

LOC	OBJECT CODE	ADDR1	ADDR2	STMT	SOURCE STATEMENT	ASM H V 05 18.56 09/03/76
				2	PRINT OFF SUPPRESS LISTING OF MACROS	1.002
				1037	PRINT ON,GEN,NODATA	1.006
				1038	GBLA &@(256)	1.008
000000				1039	MCS80 CSECT	1.01
				1040	ASCII , DEFINE ASCII CHARACTER SET	1.012
000000				1077	PONG CSECT	2.
				1078	*DEFINITIONS	3.
				1079	*PROGRAM BY EDWARD FRANK	3.1
				1080	*CCPYRIGHT 1976	3.3
				1081	*	3.4
				1082	* EHF	4.
				1083	*	5.
				1084	*INCLUDE #VGTDEFNS USER VGT GRO CG ON CAT	6.
				1085	*****	6.001
				1086	*	6.002
				1087	* THE VGT - VIDEO GRAPHICS TERMINAL	6.003
				1088	*	6.004
				1089	* **** HARDWARE DEFINITIONS ****	6.005
				1090	*	6.006
				1091	*	6.007
				1092	*****	6.008
				1093	* UPDATE LIST 23 MAY 76 LJS	6.009
				1094	* 3 AUG 76 MRD	6.01
				1095	* 5 AUG 76 LJS	6.011
				1096	* 30 AUG 76 EHF (ADDED MODESET2)	6.012
				1097	*	6.013
				1098	* MEMORY MAP	6.014
				1099	*	6.015
00400				1100	K EQU 1024	6.016
00000				1102	ROM EQU 0*K ROMS	6.017
02000				1104	CPURAM EQU 8*K CPU SCRATCHPAD RAM	6.018
00400				1106	CPURAMSZ EQU 1*K SIZE OF IT	6.019
02800				1108	CHGENROM EQU 10*K STANDARD CHARACTER GENERATOR	6.02
03000				1110	CHGENRAM EQU 12*K WRITABLE CHARACTER GENERATOR	6.021
04000				1112	RAM EQU 16*K VIDEO RAM	6.022
04400				1114	WRAPADDR EQU 17*K WHERE DISPLAY HARDWARE WRAPS TO	6.023
				1116	*	6.024
				1117	*	6.025
				1118	* I/O SYMBOLS MARKED '(PORT)' ARE I/O PORTS,	6.026
				1119	* OTHERS ARE BIT-WITHIN-PORT DEFINITIONS.	6.027
				1120	*	6.028
				1121	*	6.029
				1122	* INPUT PORTS	6.03
				1123	*	6.031
00084				1124	KEYBOARD EQU X'84' (PORT) KEYBOARD CHARACTER, X'80' IS STROBE	6.032
00085				1126	STATBITS EQU X'85' (PORT) STATUS BITS:	6.033
00010				1128	KBNEWCHR EQU X'10' NEW KB CHAR (IE STROBE CHANGED)	6.034
00008				1130	FRAMECNT EQU X'08' EVEN/ODD FRAME BIT	6.035
00004				1132	FRAMEINT EQU X'04' END-OF-FRAME INTERRUPT	6.036
00002				1134	KBATTN EQU X'02' KEYBOARD 'ATTN' KEY PRESSED	6.037
00001				1136	KBRPT EQU X'01' KEYBOARD 'REPEAT' KEY PRESSED	6.038
00041				1138	URTSTAT EQU X'41' (PORT) USART STATUS	6.039
00001				1140	URTTXRDY EQU X'01' XMIT READY BIT	6.04
00002				1142	URTRXRDY EQU X'02' RCVR CHAR READY	6.041
00038				1144	URTRERR EQU B'00111000' RECVR ERRORS. PUNT.	6.042

PONG

PONG

00020	1146	URTREVBK	EQU	B'00100000'
00061	1148	URTRCV	EQU	X'01' (PORT)
00086	1150	ATODVAL	EQU	X'86' (PORT)
00087	1152	PARIN	EQU	X'87' (PORT)
00001	1154	PRTREADY	EQU	X'01'
00083	1156	SCRWHEEL	EQU	X'83' (PORT)

REVERSE BREAK RECIEVED	6.043
USART RECEIVED CHARACTER	6.044
VALUE OF LAST A/D CONVERSION	6.045
PARALLEL DATA INPUT	6.046
VERSATEC PRINTER READY BIT	6.047
SCROLLING WHEEL POSITION	6.048

