

Home Video Game System Library

Bally Astrocade On-Board ROM (3159 Version)

Assembly Listing

December 28, 2003

Source Code	Pages
-----	-----
HVGLIB.H	1 - 12
System Subroutines	13 - 80
Scribbling	81 - 92
Calculator	93 - 106
Checkmate	107 - 126
Gunfight	127 - 170
Symbol Table	171 - 178

This page intentionally left blank for double-sided printing purposes

Home Video Game System Library - HVGLIB.H

```

1:          ; BALLYROM.ASM - VERSION 1.4 - MATCHES EXACTLY WITH 3159 ROM
2:          ;                               1.32 - DEC. 26, 2003
3:          ;                               - ASSEMBLED WITH NO ERRORS
4:          ;                               1.3 - DEC. 25, 2003
5:          ;                               - FINISHED TYPING
6:          ;                               1.15 - MAR. 1 2002
7:          ;
8:          ; zmac -m -o ballyrom.bin -x ballyrom.lst ballyrom.asm
9:          ;
10:         INCLUDE "HVGLIB.H"           ; HOME VIDEO GAME LIBRARY
**** HVGLIB.H ****
1:          ; HVGLIB.H (formally called ballyequ.h)
2:          ; Version 2.55 - December 15, 2003
3:          ; Bally Astrocade Equates and Macros Header File
4:          ;
5:          ; Retyped and proofread by Adam Trionfo and Lance F. Squire
6:          ; Version 1.0 (as ballyequ.h) - January 17, 2002
7:          ; Version 2.52 (Version 1.0 of HVGLIB.H) - March 28, 2003
8:          ; This file contains the equates and macros that Bally
9:          ; programs require for assembly. This file has been
10:         ; written to assemble with ZMAC 1.3 (a freely distribut-
11:         ; able Z-80 assembler (with C source), that has a 25-year
12:         ; history. ZMAC can be compiled under just about any O.S.
13:         ; in existence, so try it out. This file will probably
14:         ; require changes to be assembled under other assemblers.
15:         ;
16:         ; To assemble your Z-80 source code using ZMAC:
17:         ;
18:         ; zmac -d -o <outfile> -x <listfile> <filename>
19:         ;
20:         ; For example, assemble this Astrocade Z-80 ROM file:
21:         ;
22:         ; zmac -d -o BallyROM.bin -x BallyROM.lst BallyROM.asm
23:         ;
24:         ; Currently the Listing file is full of 'Undeclared'
25:         ; errors. When all of the source is typed-in, these will
26:         ; vanish. I'm leaving the others until all the source is
27:         ; re-typed.
28:         ;
29:         ; *****
30:         ; * HOME VIDEO GAME EQUATES *
31:         ; *****
32:         ;
33:         ;
34:         ; ASSEMBLY CONTROL
35:         ;
36:         0001          XPNDON EQU 1           ; ** SET TO 1 WHEN HARDWARE EXP
37:         0001          NWHDWR EQU 1          ; ** SET TO 1 WHEN NEW HARDWARE
38:         ;
39:         ; GENERAL GOODIES
40:         4000          NORMEM EQU $4000
41:         2000          FIRSTC EQU $2000      ; FIRST ADDRESS IN CARTRIDGE
42:         0000          SCREEN EQU 0
43:         0028          BYTEPL EQU 40         ; BYTES PER LINE
44:         00A0          BITSPL EQU 160        ; BITS PER LINE
45:         ; STUFF IN SYSTEM DOPE VECTOR
46:         0200          STIMER EQU $200       ; SECONDS AND GAME TIME, MUSIC
47:         0203          CTIMER EQU $203       ; CUSTOM TIMERS

```

Home Video Game System Library - HVGLIB.H

```

48: 0206          FNTPSYS EQU    $206      ; SYSTEM FONT DESCRIPTOR
49: 020D          FNTPSML EQU    $20D      ; SMALL FONT DESCRIPTOR
50: 0214          ALKEYS  EQU    $214      ; KEYMASK OF ALL KEYS
51: 0218          MENUST  EQU    $218      ; HEAD OF ONBOARD MENU
52: 021E          MXSCR   EQU    $21E      ; ADDRESS OF 'MAX SCORE'
53: 0228          NOPLAY  EQU    $228      ; ADDRESS OF '# OF PLAYERS'
54: 0235          NOGAME  EQU    $235      ; ADDRESS OF '# OF GAMES'
55:              ; BITS IN PROCESSOR FLAG BYTE
56: 0007          PSWSGN  EQU    7          ; SIGN BIT
57: 0006          PSWZRO  EQU    6          ; ZERO BIT
58: 0002          PSWPV   EQU    2          ; PARITY          OVERFLOW
59: 0000          PSWCY   EQU    0          ; CARRY
60:              ; BITS IN GAME STATUS BYTE
61: 0000          GSBTIM  EQU    0
62: 0001          GSBSCR  EQU    1
63: 0007          GSBEND  EQU    7
64:              ; STANDARD VECTOR DISPLACEMENTS AND BITS
65: 0000          VBMR    EQU    0          ; MAGIC REGISTER
66: 0001          VBSTAT  EQU    1          ; STATUS
67: 0002          VBTIMB  EQU    2          ; TIME BASE
68: 0003          VBDXL   EQU    3          ; DELTA X LO
69: 0004          VBDXH   EQU    4          ; DELTA X HI
70: 0005          VBXL    EQU    5          ; X COORD LO
71: 0006          VBXH    EQU    6          ; X COORD HI
72: 0007          VBXCHK  EQU    7          ; X CHECK FLAGS
73: 0008          VBDYL   EQU    8          ; DELTA Y LO
74: 0009          VBDYH   EQU    $09      ; DELTA Y HI
75: 000A          VBYL    EQU    $0A      ; Y COORD LO
76: 000B          VBYH    EQU    $0B      ; Y COORD HI
77: 000C          VBYCHK  EQU    $0C      ; Y CHECK FLAGS
78: 000D          VBOAL   EQU    $0D      ; OLD ADDRESS OF L.O.
79: 000E          VBOAH   EQU    $0E      ; OLD ADDRESS OF H.O.
80:              ; DISPLACEMENTS FROM START OF COORDINATE AREA
81: 0000          VBDCL   EQU    0          ; LO DELTA
82: 0001          VBDCH   EQU    1          ; HI DELTA
83: 0002          VBCL    EQU    2          ; LO COORD
84: 0003          VBCH    EQU    3          ; HI COORD
85: 0004          VBCCHK  EQU    4          ; CHECK BITS
86:              ; BITS IN STATUS BYTE
87: 0007          VBSACT  EQU    7          ; VECTOR ACTIVE STATUS
88: 0006          VBBLNK  EQU    6          ; BLANK STATUS
89:              ; BITS IN CHECK BIT MASK
90: 0000          VBCLMT  EQU    0          ; DO LIMIT CHECKING
91: 0001          VBCREV  EQU    1          ; REVERSE DELTA ON LIMIT ATTAIN
92: 0003          VBCLAT  EQU    3          ; COORDINATE IS AT LIMIT
93:              ; FONT TABLE DISPLACEMENTS FOR NEW CHARACTER DISPLAY ROU
94: 0000          FTBASE  EQU    0          ; BASE CHARACTER
95: 0001          FTFSX   EQU    1          ; X FRAME SIZE
96: 0002          FTFSY   EQU    2          ; Y FRAME SIZE
97: 0003          FTBYTE  EQU    3          ; X SIZE FOR CHAR IN BYTES
98: 0004          FTYSIZ  EQU    4          ; Y SIZE IN BITS
99: 0005          FTPTL   EQU    5          ; PATTERN TABLE ADDRESS LO
100: 0006          FTPTH   EQU    6          ; PATTERN TABLE ADDRESS HI
101:              ; BITS FOR MAGIC REGISTER WRITE OPTION BYTE
102: 0006          MRFLOP  EQU    6          ; WRITE WITH FLOP
103: 0005          MRXOR   EQU    5          ; WRITE WITH EXCLUSIVE OR
104: 0004          MROR    EQU    4          ; WRITE WITH OR

