	POP DE	2705.
				6600	POP HL	2706.

direction bits
XMASK

LOC	OBJECT CODE	ADDR1	ADDR2	STMT	SOURCE	STATEMENT	ASM H V 05 01.23 10/02/76
00119E	68			6602		LOD L,B	2707.
00119F	C1			6604		POP BC	2708.
0011A0	C9			6606		RET ,	2709.
0011A1	07			6608	DRAG1	ROT L	2710.
0011A2	D26511			6610		JMP NC,DRAG2	2711.
0011A5	2B			6612		DEC HL	2712.
0011A6	C36511			6614		JMP DRAG2	2713.
				6616	*		2714.

LOC	OBJECT CODE	ADDR1	ADDR2	STMT	SOURCE STATEMENT	ASM H V 05 01.23 10/02/76
				6618	*****	2716.
				6619	*****	2717.
				6620	*****HERE IS THE GRAPHING STUFF	2718.
				6621	*****	2719.
				6622	*****	2720.
0011A9				6623	GRAPH DS 0X	2721.
0011A9	2A0021			6625	LD HL,CURSLOC	2722.
0011AC	EB			6627	XCH HL,DE	2723.
0011AD	210000			6629	LODI HL,0 (ASSUMES ROM AT LOCATION 0)	2724.
0011B0	220021			6631	ST HL,CURSLOC MOVE CURSOR OFF SCREEN	2725.
0011B3	7C			6633	LOD A,H	2726.
0011B4	326421			6635	ST A,MARGIN SET MARGIN A LEFT	2727.
0011B7	3A0421			6637	LD A,GRPOLDC	2728.
0011BA	12			6639	ST A,(DE)	2729.
0011BB	97			6641	SUB A	2730.
0011BC	320421			6643	ST A,GRPOLDC	2731.
0011BF	CD0112			6645	CAL GRAPHPT GET A POINT IN TEKXXX	2732.
0011C2	CD4C12			6647	CAL GRAPHCON CONVERT TO 10 BIT NOS. IN BCDE	2733.
0011C5	214E21			6649	DLD HL,GRAPHPOS DRAW A DARK VECTOR (IE JUST SAVE	2734.
0011C8	CD0111			6651	CALL BTOESAVE NEW COORDS)	2735.
0011CB	CD0911			6653	CAL SCOPLOAD	2736.
				6655	*	2737.
				6656	*	2738.
				6657	*	2739.
				6658	GRAPHLP DS 0X DRAW DARK VECTORS	2740.
0011CE	CD0112			6660	CAL GRAPHPT GET NEXT POINT IN TEKXXX	2741.
0011D1	CD4C12			6662	CAL GRAPHCON CONVERT TO 10 BIT NOS. IN BCDE	2742.
0011D4	215821			6664	DLD HL,GRAPHTEM SAVE IN TEMPORAY	2743.
0011D7	CD0111			6666	CALL BTOESAVE	2744.
0011DA	215121			6668	DLD HL,GRAPHPOS+3 ADDRESS CURRENT POSITION	2745.
0011DD	7B			6670	LOD A,E FORM DELTA X, DELTA Y	2746.
0011DE	96			6672	SUB M	2747.
0011DF	5F			6674	LOD E,A	2748.
0011E0	7A			6676	LOD A,D *	2749.
0011E1	2D			6678	DEC L	2750.
0011E2	9E			6680	SBB M	2751.
0011E3	57			6682	LOD D,A	2752.
0011E4	79			6684	LOD A,C **	2753.
0011E5	2D			6686	DEC L	2754.
0011E6	96			6688	SUB M	2755.
0011E7	4F			6690	LOD C,A	2756.
0011E8	78			6692	LOD A,B *	2757.
0011E9	2D			6694	DEC L	2758.
0011EA	9E			6696	SBB M	2759.
0011EB	47			6698	LOD B,A	2760.
0011EC	CD8712			6700	CAL VECTOR NOW DRAW A REAL VECTOR	2761.
0011EF	215821			6702	DLD HL,GRAPHTEM RETRIEVE THE FINAL COORDS	2762.
0011F2	CDF910			6704	CALL BTOELOAD	2763.
0011F5	214E21			6706	DLD HL,GRAPHPOS SAVE AS CURRENT POSITION	2764.
0011F8	CD0111			6708	CALL BTOESAVE	2765.
0011FB	CD0911			6710	CAL SCOPLOAD ALSO LOAD INTO HARDWARE	2766.

LOC	OBJECT CODE	ADDR1	ADDR2	STMT	SOURCE STATEMENT	ASM H V 05 01.23 10/02/76
				6712	* THE ABOVE IS NOT STRICTLY NECESSARY, SINCE 'VECTOR' SHOULD HAVE	2767.
				6713	* CORRECTLY UPDATED THE HARDWARE REGISTERS. IT PAYS TO BE CAUTIOUS,	2768.
				6714	* HOWEVER, SINCE ANY ERROR IS CUMULATIVE.	2769.
0011FE	C3CE11			6715	JMP GRAPHLP CONTINUE UNTIL GRAPH MODE IS ENDED	2770.

LOC	OBJECT CODE	ADDR1	ADDR2	STMT	SOURCE STATEMENT	ASM H V 05 01.23 10/02/76
				6718 *		2772.
				6719 *	GRAPHPT: GET GRAPH POINT COORDS FROM BUFFER & PUT IN TEKXXX	2773.
				6720 *		2774.
001201	0E01			6721	GRAPHPT LODI C,1	2775.
001203				6723	GRAPHNEX DS 0X	2776.
001203	21FAFF			6725	RESUME DISPL,CS	2777.
001206	39			6727		
001207	220020			6728		
00120A	2E02			6729		
00120C	CD0C06			6730		
00120F	FE20			6731		
001211	F21D12			6733	CMPI X*20'	2778.
001214	FE0F			6735	JMP P,NOTCTL	2779.
001216	CA4312			6737	CMPI @SI	CHECK FOR VECTOR ERASING MODE
001219	E1			6739	JMP Z,ALTGRP	GO SETUP CODE FOR ERASE VECTORS
00121A	C3CC0F			6741	POP HL	2782.
00121D	47			6743	JMP ALPHACHK	2783.
00121E	214C21			6745	LOD B,A	SAVE IN B
001221	1717			6747	DLD HL,TEKYHI	ASSUME Y HIGH
001223	DA2F12			6749	ROT LC,2	TEST HIGH-ORDER CONTROL BIT
001226	0D			6751	JMP C,GRAPHLOW	LOW X OR Y...
001227	CA3812			6753	DEC C	HIGH X OR Y: IS THIS THE FIRST
00122A	2E4A			6755	JMP Z,GRAPHST	YES - Y HIGH WAS RIGHT
00122C	C33812			6757	LODI L,TEKXHI,>	ELSE USE X HIGH
00122F	0D			6759	JMP GRAPHST	2791.
001230	2E4B			6761	GRAPHLOW DEC C	FLAG 'COORDNATE RECEIVED'
001232	17			6763	LODI L,TEKXLOW,>	ASSUME X LOW
001233	D23812			6765	ROT LC	TEST 2ND CONTROL BIT
001236	2E4D			6767	JMP NC,GRAPHST	2795.
001238	78			6769	LODI L,TEKYLOW,>	WRONG - Y LOW
001239	E61F			6771	GRAPHST LOD A,B	RETRIEVE THE CHAR
00123B	77			6773	ANDI X*1F'	REMOVE CONTROL BITS
00123C	3E4B			6775	LOD M,A	STORE IT IN TEKXXX
00123E	95			6777	LODI A,TEKXLOW,>	WAS THIS XLOW?
00123F	C20312			6779	SUB L	2801.
001242	C9			6781	JMP NZ,GRAPHNEX	NO - WAIT FOR MORE CORRDS
001243	212FA6			6783	RET	YES - END OF POINT
001246	227121			6785	ALTGRP LODI HL,X(A62F)	ERASE POINTS (CMA, AND M)
001249	C30312			6787	ST HL,GRPDRAW	2805.
				6789 *	JMP GRAPHNEX	GO GET ANOTHER CHARACTER
				6790 *		2807.
				6791 *	GRAPHCON: CONVERT 5 BIT COORDS AT TEKXXX TO 10 BIT IN BCDE	2808.
				6792 *		2809.
00124C	214A21			6793	GRAPHCON DLD HL,TEKXHI	POINT TO X
00124F	CD5512			6795	CAL GRAPHCNV	DO X IN DE
001252	42			6797	LOD B,D	MOVE X TO BC
001253	4B			6799	LOD C,E	2813.
001254	2C			6801	INC L	POINT TO Y
001255	7E			6803	GRAPHCNV LOD A,M	DO Y (X) TO DE
001256	0F0F0F			6805	ROT R,3	2816.
001259	5F			6807	LOD E,A	2817.
00125A	E603			6809	ANDI X*03'	ISOLATE 2 HIGH BITS
00125C	57			6811	LOD D,A	PUT IN D
00125D	2C			6813	INC L	POINT TO LOW PART
00125E	7B			6815	LCD A,E	2821.
						2822.