LOC	OBJECT CODE	ADDR1	ADDR2	STMT	SOURCE STATEMENT	ASM H V 05 18.56 09/03/76
				1159 *		6.05
				1160 *	OUTPUT PORTS	6.051
				1161 *		6.052
00082	1162	RSFRMINT	EQU	X'82'	(PORT) RESET FRAME INTERRUPT	6.053
00083	1164	RSURTINT	EQU	X'83'	(PORT) RESET USART INTERRUPT	6.054
00084	1166	DISADDRH	EQU	X'84'	(PORT) HIGH-ORDER DISPLAY ADDRESS	6.055
00085	1168	DISADDRL	EQU	X'85'	(PORT) LOW-ORDER DISPLAY ADDRESS	6.056
00086	1170	BELL	EQU	X'86'	(PORT) BEEPER	6.057
00087	1172	CHLINE1	EQU	X'87'	(PORT) 1ST LINE OF ROW 1 TO DISPLAY	6.058
00080	1174	MODESET	EQU	X'80'	(PORT) CONTROL BITS:	6.059
00080	1176	REVRSVID	EQU	X'80'	REVERSE VIDEO CONTROL	6.06
00040	1178	SCRNBLNK	EQU	X'40'	SCREEN BLANKING CONTROL	6.061
00020	1180	NORCMCHR	EQU	X'20'	NO ROM CHARS (RAM ONLY) IN TEXT MODE	6.062
0000C	1182	URTCLOCK	EQU	X'0C'	USART EXTERNAL/INTERNAL CLOCK CONTRL	6.063
00010	1184	GRAPHMD	EQU	X'10'	GRAPH MODE	6.064
00002	1186	QUICKMD	EQU	X'02'	QUICK MODE FOR RAM ACCESS	6.065
00001	1188	SIXTNMD	EQU	X'01'	16 RASTERS/ROW MODE	6.066
000A6	1190	MODESET2	EQU	X'A6'	(PORT) MORE MODEBITS	6.067
00080	1192	HIGHWRAP	EQU	X'80'	WRAP TO X'7600'	6.068
00040	1194	ATODSTR	EQU	X'40'	START A/D CONVERSION (20 USEC.)	6.069
00041	1196	URTCTL	EQU	X'41'	(PORT) USART CONTROL BITS:	6.07
00057	1198	URTINTRS	EQU	B'01010111'	INTERNAL RESET (TO SET MODE)	6.071
0007A	1200	URTMODE	EQU	B'01111010'	ASYNCR, EVEN PARITY, 7BITS, 16XCLK	6.072
00079	1202	URTX1MD	EQU	B'01111001'	ASYNCR, EVEN PARITY, 7BITS, 1XCLK<--	6.073
0000F	1204	URTBREAK	EQU	B'00001111'	SEND BREAK	6.074
00007	1206	URTRSERR	EQU	B'00000111'	RESET BREAK	6.075
00017	1208	URTRSERR	EQU	B'00010111'	RESET RCV ERR, RCV ENB, DTR, XMT ENB	6.076
00001	1210	URTXMT	EQU	X'01'	(PORT) USART TRANSMITTED CHARACTER	6.077
0008E	1212	URTSPEED	EQU	X'8E'	(PORT) USART BAUD RATE; 4 BITS RCV, 4 XMIT	6.078
00096	1214	KBRESET	EQU	X'96'	(PORT) RESET KB STROBE FF (IE 'KBNEWCHR')	6.079
0009F	1216	ATODSEL	EQU	X'9E'	(PORT) SELECT ANALOG SOURCE BY 1 OF A3-A0	6.08
000AE	1218	KBCLICK	EQU	X'AE'	(PORT) KEYBOARD CLICK	6.081
000B6	1220	PAROUT	EQU	X'B6'	(PORT) PARALLEL DATA OUT	6.082
000BE	1222	PARRESET	EQU	X'BE'	(PORT) PARALLEL OUTPUT RESET	6.083
	1224	*INCLUDE			VGTRAMAP USER VGT GRO CG ON CAT	7.
00080	1225	CMDKEY	EQU	X'80'		7.001

LOC	OBJECT CODE	ADDR1	ADDR2	STMT	SOURCE STATEMENT	ASM H V 05 18.56 09/03/76
1228					*****	7.003
1229					* HERE ARE THE COROUTINE MACRO DEFINITIONS	7.004
1230					*****	7.005
1231					MACRO	7.006
1232	&L				CCB &STACKSZ=32	7.007
1233					GBLA &CCBLEN	7.008
1234					AIF (K'&L GT 6),ERR LENGTH OF NAME<7	7.009
1235	&L				DS 0X	7.01
1236	&L,SP				DS 2 THIS STACKPTR IS 2 BECAUSE WE DO A	7.011
1237	.*				DOUBLE STORE OF HL TO &L,SP BELOW	7.012
1238	&L,CS				DS 1 POINTER TO CHARACTER SOURCE CCB	7.013
1239	&L,CP				DS 1 POINTER TO CHARACTER PROC. CCB	7.014
1240					DS &STACKSZ HERE IS THE STACK	7.015
1241	&L,ST				DS 0 THE END OF IT	7.016
1242	&CCBLEN				SETA &CCBLEN+&STACKSZ+4 INCREMENT COUNTER	7.017
1243					AIF (&CCBLEN LT 256).OK AND MAKE SURE WE STAY ON ONE PAGE	7.018
1244					MNOTE 8,'***ERROR*** CONTROL BLOCKS EXCEED 255'	7.019
1245	.OK				MEXIT	7.02
1246	.ERR				MNOTE 8,'***ERROR*** COROUTINE NAME TOO LONG'	7.021
1247					MEND	7.022
1249					MACRO	7.024
1250	&LABEL				CCBSET &CCB,&EP=,&CS=,&CP=	7.025
1251					AIF ('&LABEL' EQ '').SKIP	7.026
1252	&LABEL				EQU *	7.027
1253	.SKIP				ANOP	7.028
1254					AIF ('&EP' EQ '').A1	7.029
1255					LODI A,&CCB,ST-6,> SET THE INITIAL ENTRY POINT	7.03
1256					ST A,&CCB,SP PUT IT IN THE STACK POINTER	7.031
1257					LODI HL,&EP SET THE INITIAL ENTRY POINT	7.032
1258					ST HL,&CCB,ST-2	7.033
1259	.A1				AIF ('&CS' EQ '').A2	7.034
1260					LODI A,&CS,SP,>	7.035
1261					ST A,&CCB,CS STORE CHARACTER SOURCE CCB LOCATION	7.036
1262	.A2				AIF ('&CP' EQ '').A3	7.037
1263					LODI A,&CP,SP,>	7.038
1264					ST A,&CCB,CP SIMILARLY FOR CHARACTER PROC.	7.039
1265	.A3				ANOP	7.04
1266					MEND	7.041
1268					MACRO	7.043
1269	&LABEL				RESUME &CURTASK,&NEWTASK	7.044
1270	&LABEL				LODI HL,-6	7.045
1271					ADD HL,SP PREPARE TO STORE BC,DE	7.046
1272					ST HL,&CURTASK,SP	7.047
1273					LODI L,&CURTASK,&NEWTASK,> HL NOW POINTS AT &CCB,CP OR CS	7.048
1274					CALL RESUMER CALL GLOBAL ROUTINE	7.049
1275					MEND	7.05
1277					MACRO	7.052
1278	&LABEL				CALL@ &TASK THIS IS FOR CALL INDIRECTS	7.053
1279	&LABEL				LD HL,&TASK GET THE ROUTINE ADDRESS	7.054
1280					CALL JUMPER AND CALL GLOBAL ROUTINE	7.055
1281					MEND	7.056