```

Home Video Game System Library - HVGLIB.H

```

105: 0003      MRXPND EQU 3      ; WRITE WITH EXPAND
106: 0002      MRROT EQU 2      ; WRITE WITH ROTATE
107: 0003      MRSHFT EQU $03    ; MASK OF SHIFT AMOUNT
108:           ; BITS OF CONTROL HANDLE INPUT PORT
109: 0004      CHTRIG EQU 4      ; TRIGGER
110: 0003      CHRIGH EQU 3      ; JOYSTICK RIGHT
111: 0002      CHLEFT EQU 2      ; JOYSTICK LEFT
112: 0001      CHDOWN EQU 1      ; DOWN
113: 0000      CHUP EQU 0       ; UP
114:           ; CONTEXT BLOCK REGISTER DISPLACEMENTS
115: 0000      CBIYL EQU 0       ; IY
116: 0001      CBIYH EQU 1
117: 0002      CBIXL EQU 2      ; IX
118: 0003      CBIXH EQU 3
119: 0004      CBE EQU 4        ; DE
120: 0005      CBD EQU 5
121: 0006      CBC EQU 6        ; BC
122: 0007      CBB EQU 7
123: 0008      CBFLAG EQU 8     ; AF
124: 0009      CBA EQU 9
125: 000A      CBL EQU $0A     ; HL
126: 000B      CBH EQU $0B
127:           ; SENTRY RETURN CODES EQUATES:
128: 0000      SNUL EQU 0       ; NOTHING HAPPENED
129: 0001      SCT0 EQU 1       ; COUNTER-TIMER 1 THRU 8
130: 0002      SCT1 EQU 2
131: 0003      SCT2 EQU 3
132: 0004      SCT3 EQU 4
133: 0005      SCT4 EQU 5
134: 0006      SCT5 EQU 6
135: 0007      SCT6 EQU 7
136: 0008      SCT7 EQU 8
137: 0009      SF0 EQU 9        ; FLAG BIT 0
138: 000A      SF1 EQU $0A
139: 000B      SF2 EQU $0B
140: 000C      SF3 EQU $0C
141: 000D      SF4 EQU $0D
142: 000E      SF5 EQU $0E
143: 000F      SF6 EQU $0F
144: 0010      SF7 EQU $10
145: 0011      SSEC EQU $11     ; SECONDS TIMER HAS COUNTED DOWN
146: 0013      SKYD EQU $13     ; KEY IS DOWN
147: 0012      SKYU EQU $12     ; YES IS UP
148: 001C      SP0 EQU $1C     ; POT IS 0
149: 001D      SP1 EQU $1D     ; POT IS 1
150: 001E      SP2 EQU $1E     ; POT IS 2
151: 001F      SP3 EQU $1F     ; POT IS 3
152: 0014      ST0 EQU $14     ; TRIGGER 0
153: 0015      SJ0 EQU $15     ; JOYSTICK 0
154: 0016      ST1 EQU $16     ; SIMILARLY FOR 1-3
155: 0017      SJ1 EQU $17
156: 0018      ST2 EQU $18
157: 0019      SJ2 EQU $19
158: 001A      ST3 EQU $1A
159: 001B      SJ3 EQU $1B
161:           ; *****
162:           ; * HOME VIDEO GAME PORT EQUATES *