LOC	OBJECT CODE	ADDR1	ADDR2	STMT	SOURCE	STATEMENT	ASM H V 05 01.23 10/02/76	
00125F	E6E0			6817	ANDI	X'E0'	3 BITS LEFT OVER FROM HIGH PART	2823.
001261	86			6819	ADD	M	ADD IN LOW PART	2824.
001262	5F			6821	LOD	E,A	PUT IN E	2825.
001263	3A6021			6823	LD	A,TYPE		2826.
001266	FE42			6825	CMPI	'B'	BIG MODE	2827.
001268	C8			6827	RET	Z	THEN NO SCALLING	2828.
001269	FE53			6829	CMPI	'S'	SMALL SIZE	2829.
00126B	C27312			6831	JMP	NZ,SCALNORM	IF NOT DC4 THE 5/8	2830.
00126E	13			6833	INC	DE	FOR ROUNDING	2831.
00126F	CDF110			6835	CALL	SHIFT	SHIFT DE RIGHT ONE IBT	2832.
001272				6837	OTO	DS	0X	2833.
001272	C9			6839	RET	,	ABOVE, OR TO CALLER	2834.
		01273		6841	SCALNORM	EQU	*	2835.
001273	62			6843	LOD	H,D	GET HIGH PART	2836.
001274	6B			6845	LOD	L,E	GET LOW PART	2837.
001275	29			6847	ADD	HL,HL		2838.
001276	29			6849	ADD	HL,HL	*4	2839.
001277	19			6851	ADD	HL,DE	*5	2840.
001278	EB			6853	XCH	HL,DE	PUT IN DE	2841.
001279	CDF110			6855	CALL	SHIFT		2842.
00127C	CDF110			6857	CALL	SHIFT		2843.
00127F	13			6859	INC	DE		2844.
001280	CDF110			6861	CALL	SHIFT	DIVIDE BY EIGHT	2845.
001283	214B21			6863	LODI	HL,TEKXLOW	FOR SECOND TIM AROUND	2846.
001286	C9			6865	RET	,		2847.

LOC	OBJECT CODE	ADDR1	ADDR2	STMT	SOURCE STATEMENT	ASM H V 05 01.23 10/02/76
				6869 *		2850.
				6870 *		2851.
				6871 *	VECTOR DRAW A VECTOR TO ANY POINT ON THE SCREEN.	2852.
				6872 *		2853.
				6873 *	SIGNED DELTA X (DX) IS IN BC	2854.
				6874 *	SIGNED DELTA Y (DY) IS IN DE	2855.
				6875 *		2856.
001287				6876	VECTOR DS OX	2857.
	00008			6878	INCX EQU X'08'	2858.
	0000C			6880	DECX EQU X'0C'	2859.
	00002			6882	INCY EQU X'02'	2860.
	00003			6884	DECY EQU X'03'	2861.
				6886 *		2862.
				6887 *	FIRST MAKE BOTH DELTAS POSITIVE, AND INSURE DX GE DY	2863.
				6888 *	UPPER H WILL CONTAIN 'DY' CNTRL BITS, LOWER H 'DX'	2864.
				6889 *		2865.
001287	2628			6890	LODI H, INCX+16*INCY	2866.
001289	78			6892	LCD A,B	2867.
00128A	B7			6894	IOR A	2868.
00128B	F29812			6896	JMP NS, TESTY	2869.
00128E	262C			6898	LODI H, DECX+16*INCY	2870.
001290	3E00			6900	LODI A,0	2871.
001292	91			6902	SUB C	2872.
001293	4F			6904	LCD C,A	2873.
001294	3E00			6906	LODI A,0	2874.
001296	98			6908	SBB B	2875.
001297	47			6910	LCD B,A	2876.
001298	7A			6912	TESTY LCD A,D	2877.
001299	B7			6914	IOR A	2878.
00129A	F2A912			6916	JMP NS, TESTBIG	2879.
00129D	7C			6918	LCD A,H	2880.
00129E	F630			6920	IORI 16*DECY	2881.
0012A0	67			6922	LCD H,A	2882.
0012A1	3E00			6924	LODI A,0	2883.
0012A3	93			6926	SUB E	2884.
0012A4	5F			6928	LCD E,A	2885.
0012A5	3E00			6930	LODI A,0	2886.
0012A7	9A			6932	SBB D	2887.
0012A8	57			6934	LCD D,A	2888.
0012A9	78			6936	TESTBIG LOD A,B	2889.
0012AA	92			6938	SUB D	2890.
0012AB	DAB612			6940	JMP C, YBIG	2891.
0012AE	C2C212			6942	JMP NZ, XBIG	2892.
0012B1	79			6944	LCD A,C	2893.
0012B2	93			6946	SUB E	2894.
0012B3	D2C212			6948	JMP NC, XBIG	2895.
0012B6	7C			6950	YBIG LOD A,H	2896.
0012B7	0F0F0F0F			6952	ROT R,4	2897.
0012BB	67			6954	LCD H,A	2898.
0012BC	78			6956	LCD A,B	2899.
0012BD	42			6958	LCD B,D	2900.
0012BE	57			6960	LCD D,A	2901.

0012BF 79
0012C0 4B
0012C1 5F
0012C2

6962
6964
6966
6968 XBIG
6970 *
6971 *
6972 *
6973 *
6974 *
6975 *

LOD A,C
LOD C,E
LOD E,A
DS 0X

EXCHANGE LOW-ORDER DELTAS

DIVIDE THE INTERVAL INTO HALVES OR QUARTERS SO THAT BOTH
DELTAS ARE LT 256.
(NOTE: THIS IMPLIES THAT FOR EXACT RESULTS, DELTAS OVER 511
SHOULD BE DIVISIBLE BY 4, AND THOSE OVER 255 SHOULD BE
DIVISIBLE BY 2. IF THIS STRICTURE IS IGNORED, HOWEVER,

2902.
2903.
2904.
2905.
2906.
2907.
2908.
2909.
2910.
2911.

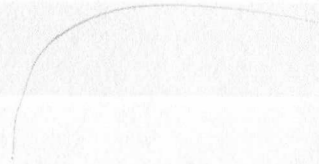
LOC	OBJECT CODE	ADDR1	ADDR2	STMT	SOURCE STATEMENT	ASM H V 05 01.23 10/02/76
				6976 *	THIS MAXIMUM TOTAL ERROR PER VECTOR IS ONLY 3 UNITS OVER	2912.
				6977 *	THE FULL LENGTH.	2913.
				6978 *		2914.
0012C2	2E01			6979	LODI L,1	TENTATIVELY SET FOR 1 SEGMENT
0012C4	78			6981	REDUCE LCD A,B	IS DX GT 255?
0012C5	B7			6983	IOR A	(ALSO CLR CARRY FOR LATER)
0012C6	CADC12			6985	JMP Z,NOTOOBIG	NO - FNE
				6987 *	PRECEEDING CODE INSURED DX GE DY, SO DY GT 255 NEEDN'T BE CHECKED.	2919.
0012C9	78			6988	LOD A,B	SHIFT DX (BC) RIGHT 1 BIT
0012CA	1F			6990	ROT RC	(CARRY MUST BE CLEAR)
0012CB	47			6992	LOD B,A	
0012CC	79			6994	LOD A,C	
0012CD	1F			6996	ROT RC	
0012CE	4F			6998	LOD C,A	
0012CF	AF			7000	XOR A	SHIFT DY (DE) RIGHT 1 BIT
0012D0	7A			7002	LOD A,D	
0012D1	1F			7004	ROT RC	
0012D2	57			7006	LOD D,A	
0012D3	7B			7008	LOD A,E	
0012D4	1F			7010	ROT RC	
0012D5	5F			7012	LOD E,A	
0012D6	7D			7014	LOD A,L	DOUBLE THE SEGMENT(REPEAT) COUNT
0012D7	07			7016	ROT L	
0012D8	6F			7018	LOD L,A	
0012D9	C3C412			7020	JMP REDUCE	TRY AGAIN (MAX OF 2 TIMES)
0012DC	45			7022	NOTOOBIG LOD B,L	SAVE REPEAT COUNT IN B
0012DD	7C			7024	LOD A,H	GET CTRL BITS
0012DE	0F0F0F0F			7026	ROT R,4	SAVE X-Y EXCHANGED COPY
0012E2	57			7028	LOD D,A	
0012E3	7C			7030	LOD A,H	GET 'X' CTRL BITS
0012E4	E60F			7032	ANDI X'0F'	*
0012E6	325F21			7034	ST A,DIREC2	*
0012E9	B2			7036	IOR D	GET X+Y CTRL BITS
0012EA	E60F			7038	ANDI X'0F'	*
0012EC	325E21			7040	ST A,DIREC1	
0012EF	215221			7042	LODI HL,#QUADS	
0012F2	70			7044	LOD M,B	
0012F3	2A6321			7046	LD HL,XMASK	
0012F6				7048	LOOP0 DS 0X	
0012F6	AF			7050	XOR A	SET HL = $-X/2$
0012F7	91			7052	SUB C	
0012F8	C2FF12			7054	JMP NZ,DNPLT	IF ZERO, DX=DY=0; ONLY DRAW A POINT
0012FB	97			7056	SUB A	
0012FC	C34F11			7058	JMP XYPTINC	
0012FF	1F			7060	DNPLT ROT RC	
001300	57			7062	LOD D,A	
001301	06FF			7064	LODI B,255	
001303	61			7066	LOD H,C	H = DX (LOOP COUNT)
				7068 *		2959.
				7069 *	ALL THE DISTASTEFUL INITIALIZATION IS OVER, NOW	2960.
				7070 *	DO THE ZIPPY LOOP TO DRAW THE LINE	2961.
				7071 *		2962.
001304				7072	LOOP1 DS 0X	2963.
001304	7A			7074	LOD A,D	BD=BD+DY (HERE C=DX, E=DY)
001305	83			7076	ADD E	2965.
						2966.