LOC	OBJECT CODE	ADDR1	ADDR2	STMT	SOURCE STATEMENT	ASM H V 05 18.56 09/03/76
				1283	*****	7.058
				1284	*	7.059
				1285	ASSEMBLY CONSTANTS AND RAM LOCATIONS	7.06
				1286	*	7.061
				1287	*****	7.062
				1288	*	7.063
	04400			1289	ITEXT EQU X'4400'	7.064
	00051			1291	LINESIZE EQU 81	7.065
	00025			1293	NLINES EQU 37	7.066
	008B5			1295	SCRNSIZE EQU LINESIZE*NLINES	7.067
	058F0			1297	GRAPHEVN EQU X'10000'-(512+15+1)*LINESIZE	7.068
	0AA41			1299	GRAPHODD EQU GRAPHEVN+257*LINESIZE	7.069
	004BF			1301	TOPOFF EQU 15*LINESIZE	7.07
	0000D			1303	CHARHITE EQU 13	7.071
	0BC00			1305	ITXTSIZE EQU X'10000'-ITEXT	7.072
	0000E			1307	IWASTED EQU ITXTSIZE-ITXTSIZE/LINESIZE	7.073
				1309	*	7.074
000000		02000		1310	ORG CPURAM	7.075
				1312	*	7.076
002000				1313	DISPL CCB	7.077
002000				1315		
002002				1316		
002003				1317		
002004				1318		
		02024		1319		
002024				1320	LOADER CCB	7.078
002024				1322		
002026				1323		
002027				1324		
002028				1325		
		02048		1326		
002048				1327	HEXER CCB	7.079
002048				1329		
00204A				1330		
00204B				1331		
00204C				1332		
		0206C		1333		
00206C				1334	GETCHR CCB	7.08
00206C				1336		
00206E				1337		
00206F				1338		
002070				1339		
		02090		1340		
				1341	*	7.081
		00070		1342	ALIGN 0,256,FILL=	7.082
				1344		
002090				1345	CURSLOC DS 2	7.083
002100				1347	CURSX DS 1	7.084
002102				1349	CURSCTR DS 1	7.085
002103				1351	CURSCTR DS 1	7.086
002104				1353	PREVCHAR DS 1	7.087
002105				1355	*	7.088
002106				1356	ESCCHAR DS 1	7.089
002107				1358	PRVCCHAR DS 1	7.09
002108				1360	KEYCTR DS 1	7.091

002109		1362	KEYSAVED	DS	1	CHAR TO BE REPEATED	7.092
00210A		1364	NXTCHL1	DS	2	ADDR OF NEXT CHL1TAB ENTRY TO USE	7.093
00210C		1366	NXTDISA	DS	2	NEXT DISPLAY ADDR TO BE LOADED	7.094
00210E		1368	SAVDISA	DS	2	SAVED DISPLAY ADDRESS AT "HOME" TIME	7.095
002110		1370	SCRCTR	DS	1	SCROLL COUNTER	7.096
002111		1372	BRKSTATE	DS	1	IS BREAK BEING ISSUED	7.097
002112		1374	TIME	DS	4	HHMMSSTT, IN DECIMAL	7.098
002116		1376	TIMELOC	DS	2	LOCATION OF TIME MSG	7.099
002118		1378	CLRFLG	DS	1	MSB=1 FOR CLR NEEDED, LSB FOR BANNER	7.1
		1380	*			X'C0' FOR CLR TO ZERO, NOT BLANK	7.101
002119		1381	FLAGS	DS	1	MISCELLANEOUS FLAGS	7.102
	00080	1383	FULLDUPL	EQU	X'80'	FULL DUPLEX MODE	7.103
	00001	1385	TABREF	EQU	X'01'	HAVE TABS BEEN CLEAR?	7.104
		1387	*				7.105
		1388	* COROUTINE CONTROL BLOCKS...				7.106
		1389	*				7.107
		1390	*				7.108
00211A		1391	SOURCE	DS	2	CURRENT CHARACTER SOURCE	7.109
00211C		1393	PATSTAT	DS	1	PATCHING STATUS	7.11
	00080	1395	LOUSYPAR	EQU	X'80'	PARITY N3 GOOD	7.111
		1397	NOTHEX	EQU	X'40'	NONHEX CHAR RECVD	7.112