```

Home Video Game System Library - HVGLIB.H

```

163:          ; *****
164:          ; OUTPUT PORTS FOR VIRTUAL COLOR
165: 0000      COLOR   EQU    0          ; COLOR 0 RIGHT
166: 0001      COL1R   EQU    1          ; COLOR 1 RIGHT
167: 0002      COL2R   EQU    2          ; COLOR 2 RIGHT
168: 0003      COL3R   EQU    3          ; COLOR 3 RIGHT
169: 0004      COL0L   EQU    4          ; COLOR 0 LEFT
170: 0005      COL1L   EQU    5          ; COLOR 1 LEFT
171: 0006      COL2L   EQU    6          ; COLOR 2 LEFT
172: 0007      COL3L   EQU    7          ; COLOR 3 LEFT
173: 000B      COLBX   EQU   $0B        ; COLOR BLOCK OUTPUT PORT
174: 0009      HORCB   EQU    9          ; HORIZONTAL COLOR BOUNDARY
175: 000A      VERBL   EQU   $0A        ; VERTICAL BLANKING LINE
176:          ; OUTPUT PORTS FOR MUSIC AND SOUNDS
177: 0010      TONMO   EQU   $10        ; TONE MASTER OSCILLATOR
178: 0011      TONEA   EQU   $11        ; TONE A OSC.
179: 0012      TONEB   EQU   $12        ; TONE B OSC.
180: 0013      TONEC   EQU   $13        ; TONE C OSC.
181: 0014      VIBRA   EQU   $14        ; VIBRATO
182: 0016      VOLAB   EQU   $16        ; TONES A,B VOLUME
183: 0015      VOLC    EQU   $15        ; TONE C VOLUME
184: 0017      VOLN    EQU   $17        ; NOISE VOLUME
185: 0018      SNDBX   EQU   $18        ; SOUND BLOCK OUTPUT PORT
186:          ; INTERRUPT AND CONTROL OUTPUT PORTS
187: 000D      INFBK   EQU   $0D        ; INTERRUPT FEEDBACK
188: 000E      INMOD   EQU   $0E        ; INTERRUPT MODE
189: 000F      INLIN   EQU   $0F        ; INTERRUPT LINE
190: 0008      CONCM   EQU    8          ; CONSUMER      COMMERCIAL
191: 000C      MAGIC   EQU   $0C        ; MAGIC REGISTER
192: 0019      XPAND   EQU   $19        ; EXPANDER PIXEL DEFINITION PORT
193:          ; INTERRUPT AND INTERCEPT INPUT PORTS
194: 0008      INTST   EQU    8          ; INTERCEPT STATUS
195: 000E      VERAFA  EQU   $0E        ; VERTICAL ADDRESS FEEDBACK
196: 000F      HORAFA  EQU   $0F        ; HORIZONTAL ADDRESS FEEDBACK
197:          ; HAND CONTROL INPUT PORTS
198: 0010      SW0     EQU   $10        ; PLAYER 0 HAND CONTROL
199: 0011      SW1     EQU   $11        ; PLAYER 1 HAND CONTROL
200: 0012      SW2     EQU   $12        ; PLAYER 2 HAND CONTROL
201: 0013      SW3     EQU   $13        ; PLAYER 3 HAND CONTROL
202: 001C      POT0    EQU   $1C        ; PLAYER 0 POT
203: 001D      POT1    EQU   $1D        ; PLAYER 1 POT
204: 001E      POT2    EQU   $1E        ; PLAYER 2 POT
205: 001F      POT3    EQU   $1F        ; PLAYER 3 POT
206:          ; KEYBOARD INPUT PORTS
207: 0014      KEY0    EQU   $14        ; KEYBOARD COLUMN 0
208: 0015      KEY1    EQU   $15        ; KEYBOARD COLUMN 1
209: 0016      KEY2    EQU   $16        ; KEYBOARD COLUMN 2
210: 0017      KEY3    EQU   $17        ; KEYBOARD COLUMN 3
212:          ; *****
213:          ; * HOME VIDEO GAME SYSTEM CALL INDEXES *
214:          ; *****
215:          ; USER PROGRAM INTERFACE
216: 0000      UPISTR   EQU    0          ;
217: 0000      INTPC    EQU   UPISTR      ; INTERPRET WITH CONTEXT CREATE
218: 0002      XINTC    EQU   INTPC+2    ; EXIT INTERPRETER WITH CONTEXT
219: 0004      RCALL    EQU   XINTC+2    ; CALL ASM LANGUAGE SUBROUTINE
220: 0006      MCALL    EQU   RCALL+2    ; CALL INTERPRETER SUBROUTINE

```

Home Video Game System Library - HVGLIB.H

```

221: 0008          MRET  EQU   MCALL+2    ; RETURN FROM INTERPRETER SUBRO
222: 000A          MJUMP  EQU   MRET+2     ; MACRO JUMP
223: 000C          SUCK   EQU   MJUMP+2   ; SUCK INLINE ARGS INTO CB
224:                ; SCHEDULER ROUTINES
225: 000C          SCHEDR EQU   SUCK
226: 000E          ACTINT EQU   SCHEDR+2   ; SET SUB TIMER
227: 0010          DECCTS EQU   ACTINT+2   ; DEC CT'S UNDER MASK
228:                ; MUSIC AND SOUNDS
229: 0012          MUZAK  EQU   DECCTS+2
230: 0012          BMUSIC EQU   MUZAK      ; BEGIN PLAYING MUSIC
231: 0014          EMUSIC EQU   BMUSIC+2   ; STOP PLAYING MUSIC
232:                ; SCREEN HANDLER ROUTINES
233: 0016          SCRSTR EQU   EMUSIC+2
234: 0016          SETOUT EQU   SCRSTR     ; SET SCREEN SIZE
235: 0018          COLSET EQU   SETOUT+2   ; SET COLORS
236: 001A          FILL   EQU   COLSET+2   ; FILL MEMORY WITH DAT
237: 001C          RECTAN EQU   FILL+2     ; PAINT RECTANGLE
238: 001E          VWRITR EQU   RECTAN+2   ; WRITE RELATIVE FROM VECTOR
239: 0020          WRITR  EQU   VWRITR+2   ; WRITE RELATIVE
240: 0022          WRITP  EQU   WRITR+2    ; WRITE WITH PATTERN SIZE LOOKUP
241: 0024          WRIT   EQU   WRITP+2   ; WRITE WITH SIZES PROVIDED
242: 0026          WRITA  EQU   WRIT+2     ; WRITE ABSOLUTE
243: 0028          VBLANK EQU   WRITA+2    ; BLANK AREA FROM VECTOR
244: 002A          BLANK  EQU   VBLANK+2   ; BLANK AREA
245: 002C          SAVE   EQU   BLANK+2    ; SAVE AREA
246: 002E          RESTOR EQU   SAVE+2     ; RESTORE AREA
247: 0030          SCROLL EQU   RESTOR+2   ; SCROLL AREA OF SCREEN
248:                ;
249: 0032          CHRDIS EQU   SCROLL+2   ; NEW DISPLAY CHARACTER
250: 0034          STRDIS EQU   CHRDIS+2   ; NEW DISPLAY STRING
251: 0036          DISNUM EQU   STRDIS+2   ; DISPLAY NUMBER
252:                ;
253: 0038          RELABS EQU   DISNUM+2   ; RELATIVE TO ABSOLUTE CONVERSI
254: 003A          RELAB1 EQU   RELABS+2   ; NONMAGIC RELABS
255: 003C          VECTC  EQU   RELAB1+2   ; VECTOR SINGLE COORDINATE
256: 003E          VECT   EQU   VECTC+2    ; VECTOR COORDINATE PAIR
257:                ; HUMAN INTERFACE ROUTINES
258: 0040          HUMANR EQU   VECT +2
259: 0040          KCTASC EQU   HUMANR     ; KEY CODE TO ASCII
260: 0042          SENTRY EQU   KCTASC+2   ; SENSE TRANSITION
261: 0044          DOIT   EQU   SENTRY+2   ; BRANCH TO TRANSITION HANDLER
262: 0046          DOITB  EQU   DOIT+2     ; USE B INSTEAD OF A
263: 0048          PIZBRK EQU   DOITB+2    ; TAKE A BREAK
264: 004A          MENU   EQU   PIZBRK+2   ; DISPLAY A MENU
265: 004C          GETPAR EQU   MENU+2     ; GET GAME PARAMENTER FROM USER
266: 004E          GETNUM EQU   GETPAR+2   ; GET NUMBER FROM USER
267: 0050          PAWS   EQU   GETNUM+2   ; PAUSE
268: 0052          DISTIM EQU   PAWS+2     ; DISPLAY TIME
269: 0054          INCSCR EQU   DISTIM+2   ; INC SCORE
270:                ; MATH ROUTINES
271: 0056          MATH   EQU   INCSCR+2
272: 0056          INDEXN EQU   MATH       ; INDEX NIBBLE
273: 0058          STOREN EQU   INDEXN+2   ;
274: 005A          INDEXW EQU   STOREN+2   ; INDEX WORD
275: 005C          INDEXB EQU   INDEXW+2   ; INDEX BYTE
276: 005E          MOVE   EQU   INDEXB+2   ; BLOCK TRANSFER
277: 0060          SHIFTU EQU   MOVE+2     ; SHIFT UP A DIGIT