001306 57
001307 D22713

7078
7080

L0D D, A
JMP NC, STILLNEG

2967.
2968.



LOC	OBJECT CODE	ADDR1	ADDR2	STMT	SOURCE STATEMENT	ASM H V 05 01.23 10/02/76
00130A	04			7082	INC B	(INCREMENT HIGH-ORDER PART) 2969.
00130B	C22713			7084	JMP NZ, STILLNEG	DIDN'T GO POSITIVE 2970.
00130E	91			7086	SUB C	HL=HL-DX 2971.
00130F	57			7088	LOD D,A	2972.
001310	05			7090	DEC B	(SET B=255 FOR HIGH-ORDER PART) 2973.
				7092	* INCREMENT(DECREMENT) X(Y) AND Y(X) BOTH.	2974.
001311	3A5E21			7093	LD A,DIREC1	2975.
001314	CD4F11			7095	OUTSCOPE CALL XYPTINC	2976.
001317	25			7097	DEC H	LOOP FOR DX TIMES 2977.
001318	C20413			7099	JMP NZ, LOOPI	2978.
00131B	3A5221			7101	LD A,#QUADS	2979.
00131E	D601			7103	SUBI 1	2980.
001320	C8			7105	RET Z	RETURN IF ALL WERE DONE 2981.
001321	325221			7107	ST A,#QUADS	2982.
001324	C3F612			7109	JMP LOOPO	NO - DO NEXT 2983.
				7111	* INCREMENT(DECREMENT) X(Y) ONLY	2984.
001327				7112	STILLNEG DS 0X	2985.
001327	3A5F21			7114	LD A,DIREC2	2986.
00132A	C31413			7116	JMP OUTSCOPE	(COULD SAVE JMP BY DUPL. INSTRS.) 2987.
				7118	*	2988.
				7119	* HERE IS WHERE RCM CONSTANTS GO	2989.
00132D				7120	DODAX DS 0X	2990.
00132D	80			7122	DC X'80'	2991.
00132E	40			7123	DC X'40'	2992.
00132F	20			7124	DC X'20'	2993.
001330	10			7125	DC X'10'	2994.
001331	08			7126	DC X'08'	2995.
001332	04			7127	DC X'04'	2996.
001333	02			7128	DC X'02'	2997.
001334	01			7129	DC X'01'	2998.
				7130	* END OF RCM CONSTANTS	2999.

LOC	OBJECT CODE	ADDR1	ADDR2	STMT	SOURCE STATEMENT	ASM H V 05 01.23 10/02/76
				7132 *		3001.
				7133 *	'HELP' TEXT FOR THE VGT	3002.
				7134 *		3003.
001335			01C00	7135	ORG X'1C00'	3004.
001C00	0D0A			7137	CHAR @CR,@LF	3005.
001C02	2A2A2A205374616E			7139	CHAR '*** STANFORD - SLAC VIDEO GRAPHICS TERMINAL (V17) ***'	3006.
001C0A	666F7264202D2053			7141		
001C12	4C41432020566964			7142		
001C1A	656F204772617068			7143		
001C22	696373205465726D			7144		
001C2A	696E616C20285631			7145		
001C32	3729202A2A2A			7146		
001C38	0D0A0D0A			7147	CHAR @CR,@LF,@CR,@LF	3007.
001C3C	436F6D6D616E6473			7149	CHAR 'COMMANDS ARE <CMD> PRESSED WITH: ',@DC4	3008.
001C44	20617265203C434D			7151		
001C4C	443E207072657373			7152		
001C54	656420776974683A			7153		
001C5C	14			7154		
001C5D	28205E203D203C43			7155	CHAR '(= <CTRL>) ',@CR,@LF,@CR,@LF	3009.
001C65	54524C3E2029			7157		
001C6B	0D0A0D0A			7158		
001C6F	5343524F4C4C494E			7159	CHAR 'SCROLLING: ',@DC4,'GRAPH MODE: ',@CR,@LF	3010.
001C77	473A			7161		
001C79	14			7162		
001C7A	4752415048204D4F			7163		
001C82	44453A			7164		
001C85	0D0A			7165		
001C87	3A202D2070616765			7166	CHAR ': - PAGE DOWN ',@DC4,'G - ENTER ',@CR,@LF	3011.
001C8F	20646F776E			7168		
001C94	14			7169		
001C95	67202D20656E7465			7170		
001C9D	72			7171		
001C9E	0D0A			7172		
001CA0	3B202D2070616765			7173	CHAR '; - PAGE UP ',@DC4,'N - LEAVE ',@CR,@LF	3012.
001CA8	207570			7175		
001CAB	14			7176		
001CAC	6E202D206C656176			7177		
001CB4	65			7178		
001CB5	0D0A			7179		
001CB7	5C202D206C696E65			7180	CHAR ' - LINE DOWN QUICK ',@DC4,'L - ENTER + CLR'	3013.
001CBF	20646F776E207175			7182		
001CC7	69636B			7183		
001CCA	14			7184		
001CCB	5E6C202D20456E74			7185		
001CD3	6572202B20636C72			7186		
001CDB	0D0A			7187	CHAR @CR,@LF,'@ - UP QUICK ',@DC4,'K - LEAVE + CLR ',@CR,@LF	3014.
001CDD	40202D2075702071			7189		
001CE5	7569636B			7190		
001CE9	14			7191		
001CEA	6B202D204C656176			7192		
001CF2	65202B20636C72			7193		
001CF9	0D0A			7194		
001CFB	6F202D2075702073			7195	CHAR 'O - UP SLOW ',@DC4,'Y - ENABLE ',@CR,@LF	3015.
001D03	6C6F77			7197		
001D06	14			7198		

001D07	59202D20456E6162	7199		
001D0F	6C65	7200		
001D11	0D0A	7201		
001D13	70202D20646F776E	7202	CHAR 'P - DOWN SLOW',@DC4,'N - DISABLE',@CR,@LF	3016.
001D1B	20736C6F77	7204		
001D20	14	7205		
001D21	4E202D2044697361	7206		
001D29	626C65	7207		
001D2C	0D0A	7208		
001D2E	5D202D20686F6D65	7209	CHAR ' - HOME DISPLAY TO CURSOR',@DC4,'S,M,B - SCALE'	3017.
001D36	20646973706C6179	7211		
001D3E	20746F2063757273	7212		
001D46	6F72	7213		
001D48	14	7214		
001D49	532C4D2C42202D20	7215		
001D51	5363616C65	7216		
001D56	0D0A	7217	CHAR @CR,@LF,' - UNHOME',@DC4,'TABS:',@CR,@LF	3018.
001D58	5B202D20756E686F	7219		
001D60	6D65	7220		
001D62	14	7221		
001D63	544142533A	7222		
001D68	0D0A	7223		
001D6A	6D202D2070757420	7224	CHAR 'M - PUT CURSOR ON SCREEN',@DC4,'1 - SET, 2 - CLR'	3019.
001D72	637572736F72206F	7226		
001D7A	6E2073637265656E	7227		
001D82	14	7228		
001D83	31202D205365742C	7229		
001D8B	2032202D20436C72	7230		
001D93	0D0A	7231	CHAR @CR,@LF,'CURSOR MOTION:',@DC4,'3 - CLR ALL + ENABLE'	3020.