LOC	OBJECT CODE	ADDR1	ADDR2	STMT	SOURCE STATEMENT	ASM H V 05 18.56 09/03/76
			00040	1398		
			00020	1399	TYPERR EQU X'20'	BAD PATCH TYPE 7.113
			00010	1401	CKSMERR EQU X'10'	CHECKSUM BAD 7.114
			00008	1403	PATFAIL EQU X'08'	PATCH DID NOT STICK 7.115
			00001	1405	PATCHING EQU X'01'	INDICATES PATCHING BEING DONE 7.116
00211D				1407	PATCKSM DS 1	CHECKSUM 7.117
00211E				1409	PATCNT DS 1	COUNTER 7.118
				1411	*	7.119
00211F				1412	MODEBITS DS 1	CURRENT MODE BITS (SEE "MODESET") 7.12
002120				1414	PUTPTR DS 2	NEXT FREE BUFFER LOCATIN 7.121
002122				1416	GETPTR DS 2	NEXT BUFFERED CHAR TO PRCESS 7.122
002124				1418	LCLMODE DS 1	SEND TO COMPUTER=0,NOT=ANYTHING 7.123
		00004		1420	NORCV EQU X'04'	7.124
		00002		1422	NOSEND EQU X'02'	7.125
		00020		1424	PATFLAG EQU X'20'	7.126
002125				1426	SAVSTK DS 2	7.127
002127				1428	POINTER DS 2	CURRENT CHARACTER TO BE SENT 7.128
002129				1430	HEXADDR DS 2	7.129
00212B				1432	INTROUT DS 2	ADDRESS OF INTERRUPT HANDLER 7.13
00212D				1434	FOOADDR DS 2	HANDLER FOR UNDEF. COMMANDS 7.131
00212F				1436	PREVSCRL DS 1	PREVIOUS SCROLLING POSITION 7.132
				1438	*	TO GETCHAR IF GETBUF IS NOT SOURCE 7.133
002130				1439	CRLFOVR DS 1	7.134
002131				1441	CNTLOMD DS 1	IS COPY OF FLAGS FOR FULLDPLX FAKE 7.135
002132				1443	ALLOWGMD DS 1	0=ALLOW GRAPH MODE, NZ = NO GMODE 7.136
				1445	*	7.137
002133				1446	XMITSTAT DS 1	1 IF XMITBUF IS FULL, 0 OTHERWISE 7.138
002134				1448	XMITTYPE DS 1	7.139
002135				1450	XMITGET DS 2	ADD OF WHERE TO GET CHAR TO SND 7.14
002137				1452	XMITPUT DS 2	ADD OF WHERE TO PUT CHAR IN BUF 7.141
002139				1454	XMITS AV DS 2	SAVED XMIT POINTER 7.142
00213B				1456	CRCNT DS 1	NUMBER OF <CR> IN XMIT BUF 7.143
00213C				1458	DC1CNT DS 1	WHETHER OR NOT DC1 HAS BEEN SENT 7.144
				1460	*	(I.E. CLEAR TO SEND) 7.145
				1461	*	MUST FOLLOW CRCNT 7.146
00213D				1462	SHIFTMD DS 1	WHICH CHAR SET TO USE 7.147
00213E				1464	XMITHAN DS 2	ADD OF XMITING ROUTINE 7.148
002140				1466	HTABTAB DS 10	TABTABLE 7.149
				1468	*****	7.15
				1469	*	7.151
				1470	*	7.152
				1471	*	7.153
				1472	*	7.154
				1473	*****	7.155
00214A				1474	TEKXXX DS 0X	STORGE FOR 5-BIT COORD ADDRESSES 7.156
00214A				1476	TEKXHI DS 1X	(ORDER-DEPENDENT) 7.157
00214B				1478	TEKXLOW DS 1X	7.158
00214C				1480	TEKYHI DS 1X	7.159
00214D				1482	TEKYLOW DS 1X	7.16
				1484	*	7.161
00214E				1485	GRAPHPOS DS 4X	CURRENT BEAM POSITION (10 BIT NOS.) 7.162
				1487	*	7.163
002152				1488	#QUADS DS X	# OF QUADRANTS FOR THE VECTOR 7.164
				1490	*	7.165
002153				1491	VECTEMP DS 5X	TEMP SAVE FOR VECTOR RTN 7.166

002158		1493	GRAPTEM	DS	4X	TEMP SAVE FOR GRAPH RTN	7.167
		1495	*				7.168
00215C		1496	CHARPOS	DS	2X		7.169
00215E		1498	DIREC1	DS	1X		7.17
00215F		1500	DIREC2	DS	1X		7.171
002160		1502	TYPE	DS	1X		7.172
	02102	1504	XOFF	EQU	CURSX		7.173
		1506		DS	1X		7.174
		1508	*				7.175
	058F0	1509	EVENADD	EQU	GRAPHEVN		7.176
	0AA41	1511	ODDADD	EQU	GRAPHODD		7.177

LOC	OBJECT CODE	ADDR1	ADDR2	STMT	SOURCE	STATEMENT	ASM H V 05 18.56 09/03/76
		05151	1513	NEXT	EQU	ODDADD-EVENADD	7.178
002162			1515	EVENODD	DS	1X	7.179
		02800	1517	ROMGEN	EQU	CHGENROM	7.18
			1519	*			7.181
002163			1520	XMASK	DS	1X	7.182
002164			1522	MARGIN	DS	1X	7.183
		02104	1524	GRPOLD	EQU	CURCHAR	7.184
002165			1526	GRPCSET	DS	2X	7.185
002167			1528	TEXT	DS	2	7.186
002169			1530	MTEXT	DS	2	7.187
00216B			1532	TEXTTOP	DS	2	7.188
00216D			1534	TEXTBOT	DS	2	7.189
00216F			1536	WASTED	DS	2	7.19
002171			1538	GRPDRAW	DS	3	7.191
			1540	*			7.192
		00008	1541	CURSTIME	EQU	8	7.193
		00014	1543	KEYRPTD	EQU	20	7.194
		00003	1545	KEYRPTR	EQU	3	7.195
		00004	1547	SCRQTIME	EQU	4	7.196
		00006	1549	SCRSTIME	EQU	6	7.197
		00030	1551	SCRPTIME	EQU	48	7.198
			1553	*			7.199
		01C00	1554	HELPIFND	EQU	ROM+7*K	7.2
			1556	*			7.201
			1557	*			7.202
		023FF	1558	STACK	EQU	CPURAM+CPURAMSZ-1	7.203
			1560	*			7.204
			1561	*			7.205
002174		04000	1562		ORG	RAM	7.206
004000			1564	XMITBUF	DS	256	7.207
		04100	1566	XMITEND	EQU	*	7.208
			1568	*			7.209
			1569	*			7.21
		04100	1570	BUFFER	EQU	*	7.211
			1572	*			7.212
		04400	1573	BUFEND	EQU	X'4400'	7.213
			1575	*			7.214