```

Home Video Game System Library - HVGLIB.H

```

278: 0062          BCDADD EQU   SHIF TU+2   ; BCD ADD
279: 0064          BCDSUB EQU   BCDADD+2 ; BCD SUBTRACT
280: 0066          BCDMUL EQU   BCDSUB+2 ; BCD MULTIPLY
281: 0068          BCDDIV EQU   BCDMUL+2 ; BCD DIVIDE
282: 006A          BCDCHS EQU   BCDDIV+2 ; BCD CHANGE SIGN
283: 006C          BCDNEG EQU   BCDCHS+2 ; BCD NEGATE
284: 006E          DADD EQU   BCDNEG+2 ; DECIMAL ADD
285: 0070          DSMG EQU   DADD+2 ; CONVERT TO SIGN MAGNITUDE
286: 0072          DABS EQU   DSMG+2 ; DECIMAL ABSOLUTE VALUE
287: 0074          NEG T EQU   DABS+2 ; NEGATE
288: 0076          RANGED EQU   NEG T+2 ; RANGED RANDOM NUMBER
289: 0078          QUIT EQU   RANGED+2 ; QUIT CASSETTE EXECUTION
290: 007A          SETB EQU   QUIT+2 ; SET BYTE
291: 007C          SETW EQU   SETB+2 ; SET WORD
292: 007E          MSKTD EQU   SETW+2 ; MASK TO DELTAS
294:                ; *****
295:                ; * MACROS *
296:                ; *****
297:                ; MACROS TO DEFINE PATTERNS
298:          DEF2    MACRO  AA, AB
299:                DB      AA
300:                DB      AB
301:                ENDM
302:          DEF3    MACRO  BA, BB, BCC
303:                DB      BA
304:                DB      BB
305:                DB      BCC ; 'BC' reserved, so used 'BCC'
306:                ENDM
307:          DEF4    MACRO  CA, CB, CC, CD
308:                DB      CA
309:                DB      CB
310:                DB      CC
311:                DB      CD
312:                ENDM
313:          DEF5    MACRO  DA, DBB, DC, DD, DEE
314:                DB      DA
315:                DB      DBB
316:                DB      DC
317:                DB      DD
318:                DB      DEE
319:                ENDM
320:          DEF6    MACRO  EA, EB, EC, ED, EE, EF
321:                DB      EA
322:                DB      EB
323:                DB      EC
324:                DB      ED
325:                DB      EE
326:                DB      EF
327:                ENDM
328:          DEF8    MACRO  GA, GB, GC, GD, GEE, GF, GG, GH
329:                DB      GA
330:                DB      GB
331:                DB      GC
332:                DB      GD
333:                DB      GEE ; 'GE' reserved, so used 'GEE'
334:                DB      GF
335:                DB      GG

```

```

336:          DB      GH
337:          ENDM
338:          ; MACROS TO COMPUTE CONSTANT SCREEN ADDRESSES
339: XYRELL  MACRO  p1, p2, p3 ; RELATIVE LOAD
340:          LD      p1, [(p3 SHL 8)+p2]
341:          ENDM
342:          ; MACRO TO GENERATE SYSTEM CALL
343: SYSTEM  MACRO  NUMBA
344:          RST     56
345:          DB      NUMBA
346:          IF      NUMBA = INTPC
347: INTPCC  DEFL   1
348:          ENDF
349:          ENDM
350:          ; MACRO TO GENERATE SYSTEM CALL WITH SUCK OPTION ON
351: SYSSUK  MACRO  UMBA
352:          RST     56
353:          DB      UMBA+1
354:          IF      UMBA = INTPC
355: INTPCC  DEFL   1
356:          ENDF
357:          ENDM
358:          ; MACROS TO GENERATE MACRO INSTRUCTION CALLS
359:          ; FILL SCREEN WITH CONSTANT DATA (was 'FILL?')
360: FILLq   MACRO  START, NBYTES, DATA
361:          DB      FILL+1
362:          DW      START
363:          DW      NBYTES
364:          DB      DATA
365:          ENDM
366:          ; EXIT INTERPRETER WITH CONTEXT RESTORE
367: EXIT    MACRO
368:          DB      XINTC
369: INTPCC  DEFL   0
370:          ENDM
371:          ; INTERPRET WITH INLINE SUCK
372: DO      MACRO  CID
373:          DB      CID+1
374:          ENDM
375:          ; INTERPRET WITHOUT INLINE SUCK
376: DONT    MACRO  CID
377:          DB      CID
378:          ENDM
379:          ; MACRO CALL FROM DOIT TABLE
380: 00C0   ENDx    EQU    $C0
381: MC      MACRO  AA, BB, EE
382:          DB      AA+$80
383:          DW      BB
384:          IF      EE = $C0
385:          DB      EE
386:          ENDF
387:          ENDM
388:          ; REAL CALL FROM DOIT TABLE
389: RC      MACRO  AA, BB, EE
390:          DB      AA+$40
391:          DW      BB
392:          IF      EE = $C0