LOC	OBJECT CODE	ADDR1	ADDR2	STMT	SOURCE STATEMENT	ASM H V 05 01.23 10/02/76
001D95	435552534F52204D			7233		
001D9D	4F54494F4E3A			7234		
001DA3	14			7235		
001DA4	33202D20436C7220			7236		
001DAC	616C6C202B20656E			7237		
001DB4	61626C65			7238		
001DB8	0D0A			7239	CHAR @CR,@LF	3021.
001DBA	3C42533E2C3C4C46			7241	CHAR '<BS>,<LF>,<HT>,<SP>,K,<CR>',@DC4	3022.
001DC2	3E2C3C48543E2C3C			7243		
001DCA	53503E2C5E6B2C3C			7244		
001DD2	43523E			7245		
001DD5	14			7246		
001DD6	34202D2053657420			7247	CHAR '4 - SET 4013 TABS',@CR,@LF,'OTHERS :',@CR,@LF	3023.
001DDE	3430313320746162			7249		
001DE6	73			7250		
001DE7	0D0A			7251		
001DE9	4F5448455253203A			7252		
001DF1	0D0A			7253		
001DF3	63202D20436C7220			7254	CHAR 'C - CLR SCREEN',@DC4,'V - INV. VIDEO',@CR,@LF	3024.
001DFB	73637265656E			7256		
001E01	14			7257		
001E02	56202D20496E762E			7258		
001E0A	20566964656F			7259		
001E10	0D0A			7260		
001E12	51202D2074797065			7261	CHAR 'Q - TYPE AHEAD',@DC4	3025.
001E1A	206168656164			7263		
001E20	14			7264		
001E21	43202D20586D6974			7265	CHAR 'C - XMIT NORMAL',@CR,@LF	3026.
001E29	206E6F726D616C			7267		
001E30	0D0A			7268		
001E32	46202D2046756C6C			7269	CHAR 'F - FULL DUPLEX',@DC4	3027.
001E3A	204475706C6578			7271		
001E41	14			7272		
001E42	48202D2048616C66			7273	CHAR 'H - HALF DUPLEX',@CR,@LF	3028.
001E4A	204475706C6578			7275		
001E51	0D0A			7276		
001E53	4C202D204C6F6361			7277	CHAR 'L - LOCAL',@DC4,'R - REMOTE',@CR,@LF	3029.
001E5B	6C			7279		
001E5C	14			7280		
001E5D	52202D2072656D6F			7281		
001E65	7465			7282		
001E67	0D0A			7283		
001E69	74202D2074696D65			7284	CHAR 'T - TIME',@DC4	3030.
001E71	14			7286		
001E72	3F202D2068656C70			7287	CHAR '? - HELP',@CR,@LF	3031.
001E7A	0D0A			7289		
001E7C	5578202D20536574			7290	CHAR 'UX - SET BAUD RATE TO X',@DC4	3032.
001E84	2042617564205261			7292		
001E8C	746520746F2078			7293		
001E93	14			7294		
001E94	2178202D20636867			7295	CHAR 'X - CHG <ESC> TO X',@CR,@LF	3033.
001E9C	203C4553433E2074			7297		
001EA4	6F2078			7298		
001EA7	0D0A			7299		
001EA9	72202D2042756620			7300	CHAR 'R - BUF RESET',@DC4	3034.
001EB1	5265736574			7302		

001EB6	14	7303		
001EB7	3C4553433E3C7878	7304	CHAR	'<ESC><XXX>=<CMD-XXX>',@CR,@LF
001EBF	783E3D3C434D442D	7306		
001EC7	7878783E	7307		
001ECB	0D0A	7308		
001ECD	58202D2031362078	7309	CHAR	'X - 16 X CLOCK',@DC4,'W - 1 X CLOCK',@CR,@LF
001ED5	20636C6F636B	7311		
001EDB	14	7312		
001EDC	57202D2031207820	7313		
001EE4	636C6F636B	7314		
001EE9	0D0A	7315		
001EEB	49202D20496E742E	7316	CHAR	'I - INT. CLOCK',@DC4,'E - EXT. CLOCK',@CR,@LF
001EF3	20636C6F636B	7318		
001EF9	14	7319		
001EFA	45202D204578742E	7320		
001F02	20636C6F636B	7321		
001F08	0D0A	7322		
001F0A	5E77202D20506174	7323	CHAR	'W - PATCH MEMORY',@DC4
001F12	6368206D656D6F72	7325		
001F1A	79	7326		
001F1B	14	7327		
001F1C	68202D2050726F63	7328	CHAR	'H - PROCESS HEX LOCATION',@CR,@LF
001F24	6573732068657820	7330		
001F2C	6C6F636174696F6E	7331		
001F34	0D0A	7332		
		7333	CHAR	'N - USE ALTERNATE CHAR SET',@DC4

3035.

3036.

3037.

3038.

3039.

3040.

LOC	OBJECT CODE	ADDR1	ADDR2	STMT	SOURCE STATEMENT	ASM H V 05 01.23 10/02/76
001F36	5E6E202D20557365			7334		
001F3E	20616C7465726E61			7335		
001F46	7465206368617220			7336		
001F4E	736574			7337		
001F51	14			7338		
001F52	5E6F202D20557365			7339	CHAR ' D - USE NORMAL CHAR SET',@CR,@LF	3041.
001F5A	206E6F726D616C20			7341		
001F62	6368617220736574			7342		
001F6A	0D0A			7343		
001F6C	5E73202D20436861			7344	CHAR ' S - CHANGE CHAR SET',@CR,@LF	3042.
001F74	6E67652063686172			7346		
001F7C	20736574			7347		
001F80	0D0A			7348		
001F82	FF			7349	DC X'FF'	3043.
				7350	END	3044.002

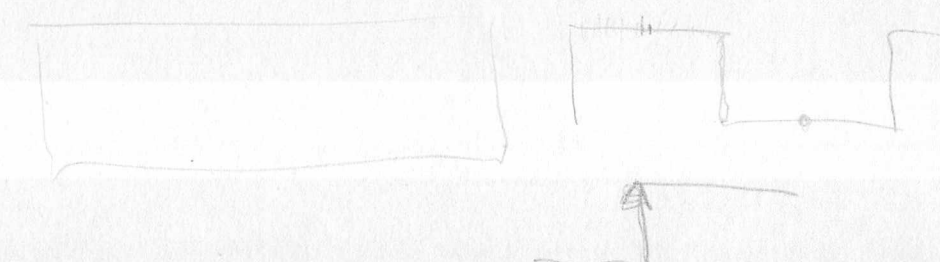
BUFINIT	00001	0008E8	4360	4180														
BYTEIN	00001	0009D2	4555	4433	4433	4433	4443	4443	4443	4453	4453	4453	4463	4463	4463	4479	4479	4479
				4511	4511	4511												
CHARHITE	00001	0000000D	1374	1916	1916	1930	1930	1949	1949	1965	1965							
CHARPOS	00001	00215C	1576	6032	6032	6032	6047	6047	6047	6128	6128	6128	6162	6162	6162	6166	6166	6166
				6170	6170	6170	6176	6176	6176	6208	6208	6208	6240	6240	6240	6260	6260	6260
				6276	6276	6276	6286	6286	6286	6290	6290	6290	6303	6303	6303	6321	6321	6321
				6335	6335	6335	6457	6457	6457	6497	6497	6497	6501	6501	6501	6531		
CHARPROC	00001	0000024C	2422	1744	3916	3916	3916	4324	4324	4324								
CHARSET	00001	0008FA	4376	4186														
CHGENRAM	00001	00003000	1179	4396	4396	4396	4412	4412	4412									
CHGENROM	00001	00002800	1177	1557	4059	4059	4059	4408	4408	4408								
CHGESC	00001	0000088F	4267	4157	4157	4157												
CHGUART	00001	00000899	4281	4161	4161	4161												
CHKATTN	00001	000127	2074	2058	2058	2058												
CHKKBRD	00001	0000FF	2036	2010	2010	2010	2024	2024	2024	2032	2032	2032						
CHLINE1	00001	00000087	1243	1850	1850													
CHL1TAB	00007	000498	3092	3025	3025	3029	3029	3039	3039	3066	3066	3070	3070	4071	4071	4071		
CIRCLEFN	00001	00077C	3988	3187	3187	3187	5229	5229	5229	5277	5277	5277	5475	5475	5475	5491	5491	5491
CKSMERR	00001	00000010	1472	4523	4523													
CLEAR	00001	00000680	3698	1747	4222	4222	4222											
CLEARIT	00001	00000689	3712	3707	3707	3707												
CLEARIT2	00001	0006A1	3740	3716	3716	3716												
CLEARLP	00001	000006DE	3807	3813	3813	3813												
CLEARLP2	00001	000006EE	3830	3838	3838	3838												
CLRATAB	00001	00000E88	5672	2671														
CLRBUS	00001	0000058D	3377	2699	3904	3904	3904											
CLRFLG	00001	002118	1449	1784	1784	1784	2805	2805	2805	2815	2815	2815	2835	2835	2835	2877	2877	2877
				3855	3855	3855	4218	4218	4218	5872	5872	5872						
CLRGRPH	00001	000003BC	2811	2759	2759	2759	2797	2797	2797									
CLRPAG	00001	00000698	3728	3736	3736	3736												
CLRSCREEN	00001	000003A6	2786	2639														
CLRSTAB	00001	00000E80	5658	2669														
CLRTABLP	00001	00000E8E	5680	5688	5688	5688												
CMDDOIT	00001	000253	2438	2369	2369	2369												
CMDKEY	00001	00000080	1296	2432	2432													
CMDTAB	00001	000002E6	2624	2440	2440	2440												
CMDTAB2	00001	00000C51	5101	5093	5093	5093												
CMDTAB3	00001	00000C85	5148	5140	5140	5140												
CMDTAB4	00001	00000FE3	6093	6087	6087	6087												
CMPLOOP	00001	00007C	1882	1888	1888	1888												
CNTLCMD	00001	002131	1521	2572	2572	2572	2916	2916	2916	4037	4037	4037	5586	5586	5586	5696	5696	5696
COPY	00001	0005AA	3416	2655	2657	2659	2663											
CPURAM	00001	00002000	1173	1381	1638													
CPURAMSZ	00001	00000400	1175	1638														
CRCNT	00001	00213B	1536	2154	2154	2154	2190	2190	2190	2243	2243	2243	2544	2544	2544			
CRLFVR	00001	002130	1519	5126	5126	5126	5132	5132	5132	5205	5205	5205						
CURCH1	00001	0000F4	2026	2020	2020	2020												
CURMOTSP	00001	000CAF	5185	5116	5371	5371	5371											
CURSCHAR	00001	002104	1422	1604	2026	2026	2026	3849	3849	3849	5079	5079	5079	5079	5333	5333	5333	5559
				5559	6028	6028	6028											
CURSCTR	00001	002103	1420	2006	2006	2006	5345	5345	5345									
CURSLOC	00002	002100	1416	2014	2014	2014	2801	2801	2801	2839	2839	2839	3159	3159	3159	3233	3233	3233
				3912	3912	3912	5083	5083	5083	5219	5219	5219	5287	5287	5287	5359	5359	5359
				5457	5457	5457	5469	5469	5469	5557	5557	5557	5567	5567	5567	6043	6043	6043
				6057	6057	6057	6626	6626	6626	6632	6632	6632						
CURSTIME	00001	00000008	1621	2012	2012	5343	5343											
CURSX	00001	002102	1418	1584	3165	3165	3165	3237	3237	3237	3853	3853	3853	5187	5187	5187	5191	5191
				5191	5201	5201	5201	5297	5297	5297	5382	5382	5382	5396	5396	5396	5445	5445
				5445	5449	5449	5449	5455	5455	5455	5522	5522	5522	5594	5594	5594	5715	5715