LOC	OBJECT CODE	ADDR1	ADDR2	STMT	SOURCE STATEMENT	ASM H V 05 18.56 09/03/76
1577					MACRO	7.216
1578	&LABEL				CMDEF &CHAR,&ROUTINE	7.217
1579	.*					7.218
1580	.*				DEFINE AN ENTRY IN THE COMMAND DEFINITION TABLE	7.219
1581	.*					7.22
1582					AIF ('&LABEL' EQ '').NOLABEL	7.221
1583	&LABEL				EQU *-2	7.222
1584	.NOLABEL				ANOP	7.223
1585					DC X'&CHAR',AL2(&ROUTINE-VGT)	7.224
1586					MEND	7.225
1588					MACRO	7.227
1589	.*					7.228
1590	.*				MULTIPLY	7.229
1591	.*					7.23
1592	&NAME				MULT &A	7.231
1593	&NAME				DS 0X	7.232
1594					AIF ('&A' NE '80').CHK81	7.233
1595					ADD HL,HL	7.234
1596					ADD HL,HL	7.235
1597					ADD HL,HL *8	7.236
1598					ADD HL,HL *16	7.237
1599					LOD D,H	7.238
1600					LOD E,L SAVE IT	7.239
1601					ADD HL,HL	7.24
1602					ADD HL,HL *64	7.241
1603					ADD HL,DE	7.242
1604					MEXIT	7.243
1605	.CHK81				AIF ('&A' NE '81').ERROR	7.244
1606					LOD DE,HL	7.245
1607					ADD HL,HL	7.246
1608					ADD HL,HL	7.247
1609					ADD HL,HL	7.248
1610					ADD HL,HL	7.249
1611					XCH HL,DE	7.25
1612					ADD HL,DE	7.251
1613					XCH HL,DE	7.252
1614					ADD HL,HL	7.253
1615					ADD HL,HL	7.254
1616					ADD HL,DE	7.255
1617					MEXIT	7.256
1618	.ERROR				MNOTE 12,'NOT MULT BE 80 OR 81'	7.257
1619					MEND	7.258
1621					MACRO	7.26
1622	&LABEL				DSALFINE	7.261
1623	.*				THIS IS A HUMOROUS MACRO WHICH MERELY COUSES A SUBROUTINE	7.262
1624	.*				TO REPEAT ITSELF BEFORE RETURNING.	7.263
1625					AIF ('&LABEL' EQ '').NOLABEL	7.264
1626	&LABEL				EQU *	7.265
1627	.NOLABEL				ANOP	7.266
1628					CALL *+3	7.267
1629					MEND	7.268
1630	*					8.

004100	02200	1631 *	DRG	X'2200'		9.
	02200	1632		*		10.
002200		1634 OLDPOS1	EQU		POSITION OF PADDLE 1	11.
002201		1636 OLDY1	DS	1	OLD Y POSITION FOR 1	12.
		1638 OLDX1	DS	1	OLD X POSITION FOR 1	13.
002202	02202	1640 OLDPOS2	EQU	*	POSITION OF PADDLE 2	14.
002203		1642 OLDY2	DS	1	OLD Y POSITION FOR 2	15.
		1644 CLDX2	DS	1	OLD X POSITON FOR 2	16.
002204	02204	1646 BALLPOS	EQU	*	BALL POSITION	17.
002205		1648 BALLY	DS	1	BALL Y POSITION	18.
		1650 BALLX	DS	1	BALL X POSITION	19.
002206	02206	1652 OLDTMP	EQU	*	TEMPORARY POSITON FOR X AND Y	20.
002207		1654 TMPY	DS	1		21.
		1656 TMPX	DS	1		22.
	0000F	1658 TOPY	EQU	TOPOFF/LINESIZE		23.
	000EB	1660 BOTY	EQU	235		24.
002208		1662 BALLTIME	DS	1	TIME FOR MOVING BALL	25.
	02209	1664 BALLDIR	EQU	*		26.
002209		1666 BALLDX	DS	1	DIRECT MOTION FOR X	27.
00220A		1668 BALLDY	DS	1	DIRECT MOTION FOR Y	28.
00220B		1670 OLDTIM	DS	1	PREVIOUS CLOCK VALUE	29.
00220C		1672 BALLSPED	DS	1	SPEED OF BALL	30.
	00028	1674 PADLSIZE	EQU	40	NUMBER OF RASTERS IN A PADDLE	31.
	02832	1676 BALLCENT	EQU	40*256+50	INITIAL BALL POSITION	32.
	0FFFF	1678 BALLDI	EQU	X'FFFF'	INITIAL BALL DIRECTION	33.
	00002	1680 INITSPED	EQU	2	INITIAL BALL SPEED	34.
	0000E	1682 BALLSIZE	EQU	14		35.
		1684 *				36.
00220D 3E02		1685	LODI	A,INITSPED	INITIAL SPEED OF BALL	37.
00220F 320C22		1687	ST	A,BALLSPED		38.
		1689	LODI	HL,BALLCENT	POSITION OF CENTER OF SCREEN	39.