```

```

393:          DB      EE
394:          ENDIF
395:          ENDM
396:          ; REAL JUMP FROM DOIT TABLE
397:          JMPd    MACRO AA, BB, EE
398:          DB      AA
399:          DW      BB
400:          IF      EE = $C0
401:          DB      EE
402:          ENDIF
403:          ENDM
404:          ; DISPLAY A STRING
405:          TEXTD   MACRO AA, BB, CC, DD
406:          DB      STRDIS+1
407:          DB      BB
408:          DB      CC
409:          DB      DD
410:          DW      AA
411:          ENDM

413:          ;*****
414:          ; MUSIC MACROS
415:          ; NOTE DURATION, FREQ(S)
416:          NOTE1   MACRO DUR, N1
417:          DB      (DUR)&($7F)
418:          DB      N1
419:          ENDM
420:          NOTE2   MACRO DUR, N1, N2
421:          DB      (DUR)&($7F)
422:          DB      N1
423:          DB      N2
424:          ENDM
425:          NOTE3   MACRO DUR, N1, N2, N3
426:          DB      DUR
427:          DB      N1
428:          DB      N2
429:          DB      N3
430:          ENDM
431:          NOTE4   MACRO DUR, N1, N2, N3, N4
432:          DB      DUR
433:          DB      N1
434:          DB      N2
435:          DB      N3
436:          DB      N4
437:          ENDM
438:          NOTE5   MACRO DUR, N1, N2, N3, N4, N5
439:          DB      DUR
440:          DB      N1
441:          DB      N2
442:          DB      N3
443:          DB      N4
444:          DB      N5
445:          ENDM
446:          MASTER  MACRO OFFSET

```

Home Video Game System Library - HVGLIB.H

```

447:          DB      $80
448:          DB      OFFSET
449:          ENDM
450:          ; STUFF OUTPUT PORT#, DATA OR
451:          ; OUTPUT SNDBX, DATA10, D11,..., DATA17
452: OUTPUT MACRO PORT, D0, D1, D2, D3, D4, D5, D6, D7
453:          ;      IF      NOT (PORT=$18)
454:          IF      PORT!=$18
455:          DB      $80+((PORT)&($7F))
456:          DB      D0
457:          ENDIFF
458:          IF      PORT=$18
459:          DB      $88
460:          DEF8    D7, D6 ,D5, D4, D3, D2, D1, D0
461:          ENDIFF
462:          ENDM
463:          ; SET VOICE BYTE
464:          ; THE FORMAT OF THE VOICE BYTE IS
465:          ; *I*A*I*B*I*C*V*N
466:          ; WHERE N = LOAD NOISE WITH DATA AT PC AND INC PC
467:          ; V = LOAD VIBRATO AND INC PC
468:          ; I = INC PC
469:          ; A,B,C = LOAD TONE A,B,C WITH DATA AT PC
470: VOICEM MACRO MASK ; 'VOICES' TO 'VOICEM'
471:          DB      $90
472:          DB      MASK
473:          ENDM
474:          ; PUSH NUMBER ONTO STACK
475: PUSHN MACRO NUMB
476:          DB      $A0+((NUMB-1) AND $F)
477:          ENDM
478:          ; SET VOLUMES
479: VOLUME MACRO P1, P2
480:          DB      $B0
481:          DB      P1
482:          DB      P2
483:          ENDM
484:          ; CALL RELATIVE 0-15 BEYOND SELF+1
485: CREL MACRO BY
486:          DB      $D0+(BY AND $F)
487:          ENDM
488:          ; DEC STACK TOP AND JNZ
489: DSJNZ MACRO ADD_IT
490:          DB      $C0
491:          DW      ADD_IT
492:          ENDM
493:          ; FLIP LEGATO STACATO
494: LEGSTA MACRO
495:          DB      $E0
496:          ENDM
497: REST MACRO TIME
498:          DB      $E1
499:          DB      TIME
500:          ENDM
501: QUIET MACRO
502:          DB      $F0
503:          ENDM

```

Home Video Game System Library - HVGLIB.H

```

504:          ; *****
505:          ; * MUSIC EQUATES *
506:          ; *****
507:          ; NOTE VALUES
508: 00FD          G0          EQU          253
509: 00EE          GS0         EQU          238
510: 00E1          A0          EQU          225
511: 00D4          AS0         EQU          212
512: 00C8          B0          EQU          200
513: 00BD          C1          EQU          189
514: 00B2          CS1         EQU          178
515: 00A8          D1          EQU          168
516: 009F          DS1         EQU          159
517: 0096          E1          EQU          150
518: 008D          F1          EQU          141
519: 0085          FS1         EQU          133
520: 007E          G1          EQU          126
521: 0077          GS1         EQU          119
522: 0070          A1          EQU          112
523: 006A          AS1         EQU          106
524: 0064          B1          EQU          100
525: 005E          C2          EQU           94
526: 0059          CS2         EQU           89
527: 0054          D2          EQU           84
528: 004F          DS2         EQU           79
529: 004A          E2          EQU           74
530: 0046          F2          EQU           70
531: 0042          FS2         EQU           66
532: 003E          G2          EQU           62
533: 003B          GS2         EQU           59
534: 0037          A2          EQU           55
535: 0034          AS2         EQU           52
536: 0031          B2          EQU           49
537: 002E          C3          EQU           46
538: 002C          CS3         EQU           44
539: 0029          D3          EQU           41
540: 0027          DS3         EQU           39
541: 0025          E3          EQU           37
542: 0022          F3          EQU           34
543: 0020          FS3         EQU           32
544: 001F          G3          EQU           31
545: 001D          GS3         EQU           29
546: 001B          A3          EQU           27
547: 001A          AS3         EQU           26
548: 0018          B3          EQU           24
549: 0017          C4          EQU           23
550: 0015          CS4         EQU           21
551: 0014          D4          EQU           20
552: 0013          DS4         EQU           19
553: 0012          E4          EQU           18
554: 0011          F4          EQU           17
555: 0010          FS4         EQU           16
556: 000F          G4          EQU           15
557: 000E          GS4         EQU           14
558: 000D          A4          EQU           13
559: 000B          C5          EQU           11
560: 000A          CS5         EQU           10