SYMBCL	LEN	VALUE	DEFN	REFERENCES	ASM H V 05 01.23 10/02/76														
				5715 5789 5789 5789 5801 5801 5801 5805 5805 5805 5827 5827 5827															
CUSRET	00001	0008A8	4297	4305 4305 4305															
DATE	00010	00073E	3922	3886 3886 3886															
DC1CNT	00001	00213C	1538	2200 2200 2200	2245 2245 2245	2319 2319 2319	2331 2331 2331	3370 3370 3370											
DECX	00001	0000000C	6881	6899 6899															
DECY	00001	00000003	6885	6921 6921															
DIREC1	00001	00215E	1578	7041 7041 7041	7094 7094 7094														
DIREC2	00001	00215F	1580	7035 7035 7035	7115 7115 7115														
DISADDRH	00001	00000084	1237	1877 1877															
DISADDRL	00001	00000085	1239	1873 1873															
DISLOOP	00001	000AEC	4770	4794 4794 4794															
DISPLCS	00001	002002	1386	3441 3441 3441	3468 3468 3468	4107 4107 4107	4543 4543 4543	4717 4717 4717											
				5517 5517 5531	5531 6002 6002	6070 6070 6729	6729 4543 4543	4717 4717 5516											
DISPLSP	00002	002000	1385	3434 3434 3461	3461 4099 4099	4103 4103 4103	4539 4539 4719	4719 5516 5516											
				5516 5530 5530	5530 6001 6001	6001 6069 6069	6069 6728 6728	6728 6728 6728											
DISPLST	00001	00002024	1389	4102 4102 4105	4105 4105														
DNPLT	00001	0012FF	7061	7055 7055 7055															
DNTBREAK	00001	000001BC	2262	2078 2078 2078															
DOBELL	00001	000F12	5856	5167															
DODATE	00001	000721	3890	3900 3900 3900															
DODAX	00001	00132D	7121	5596 5596 5596	6471 6471 6471														
DDHOSTCD	00001	000008B4	4307	4147 4147 4147															
DCWNCHK	00001	00043C	2974	2935 2935 2935	3033 3033 3033														
DOWNLINE	00001	000C15	5025	4935 4935 4935															
DRAGDOY	00001	001166	6553	6539 6539 6539															
DRAGEVOD	00001	00002162	6529	6557 6557 6557	6579 6579 6579														
DRAGSPOT	00001	001189	6581	6559 6559 6559															
DRAGXY	00001	0000215C	6531	6535 6535 6535	6581 6581 6581														
DRAG1	00001	0011A1	6609	6543 6543 6543															
DRAG2	00001	001165	6551	6547 6547 6547	6611 6611 6611	6615 6615 6615													
DRAG3	00001	00117D	6571	6561 6561 6561															
DRAG4	00001	001196	6591	6587 6587 6587															
DRAG5	00001	001186	6579	6565 6565 6565	6573 6573 6573														
DRAG6	00001	001185	6577	6569 6569 6569															
ENDWHEEL	00001	000000CB	1981	1903 1903 1903	1932 1932 1932	1942 1942 1942	1967 1967 1967												
ENTERGRP	00001	000003EC	2882	2713															
ENTGRPC	00001	000003E0	2861	2651															
ESCCHAR	00001	002106	1427	1802 1802 1802	4143 4143 4143	4271 4271 4271													
EVENADD	00001	000058F0	1589	1593															
EVENODD	00001	002162	1595	6214 6214 6214	6432 6432 6432	6529 6529 6529													
FLAGS	00001	002119	1452	2166 2166 2166	2570 2570 2570	2580 2580 2580	2904 2904 2904	2910 2910 2910											
				2914 2914 2914	4035 4035 4035	5580 5580 5580	5584 5584 5584	5690 5690 5690											
				5694 5694 5694	5705 5705 5705	5993 5993 5993	5997 5997 5997												
FOOADDR	00002	00212D	1505	2616 2616 2616	4065 4065 4065														
FOUNDTAB	00001	00000EED	5799	5773 5773 5773															
FRAMECNT	00001	00000008	1201	1840 1840 1865	1865														
FRAMEINT	00001	00000004	1203	1831 1831															
FULLDUPL	00001	00000080	1454	2168 2168 2906	2906 2912 2912	5995 5995													
GCHARSET	00001	00092B	4414	4398 4398 4398	4410 4410 4410														
GENLOOP	00001	001074	6216	6234 6234 6234															
GETBUF	00001	000866	4218	4109 4109 4109	4360 4360 4360														
GETCHAR	00001	0000080F	4132	4097 4097 4097	4205 4205 4205	4328 4328 4328	4348 4348 4348	4358 4358 4358											
				4364 4364 4364	4390 4390 4390	4416 4416 4416													
GETCHRCP	00001	00206F	1408	3438 3438 3438	3465 3465 3465	4100 4100 4100	4198 4198 4540	4540 4540 4540	4720 4720 4720										