ASM H V 05 18.56 09/03/76

LOC	OBJECT CODE	ADDR1	ADDR2	STMT	SOURCE STATEMENT	
002212	213228			1690		
002215	220422			1691	ST HL,BALLPOS	L IS Y 40.
002218	21FFFF			1693	LODI HL,BALLDI	INITIAL BALL DIRECT 41.
00221B	220922			1695	ST HL,BALLDIR	AND STORE THAT 42.
		0221E		1697	START EQU *	43.
00221E	3A1521			1699	LD A,TIME+3	SEE IF CHANGE IN TIME 44.
002221	210B22			1701	LODI HL,OLDTIM	45.
002224	BE			1703	CMP M	46.
002225	CA1E22			1705	JMP Z,START	47.
002228	77			1707	LOD M,A	SAVE IT 48.
002229	CD3222			1709	CALL PONGLOOP	49.
00222C	CD3222			1711	CALL PONGLOOP	50.
00222F	C31E22			1713	JMP START	51.
002232	2A0022			1715	PONGLOOP LD HL,OLDPOS1	GET OLD POSSITON 52.
002235	444D			1717	LOD BC,HL	AND SAVE IN BC 53.
002237	21FDFF			1719	LODI HL,X'FEFD'	PADDLES TO MOVE 54.
00223A	CD2723			1721	CALL MOVEPADL	55.
00223D	2A0622			1723	LD HL,OLDTMP	GET UPDATED POSITION 56.
002240	220022			1725	ST HL,OLDPOS1	SAVE UPDATED POSITION 57.
002243	2A0222			1727	LD HL,OLDPOS2	GET OLD POSSITON 58.
002246	444D			1729	LOD BC,HL	AND SAVE IN BC 59.
002248	21F7FB			1731	LODI HL,X'FBF7'	PADDLES TO MOVE 60.
00224B	CD2723			1733	CALL MOVEPADL	61.
00224E	2A0622			1735	LD HL,OLDTMP	GET UPDATED POSITION 62.
002251	220222			1737	ST HL,OLDPOS2	AND SAVE AS NEW POSITION 63.
002254	210822			1739	LODI HL,BALLTIME	SEE IF TIME TO MOVE THE BALL 64.
002257	35			1741	DEC M	65.
002258	C0			1743	RET NZ	NOT ZERO, THEN NOT TIME TO 66.
				1745	*	MOVE BALL 67.
002259	3A0C22			1746	LD A,BALLSPED	GET SPEED OF BALL 68.
00225C	77			1748	LOD M,A	AND SAVE IT IN BALL TIME 69.
00225D	CDA522			1750	CALL BALLMOVE	MOVE THE BALL 70.
002260	250F			1752	COLLIDE LODI H,TPY	CHECK FOR COLLISION WITH TOP 71.
002262	3A0422			1754	LD A,BALLY	GET Y POSITION 72.
002265	BC			1756	CMP H	SEE IF THE SAME 73.
002266	CCE722			1758	CALL Z,REFLECTY	GO REFLECT Y 74.
				1760	*	NOW FOR REFLECTION OFF OF BOTTOMN 75.
002269	26EB			1761	DCALREF LODI H,BOTY	76.
00226B	3A0422			1763	LD A,BALLY	77.
00226E	BC			1765	CMP H	SEE IF THE SAME 78.
00226F	CCE722			1767	CALL Z,REFLECTY	IF ZERO OR POS, THEN REFLECT Y 79.
002272	06FF			1769	LODI B,-1	REFLECTION IS X 80.
002274	3A0022			1771	LD A,OLDY1	CHECK LEFT HAND PLAYER FIST 81.
002277	C6F9			1773	ADDI -(BALLSIZE/2)	CORRECT FOR BALL 82.
002279	67			1775	LOD H,A	UPPER BOUNDER 83.

LOC	OBJECT CODE	ADDR1	ADDR2	STMT	SOURCE STATEMENT	ASM H V 05 18.56 09/03/76
00227A	C61B			1777	ADDI PADLSIZE/2+BALLSIZE/2	SPAN OF PADDLE 84.
00227C	6F			1779	LOD L,A	SAVE THAT 85.
00227D	3A0122			1781	LD A,OLDX1	86.
002280	57			1783	LOD D,A	87.
002281	5F			1785	LOD E,A	88.
002282	CDCB22			1787	CALL BALLCHEK	89.
				1789	* NOW FOR RIGHT HAND PLAYER	90.
002285	3A0222			1790	LD A,OLDY2	91.
002288	C6F9			1792	ADDI -(BALLSIZE/2)	CORRECT FOR BALL 92.
00228A	67			1794	LOD H,A	93.
00228B	C61B			1796	ADDI PADLSIZE/2+BALLSIZE/2	94.
00228D	6F			1798	LOD L,A	SPAN OF PADDLE 95.
00228E	3A0322			1800	LD A,OLDX2	X POSITION OF PADDLE 96.
002291	57			1802	LOD D,A	97.
002292	5F			1804	LOD E,A	98.
002293	CDCB22			1806	CALL BALLCHEK	99.
				1808	* NOW FOR LEFT AND RIGHT WALLS	100.
				1809	* IN THIS CASE THERE IS NO REFLECTION,	101.
				1810	* BUT INSTEAD BALL STARTS OUT	102.
				1811	*IN MIDDLE GOING THE OTHER	103.
				1812	*WAY	104.
				1813	*	105.
002296	06FE			1814	LODI B,-2	INDICATE TEST TYPE 106.
002298	110000			1816	LODI DE,0	107.
00229B	CDD522			1818	CALL BALLCHEX	JUST CHECK X 108.
00229E	115050			1820	LODI DE,80*256+80	109.
0022A1	CDD522			1822	CALL BALLCHEX	110.
0022A4	C9			1824	RET	GO BACK TO CALLER 111.
0022A5	CDBD22			1826	BALLMOVE CALL BALLCLR	ROUTINE TO ERASE BALL 112.
0022A8	2A0422			1828	LD HL,BALLPOS	NOW MOVE BALL 113.
0022AB	3A0922			1830	LD A,BALLOX	GET X POSITION 114.
0022AE	84			1832	ADD H	115.
0022AF	67			1834	LOD H,A	116.
0022B0	3A0A22			1836	LD A,BALLOY	GET Y POSTION 117.
0022B3	85			1838	ADD L	ADD IN CJRRENT POSITION 118.
0022B4	6F			1840	LOD L,A	PUT IT BACK 119.
0022B5	220422			1842	ST HL,BALLPOS	AND SAVE IT BACK 120.
0022B8	3E7F			1844	LODI A,@DEL	121.
0022BA	C3C122			1846	JMP BALLMOV2	122.
0022BD	2A0422			1848	BALLCLR LD HL,BALLPOS	GET BALLPOSITION 123.
0022C0	97			1850	SUB A	FOR CLEARING BALL 124.
0022C1	44			1852	BALLMOV2 LOD B,H	SAVE X 125.
0022C2	CD9423			1854	CALL MULT81	GO MULTIPLY Y BY 81 126.
0022C5	0607			1856	LODI B,BALLSIZE/2	127.
0022C7	CD6023			1858	CALL MOVECBJ	128.
0022CA	C9			1860	RET	GO BACK TO CALLER 129.
		022CB		1862	BALLCHEK EQU *	ROUTINE TO CHECK FOR COLLISIONS 130.
				1864	*H HAS LOW Y POSTION	131.
				1865	*L HAS HIGH Y POSTION	132.
				1866	*D HAS LOW X PCOSITON	133.
				1867	*E HAS HIGH X POSTION	134.
				1868	*B HAS TYPE OF REFLECTION	135.
				1869	*	136.
				1870	*	137.
0022CB	3A0422			1871	LD A,BALLY	GET Y POSITION OF BALL 138.