```

Home Video Game System Library - HVGLIB.H

```

561: 0009          DS5      EQU      9
562: 0008          F5       EQU      8
563: 0007          G5       EQU      7
564: 0006          A5       EQU      6
565: 0005          C6       EQU      5
566: 0004          DS6      EQU      4
567: 0003          G6       EQU      3
568: 0002          C7       EQU      2
569: 0001          G7       EQU      1
570: 0000          G8       EQU      0
571:                ; MASTER OSCILATOR OFFSETS
572: 00FE          OB0      EQU      254
573: 00F1          OC0      EQU      241
574: 00D6          OD1      EQU      214
575: 00BF          OE1      EQU      191
576: 00B4          OF1      EQU      180
577: 00A0          OG1      EQU      160
578: 008F          OA1      EQU      143
579: 0047          OA2      EQU      71
580: 0023          OA3      EQU      35
581: 0011          OA4      EQU      17
582: 0008          OA5      EQU      8
584:                ; *****
585:                ; * SYSTEM RAM MEMORY CELLS *
586:                ; *****
587: 0FFF          WASTE     EQU      0FFFH
588: 0FFF          WASTER    EQU      WASTE
589:                ;
590:                ; THE FOLLOWING ORG SHOULD BE SET TO THE VALUE OF
591:                ; THE TAG 'SYSRAM', THIS WILL CAUSE SYSTEM RAM
592:                ; TO RESIDE AT THE HIGEST POSSIBLE ADDRESS
593:                ;
594:                ;          ORG      $4FC8
595:                ;          DEFS     6          ; GOT SOME LEFT STILL
596: 4FCE          BEGRAM    EQU      $4FCE
597:                ; USED BY MUSIC PROCESSOR
598: 4FCE          MUZPC     EQU      $4FCE          ; MUSIC PROGRAM COUNTER
599: 4FD0          MUZSP     EQU      $4FD0          ; MUSIC STACK POINTER
600: 4FD2          PVOLAB    EQU      $4FD2          ; PRESET VOLUME FOR TONES A AND
B
601: 4FD3          PVOLMC    EQU      $4FD3          ; PRESET VOLUME FOR MASTER OSC
602: 4FD4          VOICES    EQU      $4FD4          ; MUSIC VOICES
603:                ; COUNTER TIMERS (USED BY DECCTS,ACTINT,CTIMER)
604: 4FD5          CT0       EQU      $4FD5          ; COUNTER TIMER 0
605: 4FD6          CT1       EQU      $4FD6          ; 1
606: 4FD7          CT2       EQU      $4FD7          ; 2
607: 4FD8          CT3       EQU      $4FD8          ; 3
608: 4FD9          CT4       EQU      $4FD9          ; 4
609: 4FDA          CT5       EQU      $4FDA          ; 5
610: 4FDB          CT6       EQU      $4FDB          ; 6
611: 4FDC          CT7       EQU      $4FDC          ; 7
612:                ;USED BY SENTRY TO TRACK CONTROLS
613: 4FDD          CNT       EQU      $4FDD          ; COUNTER UPDATE&NUMBER TRACKING
614: 4FDE          SEMI4S    EQU      $4FDE          ; FLAG BITS
615: 4FDF          OPOT0     EQU      $4FDF          ; POT 0 TRACKING
616: 4FE0          OPOT1     EQU      $4FE0          ; POT 1 TRACKING
617: 4FE1          OPOT2     EQU      $4FE1          ; POT 2 TRACKING

```

Home Video Game System Library - HVGLIB.H

```

618: 4FE2          OPOT3  EQU    $4FE2      ; POT 3 TRACKING
619: 4FE3          KEYSEX EQU    $4FE3      ; KEYBOARD TRACKING BYTE
620: 4FE4          OSW0   EQU    $4FE4      ; SWITCH 0 TRACKING
621: 4FE5          OSW1   EQU    $4FE5      ; SWITCH 1 TRACKING
622: 4FE6          OSW2   EQU    $4FE6      ; SWITCH 2 TRACKING
623: 4FE7          OSW3   EQU    $4FE7      ; SWITCH 3 TRACKING
624: 4FE8          COLLST EQU    $4FE8      ; COLOR LIST ADDRESS FOR P.B.A
625:                ; USED BY STIMER
626: 4FEA          DURAT  EQU    $4FEA      ; NOTE DURATION
627: 4FEB          TMR60  EQU    $4FEB      ; SIXTIETHS OF SEC
628: 4FEC          TIMOUT EQU    $4FEC      ; BLAKOUT TIMER
629: 4FED          GTSECS EQU    $4FED      ; GAME TIME SECONDS
630: 4FEE          GTMINS EQU    $4FEE      ; GAME TIME MINUTES
631:                ; USED BY MENU
632: 4FEF          RANSHT EQU    $4FEF      ; RANDOM NUMBER SHIFT REGISTER
633: 4FF3          NUMPLY EQU    $4FF3      ; NUMBER OF PLAYERS
634: 4FF4          ENDSCT EQU    $4FF4      ; SCORE TO 'PLAY TO'
635: 4FF7          MRLOCK EQU    $4FF7      ; MAGIC REGISTER LOCK OUT FLAG
636: 4FF8          GAMSTB EQU    $4FF8      ; GAME STATUS BYTE
637: 4FF9          PRIOR  EQU    $4FF9      ; MUSIC PROTECT FLAG
638: 4FFA          SENFLG EQU    $4FFA      ; SENTRY CONTROL SEIZURE FLAG
639: 4FFB          UMARGT EQU    $4FFB
640: 4FFD          USERTB EQU    $4FFD
641: 9FCD          SYSRAM EQU    ($5000-($-BEGRAM+1))
**** ballyrom.asm ****