SYMBOL	LEN	VALUE	DEFN	REFERENCES	ASM H V 05 01.23 10/02/76															
NOMOVE	00001	000D0C	5267	5231	5231	5231														
NOOVER	00001	000D4C	5321	5317	5317	5317														
NORCV	00001	00000004	1491	2309	2309	3306	3306	3472	3472	3478	3478	4641	4641	4643	4643					
NORMODE	00001	000003D9	2848	2723																
NCR0MCHR	00001	00000020	1251	4378	4378	4406	4406													
NOSEND	00001	00000002	1493	2480	2480	3306	3306	3321	3321	3474	3474	3478	3478	4643	4643					
NOSTICK	00001	000A2A	4619	4489	4489	4489														
NOTCRCHG	00001	0002C1	2582	2568	2568	2568														
NOTCRLF	00001	000DA5	5421	5411	5411	5411														
NOTCTL	00001	00121D	6744	6734	6734	6734														
NOTDC1	00001	00020B	2337	2317	2317	2317														
NOTHEX	00001	00000040	1468	4607	4607															
NOTICK	00001	000000DD	2003	1995	1995	1995														
NOTO0BIG	00001	0012DC	7023	6986	6986	6986														
NOTO0FAR	00001	000BFC	4997	4987	4987	4987														
NOTSAME	00001	00050E	3211	3203	3203	3203														
NOTZERO	00001	000ADB	4754	4750	4750	4750														
NOUPDATE	00001	000CD0	5219	5213	5213	5213														
NOUPDAT2	00001	000DD4	5471	5465	5465	5465														
NXTCHL1	00002	00210A	1435	1842	1842	1842	3019	3019	3019	3060	3060	3060	4073	4073	4073					
NXTDISA	00002	00210C	1437	1853	1853	1853	2799	2799	2799	2939	2939	2939	2943	2943	2943	2955	2955	2955		
				2969	2969	2969	2974	2974	2974	2988	2988	2988	3099	3099	3099	3155	3155	3155		
				3177	3177	3177	3195	3195	3195	3199	3199	3199	3205	3205	3205	3221	3221	3221		
				3227	3227	3227	3718	3718	3718	3742	3742	3742	5245	5245	5245	5267	5267	5267		
				5485	5485	5485	5563	5563	5563											
ODDADD	00001	0000AA41	1591	1593																
ONESPACE	00001	000BCF	4945	4925	4925	4925														
OUTSCOPE	00001	001314	7096	7117	7117	7117														
OUTTAB	00001	00000ED2	5763	5749	5749	5749	5785	5785	5785											
PATCHING	00001	00000001	1476	4429	4429															
PATCHIT	00001	000B48	4837	4786	4786	4786														
PATCHST	00001	0005B1	3424	2661																
PATCKSM	00001	00211D	1478	4441	4441	4441	4447	4447	4447	4451	4451	4451	4457	4457	4457	4461	4461	4461		
				4491	4491	4491	4495	4495	4495	4515	4515	4515								
PATCNT	00001	00211E	1480	4439	4439	4439	4503	4503	4503	4507	4507	4507								
PATFAIL	00001	00000008	1474	4621	4621															
PATFLAG	00001	00000020	1495	3319	3319	3321	3321	3474	3474	3478	3478									
PATSTAT	00001	00211C	1464	4330	4330	4330	4336	4336	4336	4427	4427	4427	4431	4431	4431	4521	4521	4521		
				4527	4527	4527	4605	4605	4605	4609	4609	4609	4619	4619	4619	4623	4623	4623		
				4629	4629	4629	4633	4633	4633											
PGDDOIT	00001	000546	3286	3279	3279	3279														
PGDOWN	00001	0000053D	3273	2627																
PGUDOIT	00001	000531	3257	3250	3250	3250														
PGUP	00001	00000528	3244	2629																
POINTER	00002	002127	1499	4356	4356	4356	4366	4366	4366	4372	4372	4372								
POSLOOP	00001	000091	1916	1926	1926	1926														
POS LP2	00001	0000A4	1936	1940	1940	1940														
POSRASTR	00001	0000009E	1928	1918	1918	1918														
PREVCHAR	00001	002105	1424	3393	3393	3393	4139	4139	4139	4203	4203	4203	4315	4315	4315	4812	4812	4812		
				5207	5207	5207	5265	5265	5265	5289	5289	5289	5327	5327	5327	5405	5405	5405		
PREVSCRL	00001	00212F	1516	1895	1895	1895	4057	4057	4057											
PRVCCHAR	00001	002107	1429	2046	2046	2046	2446	2446	2446	2452	2452	2452								
PUTPTR	00002	002120	1485	2214	2214	2214	2582	2582	2582	2610	2610	2610	4231	4231	4231	4237	4237	4237		
PUTREP	00001	000C00	5005	4843	4843	4843	4891	4891	4891	4951	4951	4951	4973	4973	4973	4993	4993	4993		
				5027	5027	5027	5031	5031	5031	5049	5049	5049	5053	5053	5053					

SYMBOL	LEN	VALUE	DEFN	REFERENCES															
SETLCL	00001	00000552	3302	2719															
SETNECHD	00001	00000F78	5991	5163															
SETRMT	00001	0000055B	3315	2729															
SETSIMCR	00001	0002AB	2544	2535	2535	2535													
SETTAB	00001	00000E7A	5646	2667															
SETURTX1	00001	00000F66	5959	2707															
SETURTX6	00001	00000F6F	5975	2709															
SHIFT	00001	0010F1	6350	6434	6434	6434	6487	6487	6487	6489	6489	6489	6491	6491	6491	6836	6836	6836	
				6856	6856	6856	6858	6858	6858	6862	6862	6862							
SHIFTIN	00001	000919	4400	4184															
SHIFTIN2	00001	00091D	4404	4388	4388	4388													
SHIF TMD	00001	00213D	1542	4041	4041	4041	4384	4384	4384	4394	4394	4394	4402	4402	4402	5177	5177	5177	
SHIF TOUT	00001	00090E	4392	4182															
SHOWIT	00001	000ADC	4756	4705	4705	4705	4800	4800	4800	4804	4804	4804	4965	4965	4965				
SHOWTIME	00001	0004A5	3099	2641															
SHOWT IM1	00001	0004B2	3109	3143	3143	3143													
SIMCR	00001	00016B	2154	2144	2144	2144													
SIM4023	00001	00000E03	5512	5153															
SOURCE	00002	00211A	1462	4111	4111	4111	4134	4134	4134	4352	4352	4352	4362	4362	4362				
SPACE	00001	000B85	4887	4766	4766	4766													
SPDERR	00001	0008AF	4303	4287	4287	4287													
STACK	00001	000023FF	1638	1764	1764	1764													
STAT BITS	00001	00000085	1197	1829	1829	1838	1838	1863	1863	2074	2074	2351	2351	2380	2380				
STATOUT	00001	0008C3	4330	4176															
STILLNEG	00001	001327	7113	7081	7081	7081	7085	7085	7085										
STOREIT	00001	000607	3480	3476	3476	3476													
STOR SIZE	00001	0003A2	2779	2677	2693	2701													
STPRV	00001	00000594	3387	2665	2703														
STUF XMIT	00001	000277	2487	1745															
STXMITTP	00001	0000057E	3360	3341	3341	3341	3350	3350	3350										
SUBWRP81	00001	00075B	3943	2941	2941	2941	5221	5221	5221										
SWITCH	00001	0010D8	6325	6315	6315	6315													
TABLOOP	00001	00000ED3	5767	5761	5761	5761	5777	5777	5777										
TABREF	00001	00000001	1456	5582	5582	5692	5692	5707	5707										
TABSHIFT	00001	00000ECA	5753	5759	5759	5759													
TABWRAP	00001	000EE3	5787	5719	5719	5719													
TABWRAP2	00001	00000CBC	5197	5365	5365	5365													
TEKXHI	00001	00214A	1556	6756	6756	6794	6794	6794											
TEKXLOW	00001	00214B	1558	6762	6762	6776	6776	6864	6864	6864	6864								
TEKYHI	00001	00214C	1560	6746	6746	6746													
TEKXLOW	00001	00214D	1562	6768	6768														
TESTBIG	00001	0012A9	6937	6917	6917	6917													
TESTY	00001	001298	6913	6897	6897	6897													
TEXT	00002	002167	1608	2837	2837	2837	3740	3740	3740	3867	3867	3867	3908	3908	3908	3934	3934	3934	
				3947	3947	3947	4077	4077	4077	5215	5215	5215	5319	5319	5319				
TEXTBOT	00002	00216D	1614	2996	2996	2996	3759	3759	3759	4087	4087	4087	5225	5225	5225	5233	5233	5233	
				5237	5237	5237													
TEXTTOP	00002	00216B	1612	2978	2978	2978	3181	3181	3181	3191	3191	3191	3763	3763	3763	4083	4083	4083	
				5239	5239	5239	5261	5261	5261	5471	5471	5471							
TICKTOCK	00001	0000CE	1985	2001	2001	2001													
TIME	00004	002112	1445	1780	1780	1780	1983	1983	1983	3107	3107	3107	3135	3135					
TIMELOC	00002	002116	1447	2052	2052	2052	2072	2072	2072	3105	3105	3105							
TIMERASE	00001	00011C	2062	2068	2068	2068													
TOPOFF	00001	000004BF	1372	1861	1861	1861	1869	1869	1869	6030	6030	6030	6333	6333	6333				
TXTCHAR	00001	000CA9	5177	5138	5138	5138													