LDC	OBJECT CODE	ADDR1	ADDR2	STMT	SOURCE	STATEMENT	ASM H V 05 18.56 09/03/76
0022CE	BC			1873	CMP	H	139.
0022CF	DB			1875	RET	C	140.
0022D0	BD			1877	CMP	L	141.
0022D1	CAD522			1879	JMP	Z,YHIT	142.
0022D4	D0			1881	RET	NC	143.
		022D5		1883	YHIT EQU	*	144.
				1885	*	NOW WE CHECK FOR X COLLISION	145.
0022D5	3A0522			1886	BALLCHEX LD	A,BALLX	146.
0022D8	BA			1888	CMP	D	147.
0022D9	DB			1890	RET	C	148.
0022DA	BB			1892	CMP	E	149.
0022DB	CADF22			1894	JMP	Z,XHIT	150.
0022DE	D0			1896	RET	NC	151.
		022DF		1898	XHIT EQU	*	152.
				1900	*AT THIS POINT WE GO	A COLLISION	153.
				1901	*SO SEE WHAT WE SHOULD DO		154.
0022DF	78			1902	LOD	A,B	155.
0022E0	B7			1904	IOR	A	156.
0022E1	FAF522			1906	JMP	S,REFLECTX	157.
0022E4	CA0C23			1908	JMP	Z,NONE	158.
				1910	*NONE MEANS ONLY REFLECT Y		159.
0022E7	3A0A22			1911	REFLCTY LD	A,BALLDY	160.
0022EA	2F			1913	CMA	,	161.
0022EB	3C			1915	INC	A	162.
0022EC	320A22			1917	ST	A,BALLDY	163.
0022EF	CDA522			1919	CALL	BALLMOVE	164.
0022F2	D3AE			1921	OUT	KBCLICK	165.
0022F4	C9			1923	RET	,	166.
		022F5		1925	REFLECTX EQU	*	167.
				1927	INC	A	168.
0022F5	3C			1929	JMP	Z,BEEP BELL	169.
0022F6	CAFE22			1931	OUT	KBCLICK	170.
0022F9	D3AE			1933	JMP	RFLX	171.
0022FB	C30023			1935	BEEP BELL OUT	BELL	172.
0022FE	D386			1937	RFLX EQU	*	173.
002300	3A0922			1939	LD	A,BALLDX	174.
002303	2F			1941	CMA	,	175.
002304	3C			1943	INC	A	176.
002305	320922			1945	ST	A,BALLDX	177.
002308	CDA522			1947	CALL	BALLMOVE	178.
00230B	C9			1949	RET	,	179.
				1951	*		180.
				1952	*		181.
				1953	*		182.
		0230C		1954	NONE EQU	*	183.
				1956	PUSH	HL	184.
00230C	E5			1958	CALL	BALLCLR	185.
00230D	CDBD22			1960	LODI	HL,BALLCENT	186.
002310	213228			1962	ST	HL,BALLPOS	187.
002313	220422			1964	LD	A,BALLDX	188.
002316	3A0922			1966	CMA	,	189.
002319	2F			1968	INC	A	190.
00231A	3C						

LOC	OBJECT CODE	ADDR1	ADDR2	STMT	SOURCE STATEMENT	ASM H V 05 18.56 09/03/76
00231B	320922			1970	ST A,BALLDX	191.
00231E	3EFF			1972	LODI A,-1	FOR Y MOTION 192.
002320	320A22			1974	ST A,BALLDY	AND SAVE 193.
002323	D386			1976	OUT BELL	ADND BEEP THE BELL 194.
002325	E1			1978	POP HL	195.
002326	C9			1980	RET ,	AND GO HOME 196.
				1982	*	197.
		02327		1983	MOVEPADL EQU *	ROUTINE TO MOVE A PADDLE 198.
002327	7D			1985	LOD A,L	READ Y DIRECTION 199.
002328	CD8023			1987	CALL READAD	200.
002328	D65A			1989	SUBI 90	CHANGE ITS POSITION A LITTLE 201.
00232D	320622			1991	ST A,TMPY	202.
002330	6F			1993	LOD L,A	203.
002331	7C			1995	LOD A,H	204.
002332	CD8023			1997	CALL READAD	GET X DIRECTION 205.
002335	E6FC			1999	ANDI X'FC'	GET RID OF LOW BITS 206.
002337	0F			2001	ROT R,1	207.
002338	FE50			2003	CMPX CMPI 80	208.
00233A	FA4223			2005	JMP S,NOTXOCR	DON'T NEED TO CORRECT X 209.
00233D	D650			2007	SUBI 80	0 MAKE A CORRECTION IN X 210.
00233F	C33823			2009	JMP CMPX	211.
		02342		2011	NOTXOCR EQU *	212.
002342	320722			2013	ST A,TMPX	213.
				2015	*	NOW CHECK FOR ANY CHANGES 214.
002345	5F			2016	LOD E,A	SAVE X 215.
002346	B8			2018	CMP B	SEE IF ANY X CHANGE 216.
002347	C24D23			2020	JMP NZ,OLD#NEW	IF NOT EQUAL THEN CHANGE 217.
00234A	7D			2022	LOD A,L	CHECK FOR Y CHANGE 218.
00234B	B9			2024	CMP C	CHECK AGAINST OLD 219.
00234C	C8			2026	RET Z	DON'T REWRITE IF EQUAL 220.
		0234D		2028	OLD#NEW EQU *	221.
00234D	C5			2030	PUSH BC	SAVE PLACE TO ERASE 222.
00234E	43			2032	LOD B,E	SAVE NEW X POSITION 223.
00234F	CD9423			2034	CALL MULT81	MULTIPLY Y BY 81 AND ADD DE 224.
002352	E3			2036	XCH HL,(SP)	SAVE NEW POSITION 225.
002353	44			2038	LOD B,H	SAVE OLD X FOR ADD 226.
002354	CD9423			2040	CALL MULT81	MAKE THIS X AND Y 227.
002357	97			2042	SUB A	CLEAR OUT PADDLE 228.
002358	CD5E23			2044	CALL WRITPADL	GET RID OF OLD PADDLE 229.
00235B	E1			2046	POP HL	GET BACK NEW PADDLE POSITION 230.
00235C	3E7F			2048	LODI A,@DEL	A FILLED IN LINE 231.
		0235E		2050	WRITPADL EQU *	ROUTINE TO ERASE OR WRITE A PADL 232.
00235E	0614			2052	LODI B,PADLSIZE/2	PADDLE SIZE 233.
				2054	*	234.
				2055	*	235.
				2056	*	A HAS CHARACTER TO PUT IN DISPLAY 236.
				2057	*	HL HAS OFFSET FOR TOP OF PADDLE 237.
				2058	*	238.
		02360		2059	MOVEOBJ EQU *	ROUTINE TO MOVE AN OBJECT 239.
002360	115100			2061	LODI DE,LINESIZE	AMOUNT USED TO GET TO NEXT FRAME 240.
002363	E5			2063	PUSH HL	SAVE IF FOR SECOND HALF 241.
002364	C5			2065	PUSH BC	SAVE OBJECT SIZE 242.
002365	01AF5D			2067	LODI BC,GRAPHEVN+TOPOFF	VISABLE PART OF SCREEN 243.
002368	09			2069	ADD HL,BC	ADD OFFSET AND SCREN TOGETHER 244.
002369	C1			2071	POP BC	GET BACK OBJECT SIZE 245.