```

Home Video Game System Library - System Subroutines

```

11:                ; *****
12:                ; * HVGSYS *   HOME VIDEO GAME SYSTEM
13:                ; *****

15: 0008          PFUG    EQU 08H           ; POT FUDGE FACTOR
16: 17DE          GFSTRT EQU 17DEH        ; GUNFIGHT START ADDRESS
17: 1328          CMSTRT EQU 1328H       ; CHECKMATE START ADDRESS
18: 1020          CALCST EQU 1020H       ; CALCULATOR START ADDRESS
19: 0E19          SCBST   EQU 0E19H       ; SCRIBBLING START ADDRESS

21:                ; *****
22:                ; * POWER UP RESTART *
23:                ; *****
24: 0000          ORG 0
25: 0000 00      NOP                ; WAIT FOR THINGS TO SETTLE DOW
26: 0001 F3      DI
27: 0002 AF      XOR  A
28: 0003 D308    OUT  (CONCM),A      ; *** SET CONSUMER MODE ***
29: 0005 C3610C  JP   PWRUP

31: 0008          ORG 8
32:                ; TRANSFER CONTROL TO RESTART HANDLER
33: 0008 C30720  JP  2007H           ; VECTOR OUT

35: 000B 1C      NUMBAS DB  1CH
36: 000C 3C      DB  3CH
37: 000D 1C      DB  1CH
38: 000E 20      DB  20H
39:                ;
40: 000F FF      DB  0FFH ; FILL EMPTY SPACE WITH FFH (TO MATCH
3159 ROM)
41:                ;

43: 0010          ORG 16
44: 0010 C30A20  JP  200AH           ; RESTART 2
45: 0013 06      MENUCL DB  06H           ; MENU COLORS
46: 0014 FA      DB  0FAH           ; CHANGED TO MATCH 3159 ROM
47: 0015 07      DB  07H
48: 0016 62      DB  62H           ; CHANGED TO MATCH 3159 ROM
49:                ;
50: 0017 FF      DB  0FFH ; FILL EMPTY SPACE WITH FFH (TO MATCH
3159 ROM)
51:                ;

53: 0018          ORG 24
54: 0018 C30D20  JP  200DH           ; RESTART 3
55:                ; NAME          PAUSE
56:                ; PURPOSE        HALT # OF INTERRUPTS
57:                ; INPUT          B = # OF INTERRUPTS
58:                ; INPUT          B = # OF INTERRUPTS
59: 001B FB      MPAUSE EI
60: 001C 76      HALT

```

Home Video Game System Library - System Subroutines

```

61: 001D 10FD          DJNZ ($-1)
62: 001F C9           RET
64: 0020             ORG 32
65: 0020 C31020        JP 2010H          ; RESTART 4

67:                ; NAME: SET WORD
68:                ; (HL)=DE
69: 0023 73          MSETW LD (HL),E
70: 0024 23          INC HL
71: 0025 72          LD (HL),D
72: 0026 C9           RET
73:                ;
74: 0027 FF          DB 0FFH ; FILL EMPTY SPACE WITH FFH (TO MATCH
3159 ROM)
75:                ;

77: 0028             ORG 40
78: 0028 C31320        JP 2013H          ; RESTART 5

80: 002B 210000       CONC2 LD HL,0          ; ZERO OUT HL
81: 002E C9           RET
82:                ;
83: 002F FF          DB 0FFH ; FILL EMPTY SPACE WITH FFH (TO MATCH
3159 ROM)
84:                ;

86: 0030             ORG 48
87: 0030 C31620        JP 2016H          ; RESTART 6

89: 0033 00          CKSUM DB 0          ; CHECKSUM

91: 0034 8B01        ITAB DW MACTIN        ; INTERRUPT TRANSFER
92: 0036 01          DB 1          ; ** SYSTEM REVISION LEVEL
93:                ;
94: 0037 FF          DB 0FFH ; FILL EMPTY SPACE WITH FFH (TO MATCH
3159 ROM)
95:                ;

97: 0038             ORG 56
98:                ; NAME: USER PROGRAM INTERFACE
99:                ; PURPOSE: TRANSFER OF CONTROL FROM USER TO SYSTEM
100:                ; INPUT: ROUTINE # FOLLOWS INLINE AFTER RST INSTR
101:                ; IF L.O. BIT SET, LOAD ARGUMENTS INLINE F
102:                ; OUTPUT: NONE
103:                ; STACK USE: 18 BYTES TOTAL, 16 BYTES ON EXIT
104:                ; SIDE EFFECTS: REGISTERS AF,BC,DE,HL,IX, AND OLD IY SAV
105:                ; EXPLANATION
106:                ; REGISTERS AF,BC,DE,HL,IX, AND PREVIOUS IY ARE PUSHED
107:                ; THE NUMBER FOLLOWING THE RST 56 INSTRUCTION IS USED TO
108:                ; INDEX A JUMP VECTOR GIVING THE STARTING ADDRESS OF THE