ASM H V 05 01.23 10/02/76



TXTCHAR2	00001	000CA8	5175	5169															
TXTCTL	00001	000C6B	5120	5091	5091	5091													
TXTEND	00001	000F80	5999	5118	5165	5347	5347	5347	5417	5417	5417	5569	5569	5569	5858	5858	5858	5868	
				5868	5868	5905	5905	5905											
TXTMOD	00001	00000C3E	5077	4104	4104	4104	6005	6005	6005										
TYPE	00001	002160	1582	2779	2779	2779	4045	4045	4045	6824	6824	6824							
TYPERR	00001	00000020	1470	4631	4631														
UNDOIT	00001	000A4D	4649	4645	4645	4645													
UNDOSEMI	00001	000A42	4639	4273	4273	4273	4299	4299	4299	4545	4545	4545	4713	4713	4713	4830	4830	4830	
UNDUN	00001	00000A9F	4715	4835	4835	4835													
UNHOME	00001	000512	3219	2633															
UPCHK	00001	00044A	2988	2961	2961	2961	3056	3056	3056										
UPLINE	00001	000C29	5047	4939	4939	4939													
URTBREAK	00001	0000000F	1275	2082	2082														
URTCLOCK	00001	0000000C	1253	5933	5933	5948	5948												
URTCTL	00001	00000041	1267	1768	1768	1772	1772	2084	2084	2274	2274	2301	2301	5963	5963	5967	5967	5979	
				5979	5983	5983													
URTIINTRS	00001	00000057	1269	5961	5961	5977	5977												
URTMODE	00001	0000007A	1271	5981	5981														
URTRCV	00001	00000001	1219	2311	2311														
URTRERR	00001	00000038	1215	2293	2293														
URTRSBRK	00001	00000007	1277	2272	2272														
URTRSERR	00001	00000017	1279	1770	1770	2299	2299												
URTRXRDY	00001	00000002	1213	2287	2287														
URTSPEED	00001	0000008E	1283	4051	4051	4295	4295												
URTSTAT	00001	00000041	1209	2090	2090	2283	2283												
URTTXRDY	00001	00000001	1211	2092	2092														
URTXMT	00001	00000001	1281	2140	2140														
URTX1MD	00001	00000079	1273	1766	1766	5965	5965												
VECTOR	00001	001287	6877	1743	6701	6701	6701												
VGT		****UNDEFINED****		5106															
VGT17	00001	00000000	1110	1381	1413	1413	1655	1762	1774	1774	1774	1774	1776	1776	1776	1780	1780	1780	1784
				1784	1784	1786	1786	1786	1788	1788	1788	1788	1792	1792	1792	1794	1794	1794	1802
				1802	1802	1804	1804	1804	1809	1813	1813	1813	1813	1833	1833	1833	1842	1842	1842
				1846	1846	1846	1853	1853	1853	1855	1855	1855	1855	1859	1859	1859	1867	1867	1867
				1888	1888	1888	1895	1895	1895	1903	1903	1903	1905	1905	1905	1918	1918	1918	1918
				1922	1922	1922	1926	1926	1926	1932	1932	1932	1936	1936	1936	1940	1940	1940	1940
				1942	1942	1942	1951	1951	1951	1953	1953	1953	1957	1957	1957	1961	1961	1961	1961
				1967	1967	1967	1971	1971	1971	1975	1975	1975	1983	1983	1983	1995	1995	1995	1995
				2001	2001	2001	2006	2006	2006	2010	2010	2010	2014	2014	2014	2020	2020	2020	2020
				2024	2024	2024	2026	2026	2026	2032	2032	2032	2040	2040	2040	2044	2044	2044	2044
				2046	2046	2046	2050	2050	2050	2052	2052	2052	2058	2058	2058	2068	2068	2068	2068
				2072	2072	2072	2078	2078	2078	2080	2080	2080	2094	2094	2094	2096	2096	2096	2096
				2108	2108	2108	2110	2110	2110	2114	2114	2114	2116	2116	2116	2120	2120	2120	2120
				2126	2130	2130	2130	2132	2132	2132	2136	2136	2136	2144	2144	2144	2152	2152	2152
				2152	2154	2154	2154	2166	2166	2166	2170	2170	2170	2190	2190	2190	2194	2194	2194
				2194	2196	2196	2196	2200	2200	2200	2204	2204	2204	2206	2206	2206	2212	2212	2212
				2212	2214	2214	2214	2216	2216	2216	2220	2220	2220	2233	2233	2233	2235	2235	2235
				2235	2237	2237	2237	2239	2239	2239	2243	2243	2243	2245	2245	2245	2249	2249	2249
				2249	2264	2264	2264	2270	2270	2270	2276	2276	2276	2289	2289	2289	2295	2295	2295
				2295	2307	2307	2307	2313	2313	2313	2317	2317	2317	2319	2319	2319	2323	2323	2323
				2323	2325	2325	2325	2327	2327	2327	2331	2331	2331	2333	2333	2333	2335	2335	2335
				2335	2339	2339	2339	2341	2341	2341	2343	2343	2343	2359	2359	2359	2363	2363	2363
				2363	2365	2365	2365	2369	2369	2369	2373	2373	2373	2378	2378	2378	2384	2384	2384
				2384	2398	2398	2398	2404	2404	2404	2434	2434	2434	2440	2440	2440	2442	2442	2442
				2442	2446	2446	2446	2452	2452	2452	2472	2472	2472	2476	2476	2476	2478	2478	2478
				2478	2482	2482	2482	2487	2487	2487	2493	2493	2493	2495	2495	2495	2499	2499	2499
				2501	2501	2505	2505	2505	2507	2507	2507	2511	2511	2511	2513	2513	2513	2513	2517
				2517	2517	2523	2523	2523	2535	2535	2535	2544	2544	2544	2544	2544	2548	2548	2548
				2570	2570	2572	2572	2572	2580	2580	2580	2582	2582	2582	2582	2590	2590	2590	2592

SYMBOL	LEN	VALUE	DEFN	REFERENCES	ASM	H	V	05	01.23	10/02/76
2592	2592	2594	2594	2594	2598	2598	2598	2600	2600	2600
2610	2610	2616	2616	2616	2625	2627	2629	2631	2633	2635
2645	2647	2649	2651	2653	2655	2657	2659	2661	2663	2665
2675	2677	2679	2681	2683	2685	2687	2689	2691	2693	2695
2705	2707	2709	2711	2713	2715	2717	2719	2721	2723	2725
2753	2753	2757	2757	2757	2759	2759	2759	2763	2763	2763
2779	2779	2793	2793	2793	2797	2797	2797	2799	2799	2799
2805	2805	2815	2815	2815	2819	2819	2819	2835	2835	2835
2839	2839	2841	2841	2841	2856	2856	2856	2869	2869	2869
2888	2888	2896	2896	2896	2904	2904	2904	2908	2908	2908
2914	2914	2916	2916	2916	2924	2924	2924	2932	2932	2932
2939	2939	2941	2941	2941	2943	2943	2943	2950	2950	2950
2961	2961	2965	2965	2965	2967	2967	2967	2969	2969	2969
2978	2978	2988	2988	2988	2992	2992	2992	2996	2996	2996
3017	3017	3019	3019	3019	3025	3025	3029	3029	3033	3033
3039	3039	3046	3046	3046	3054	3054	3054	3056	3056	3056
3066	3070	3070	3072	3072	3072	3099	3099	3099	3105	3105
3135	3135	3143	3143	3143	3155	3155	3155	3159	3159	3159
3177	3177	3181	3181	3181	3187	3187	3187	3189	3189	3189
3193	3193	3195	3195	3195	3199	3199	3199	3203	3203	3203
3211	3211	3219	3219	3219	3221	3221	3221	3227	3227	3227
3237	3237	3248	3248	3248	3250	3250	3250	3261	3261	3261
3277	3277	3279	3279	3279	3290	3290	3290	3294	3294	3294
3308	3308	3317	3317	3317	3323	3323	3323	3327	3327	3327
3341	3341	3346	3346	3346	3350	3350	3350	3356	3356	3356
3364	3364	3366	3366	3366	3370	3370	3370	3379	3379	3379
3393	3393	3395	3395	3395	3406	3406	3406	3412	3412	3412
3428	3429	3429	3429	3430	3430	3430	3431	3431	3431	3432
3434	3434	3435	3435	3435	3437	3437	3438	3438	3438	3440
3443	3443	3443	3447	3447	3447	3455	3455	3456	3456	3456
3458	3458	3459	3459	3460	3460	3460	3461	3461	3462	3462
3465	3465	3467	3467	3468	3468	3468	3470	3470	3470	3476
3480	3517	3517	3517	3521	3521	3521	3525	3525	3525	3529
3533	3537	3537	3537	3578	3578	3578	3582	3582	3582	3603
3609	3617	3617	3617	3623	3623	3623	3635	3635	3635	3707
3716	3718	3718	3718	3736	3736	3736	3738	3738	3738	3740
3742	3747	3747	3747	3757	3757	3757	3759	3759	3759	3761
3763	3765	3765	3765	3775	3775	3775	3813	3813	3813	3816
3824	3838	3838	3838	3843	3843	3843	3849	3849	3849	3853
3855	3867	3867	3867	3880	3880	3880	3886	3886	3886	3900
3904	3908	3908	3908	3912	3912	3912	3916	3916	3916	3934
3947	3957	3957	3957	3981	3981	3981	3983	3983	3983	3985
3997	4035	4035	4035	4037	4037	4037	4039	4039	4039	4041
4043	4045	4045	4045	4057	4057	4057	4061	4061	4061	4063
4065	4067	4067	4067	4069	4069	4069	4071	4071	4071	4073
4077	4081	4081	4081	4083	4083	4083	4087	4087	4087	4089
4093	4095	4095	4096	4096	4096	4097	4097	4097	4098	4098
4100	4100	4102	4102	4103	4103	4103	4104	4104	4104	4105
4107	4107	4107	4109	4109	4109	4111	4111	4111	4113	4113
4135	4135	4135	4139	4139	4139	4143	4143	4143	4147	4147
4157	4157	4157	4161	4161	4161	4167	4167	4167	4169	4169
4176	4178	4180	4182	4184	4186	4188	4197	4197	4197	4198
4203	4203	4203	4205	4205	4205	4218	4218	4218	4222	4222
4231	4231	4231	4235	4235	4235	4237	4237	4237	4241	4241
4251	4251	4251	4253	4253	4253	4271	4271	4271	4273	4273