LOC	OBJECT CODE	ADDR1	ADDR2	STMT	SOURCE STATEMENT	ASM H V 05 18.56 09/03/76
00236A	C5			2073	PUSH BC	SAVE BC 246.
00236B	77			2075	WLP1 ST A,(HL)	247.
00236C	19			2077	ADD HL,DE	BUMP TO NEXT RASTER 248.
00236D	05			2079	DEC B	SEE IF DONE 249.
00236E	C26823			2081	JMP NZ,WLP1	250.
002371	C1			2083	POP BC	SO WE CAN GET TO X 251.
002372	E1			2085	POP HL	252.
002373	C5			2087	PUSH BC	BUT WE NEED IT LEATTER 253.
002374	0100AF			2089	LODI BC,GRAPHODD+TOPOFF	FOR OTHER RASTER FRAM 254.
002377	09			2091	ADD HL,BC	255.
002378	C1			2093	POP BC	GET BACK PADDLE SIZE 256.
002379	77			2095	WLP2 ST A,(HL)	257.
00237A	19			2097	ADD HL,DE	BUMP TO NEXT RASTER 258.
00237B	05			2099	DEC B	259.
00237C	C27923			2101	JMP NZ,WLP2	260.
00237F	C9			2103	RET	GO BACK TO CLLING ROUTINE 261.
				2105	*	262.
				2106	*	263.
				2107	*	264.
				2108	*	265.
				2109	*	266.
		02380		2110	READAD EQU *	ROUTINE TO GET A VALUE FROM 267.
				2112	*	A TO D PORT. PORT NUMBER IS IN A 268.
				2113	*	269.
002380	D39E			2114	OUT ATODSEL	TELL HARDWARE WHICH ONE 270.
002382	CD8D23			2116	CALL WAITATOD	WAIT FOR CONVERSION 271.
002385	D340			2118	OUT ATODSTR	TELL IT TO CONVERT 272.
002387	CD8D23			2120	CALL WAITATOD	WAIT FOR CONVERSION 273.
00238A	DB86			2122	INP ATODVAL	GET IT 274.
00238C	C9			2124	RET	AND RETURN TO CALLER 275.
00238D	1604			2126	WAITATOD LODI D,4	276.
00238F	15			2128	WAITLOOP DEC D	277.
002390	C28F23			2130	JMP NZ,WAITLOOP	278.
002393	C9			2132	RET	279.
		02394		2134	MULT81 EQU * MULT BY 81 ADD IN A	280.
002394	2600			2136	LODI H,0	FIRST CLEAR OUT H 281.
002396				2138	MULT 81	MULTIPLY Y BY 81 282.
002396	545D			2140		
002398	29			2141		
002399	29			2142		
00239A	29			2143		
00239B	29			2144		
00239C	EB			2145		
00239D	19			2146		
00239E	EB			2147		
00239F	29			2148		
0023A0	29			2149		
0023A1	19			2150		
0023A2	58			2151	LOD E,B	LOD IN E 283.
0023A3	1600			2153	LODI D,0	FOR DOUBLE ADD 284.
0023A5	19			2155	ADD HL,DE	ADD THE TWO 285.
0023A6	C9			2157	RET	AND GO HOME 286.
				2159	*	287.
				2160	*	288.
				2161	END	289.002

ASM H V 05 18.56 09/03/76

NO STATEMENTS FLAGGED IN THIS ASSEMBLY

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

649 CARDS FROM SYSIN
1304 LINES OUTPUT

1045 CARDS FROM SYSLIB
10 CARDS OUTPUT

ASP JOB NO. = 2594

FRIDAY SEPTEMBER 03, 1976 (76.247)

INPUT STATEMENTS (INCLUDING DD *) = 000673

//VGT CG024 JOB VGT\$CG, CLASS=E

0.002

ELAPSED TIME ON MAIN = SYA (A8) = 000.60, START TIME = 18.56.24

DDNAME = SYSMMSG

PRINTED ON PRT4

, LINES = 000151

DDNAME = ASM.MCS80.SYSTEM

PRINTED ON PRT4

, LINES = 000007

DDNAME = CLEANUP.OUT

PRINTED ON PRT4

, LINES = 000808

LINES OUTPUT FOR THIS JOB = 000966

CARDS FROM MAIN FOR THIS JOB = NONE