4285	4285	4285	4287	4287	4287	4299	4299	4299	4301	4301	4301	4305	4305	4305
4315	4315	4315	4324	4324	4324	4328	4328	4328	4330	4330	4330	4332	4332	4332
4336	4336	4336	4344	4344	4344	4348	4348	4348	4350	4350	4350	4352	4352	4352
4356	4356	4356	4358	4358	4358	4360	4360	4360	4362	4362	4362	4364	4364	4364
4366	4366	4366	4372	4372	4372	4376	4376	4376	4382	4382	4382	4384	4384	4384
4388	4388	4388	4390	4390	4390	4394	4394	4394	4398	4398	4398	4402	4402	4402
4404	4404	4404	4410	4410	4410	4414	4414	4414	4416	4416	4416	4427	4427	4427
4431	4431	4431	4433	4433	4433	4437	4437	4437	4439	4439	4439	4441	4441	4441
4443	4443	4443	4447	4447	4447	4451	4451	4451	4453	4453	4453	4457	4457	4457
4461	4461	4461	4463	4463	4463	4467	4467	4467	4473	4473	4473	4474	4474	4475
4475	4475	4479	4479	4479	4489	4489	4489	4489	4491	4491	4491	4495	4495	4495
4503	4503	4507	4507	4507	4509	4509	4509	4509	4511	4511	4511	4515	4515	4515
4519	4519	4521	4521	4521	4527	4527	4527	4527	4535	4535	4535	4536	4536	4537
4537	4539	4539	4540	4540	4540	4542	4542	4542	4543	4543	4543	4545	4545	4551
4551	4551	4552	4552	4553	4553	4553	4559	4559	4559	4560	4560	4561	4561	4561
4565	4565	4565	4566	4566	4567	4567	4567	4567	4571	4571	4571	4573	4573	4583
4583	4583	4584	4584	4585	4585	4585	4589	4589	4589	4590	4590	4591	4591	4591
4595	4595	4595	4597	4597	4597	4605	4605	4605	4609	4609	4609	4617	4617	4617
4619	4619	4619	4623	4623	4623	4629	4629	4629	4633	4633	4633	4637	4637	4637
4639	4639	4639	4645	4645	4645	4649	4649	4649	4665	4665	4665	4669	4669	4669
4675	4675	4675	4676	4676	4677	4677	4677	4681	4681	4681	4682	4682	4683	4683
4683	4685	4685	4685	4685	4689	4689	4689	4693	4693	4693	4694	4694	4695	4695
4701	4701	4701	4705	4705	4705	4709	4709	4709	4713	4713	4713	4716	4716	4717
4717	4717	4719	4719	4720	4720	4720	4724	4724	4724	4724	4725	4725	4726	4726
4730	4730	4730	4731	4731	4732	4732	4732	4732	4738	4738	4738	4739	4739	4740
4740	4744	4744	4744	4744	4746	4746	4746	4750	4750	4750	4756	4756	4756	4760
4760	4764	4764	4764	4766	4766	4766	4774	4774	4774	4774	4775	4775	4776	4776
4780	4780	4780	4786	4786	4786	4788	4788	4788	4794	4794	4794	4800	4800	4800
4804	4804	4804	4806	4806	4806	4808	4808	4808	4812	4812	4812	4816	4816	4816
4817	4817	4818	4818	4818	4824	4824	4824	4824	4825	4825	4826	4826	4830	4830
4830	4832	4832	4832	4833	4833	4833	4835	4835	4835	4843	4843	4843	4847	4847
4847	4848	4848	4849	4849	4849	4851	4851	4851	4853	4853	4853	4861	4861	4861
4862	4862	4863	4863	4863	4865	4865	4865	4867	4867	4867	4877	4877	4877	4891
4891	4891	4899	4899	4899	4903	4903	4903	4904	4904	4905	4905	4911	4911	4911
4911	4915	4915	4915	4916	4916	4917	4917	4917	4925	4925	4925	4927	4927	4927
4931	4931	4931	4935	4935	4935	4939	4939	4939	4951	4951	4951	4965	4965	4965
4973	4973	4973	4987	4987	4987	4993	4993	4993	5013	5013	5013	5014	5014	5015
5015	5015	5019	5019	5019	5027	5027	5027	5031	5031	5031	5049	5049	5049	5053
5053	5053	5079	5079	5079	5083	5083	5083	5091	5091	5091	5093	5093	5093	5095
5095	5095	5102	5104	5108	5110	5112	5114	5116	5118	5122	5122	5122	5126	5126
5126	5132	5132	5132	5138	5138	5138	5140	5140	5140	5142	5142	5142	5149	5151
5153	5155	5157	5159	5161	5163	5165	5167	5169	5177	5177	5177	5187	5187	5187
5191	5191	5191	5195	5195	5195	5201	5201	5201	5205	5205	5205	5207	5207	5207
5213	5213	5213	5215	5215	5215	5219	5219	5219	5221	5221	5221	5225	5225	5225
5229	5229	5229	5231	5231	5231	5233	5233	5233	5235	5235	5235	5237	5237	5237
5239	5239	5239	5245	5245	5245	5253	5253	5253	5259	5259	5259	5261	5261	5261
5265	5265	5265	5267	5267	5267	5271	5271	5271	5273	5273	5273	5277	5277	5277
5279	5279	5279	5283	5283	5283	5287	5287	5287	5289	5289	5289	5293	5293	5293
5297	5297	5297	5307	5307	5307	5317	5317	5317	5319	5319	5319	5323	5323	5323
5327	5327	5327	5333	5333	5333	5339	5339	5339	5345	5345	5345	5347	5347	5347
5355	5355	5355	5359	5359	5359	5365	5365	5365	5367	5367	5367	5371	5371	5371
5382	5382	5382	5396	5396	5396	5398	5398	5398	5405	5405	5405	5411	5411	5411
5417	5417	5417	5427	5427	5427	5445	5445	5445	5449	5449	5449	5451	5451	5451
5455	5455	5455	5457	5457	5457	5461	5461	5461	5465	5465	5465	5469	5469	5469
5471	5471	5471	5475	5475	5475	5477	5477	5477	5481	5481	5481	5483	5483	5483
5485	5485	5485	5487	5487	5487	5491	5491	5491	5493	5493	5493	5497	5497	5497
5506	5506	5506	5516	5516	5516	5517	5517	5518	5518	5518	5522	5522	5522	5530
5530	5530	5531	5531	5532	5532	5532	5532	5532	5557	5557	5559	5559	5563	5563
5563	5565	5565	5565	5567	5567	5567	5567	5569	5569	5569	5580	5580	5584	5584
5584	5586	5586	5586	5594	5594	5594	5594	5596	5596	5596	5612	5612	5622	5622

ASM H V 05 01.23 10/02/76

THE FOLLOWING STATEMENTS WERE FLAGGED

05106

1 STATEMENT FLAGGED IN THIS ASSEMBLY

8 WAS HIGHEST SEVERITY CODE

OVERRIDING PARAMETERS- TERM,LINECOUNT(115)

OPTIONS FOR THIS ASSEMBLY

NODECK, OBJECT, LIST, XREF(SHORT), NORENT, NOTEST, BATCH, ALIGN, ESD, RLD, TERM, LINECOUNT(15),

FLAG(0), SYSPARM()

NO OVERRIDING DD NAMES

3423 CARDS FROM SYSIN

1061 CARDS FROM SYSLIB

8139 LINES OUTPUT

111 CARDS OUTPUT

ASP JOB NO. = 7779

SATURDAY OCTOBER 02, 1976 (76.276)

INPUT STATEMENTS (INCLUDING DD *) = 003448

//LJSCG638 JOB LJS\$CG,TIME=(1,30),CLASS=E

0.002

ELAPSED TIME ON MAIN = SYB (A8) = 003.70, START TIME = 01.22.51

DDNAME = SYSMSG

PRINTED ON PRT4

, LINES = 000153

DDNAME = ASM.MCS80.SYSTEM

PRINTED ON PRT4

, LINES = 000010

DDNAME = CLEANUP.OUT

PRINTED ON PRT4

, LINES = 004686

LINES OUTPUT FOR THIS JOB = 004849

CARDS FROM MAIN FOR THIS JOB = NONE