```

Home Video Game System Library - System Subroutines

```

109:          ; SYSTEM ROUTINE TO CALL.  IF OPTIONED, INLINE ARGUMENTS
110:          ; ARE COPIED INTO THE CONTEXT AREA.  FOR ARGUMENT ORDERIN
111:          ; SEE INTERPRETER DOCUMENTATION AND APPROP. TABLES
112:          ; A DUMMY RETURN IS INSERTED WHICH, WHEN RETURNED TO BY
113:          ; SYSTEM ROUTINE, WILL RESTORE THE REGISTER CONTENTS AND
114:          ; RETURN TO THE USER PROGRAM
115:          ;
116:          ;   *** THE UPI HAS BEEN EXTENDED TO SUPPORT USER SUPPLI
117:          ;   ROUTINES.  IF THE CALL INDEX PROVIDED IS NEGATIVE
118:          ;   THEN THE USERS DISPATCH TABLE POINTER (USERTB) IS US
119:          ;   NOTE THAT THE SIGN BIT ISN'T ZAPPED BEFORE BEING
120:          ;   USED AS AN INDEX, THIS MEANS THAT THE USERS DISPATCH
121:          ;   TABLE POINTER SHOULD POINT 128 BYTES BEFORE THE FIRS
122: 0038 E3          EX   (SP),HL          ; RETURN ADDRESS TO HL
123: 0039 F5          PUSH AF              ; CREATE CONTEXT
124: 003A C5          PUSH BC
125: 003B D5          PUSH DE
126: 003C DDE5       PUSH IX
127: 003E FDE5       PUSH IY
128: 0040 FD210000   LD   IY,0          ; POINT IY AT CONTEXT
129: 0044 FD39          ADD  IY,SP
130: 0046 7E          LD   A,(HL)          ; LOAD OPCODE
131: 0047 23          INC  HL
132: 0048 117A02     LD   DE,RETN_        ; DE = RETURN POINT
133: 004B 1F          RRA              ; SUCK WANTED
134: 004C 3836       JR   C, $(MINT0-$) ; JUMP IF YES
135: 004E E5          INTPE PUSH HL          ; SAVE PC
136: 004F D5          PUSH DE          ; SAVE DUMMY RETURN
137: 0050 21CB00     LD   HL,SYSDPT
138: 0053 07          RLCA
139: 0054 5F          LD   E,A
140: 0055 1600       LD   D,0
141: 0057 17          RLA              ; USER TABLE WANTED?
142: 0058 3003       JR   NC, $(PUSH1-$)
143: 005A 2AFD4F     LD   HL,(USERTB) ; YES   LOAD IT
144: 005D 19          PUSH1 ADD  HL,DE
145: 005E 5E          LD   E,(HL)
146: 005F 23          INC  HL
147: 0060 56          LD   D,(HL)
148: 0061 D5          PUSH DE
149: 0062 FD660B     LD   H,(IY+CBH)
150: 0065 FD6E0A     LD   L,(IY+CBL)
151: 0068 FD5603     RELD LD   D,(IY+CBIXH)
152: 006B FD5E02     LD   E,(IY+CBIXL)
153: 006E D5          PUSH DE
154: 006F DDE1       POP  IX
155: 0071 FD7E09     LD   A,(IY+CBA)
156: 0074 FD5605     DELOAD LD  D,(IY+CBD)
157: 0077 FD5E04     LD   E,(IY+CBE)
158: 007A C9          RET              ; CALL VIA RETURN
160:          ; NAME:          MACRO INTERPRETER
161:          ; PURPOSE:       INTERPRETING SEQUENCES OF SYSTEM CALLS
162:          ; INPUT:         ADDRESS OF STRING 0 TO INTERPRET PASSED ON
163:          ; STACK USE:    NO INCREASE IN DEPTH
164:          ; EXPLANATION:  IF OPTIONED (BIT 0 OF CALL INDEX SET) THE
165:          ; ARGUMENT TABLE (MRARGT) IS INDEXED GIVING A MASK WHICH
166:          ; SPECIFIES HOW TO TRANSFER INLINE ARGUMENTS INTO THE COL

```

Home Video Game System Library - System Subroutines

```

167:          ; BLOCK. THIS MASK IS FORMATTED AS FOLLOWS:
168:          ;
169:          ;
170:          ; *****
171:          ; * 7 * 6 * 5 * 4 * 3 * 2 * 1 * 0 *
172:          ; *****
173:          ; * H * L * A * IX * B * C * D * E *
174:          ; *****
175:          ; ARGUMENTS MUST FOLLOW THE CALL INDEX IN THE FOLLOWING
176:          ; (OMITTING UNUSED ARGUMENTS, OF COURSE)
177:          ; (INDEX), IXL, IXH, E, D, C, B, A, L, H
178:          ;
179:          ; THE SIMULATED PC IS SAVED AND A DUMMY RETURN IS
180:          ; INSERTED ON THE STACK. THE UPI DISPATCHING ROUTINE IS
181:          ; THEN ENTERED AT 'INTPE', WHICH EFFECTS A CONTROL TRANS
182:          ; TO THE CALLED ROUTINE. WHEN THE CALLED ROUTINE RETURN
183:          ; IT WILL COME BACK HERE TO INTERPRET THE NEXT MACRO INS
184:          ; NOTE THAT THIS ROUTINE IS REENTRANT, THEREFORE THE CAL
185:          ; ROUTINE MAY RECUR BACK THRU HERE, IF IT FEELS LIKE IT.
186:          ; ** THE UPI HAS BEEN EXTENDED TO PROVIDE USER PROVIDED
187:          ; SYSTEM ROUTINES. IF A NEGATIVE CALL INDEX IS ENCOUNTER
188:          ; BY THE INTERPRETER, AND 'SUCK INLINE' IS OPTIONED, THE
189:          ; USER MACRO ROUTINE ARGUMENT TABLE IS INDEXED FOR A
190:          ; PARAMETER MASK. THE ADDRESS OF THIS TABLE IS ASSUMED
191:          ; TO BE IN (UMARGT), (UMARGT+1). THIS POINTER SHOULD
192:          ; POINT 64 BYTES BEFORE THE FIRST REAL ENTRY.
193:          ; I.E. LD HL, USERMT-64 ; WHERE USERMT POINTS AT
194:          ; LD (UMARGT), HL
195: 007B D1 MINTPC POP DE ; DISCARD DUMMY RETURN FROM UPI
196: 007C RENTER
197: 007C E1 POP HL ; POP OFF PC

```

```

199:          ; NAME: MCALL
200:          ; PURPOSE: CALL INTERPRETER SUBROUTINE
201:          ; INPUT: HL = ROUTINE ADDRESS
202:          ; NOTES: ROUTINE MAY BE CALLED FROM MACHINE LANGUAGE
203:          ; ANOTHER INTERPRETED SEQUENCE
204:          ; STACK DEPTH INCREASED BY 4 BY CALL
205: 007D 7E MMCALL LD A, (HL) ; GET OPCODE
206: 007E 23 INC HL
207: 007F CB3F SRL A
208: 0081 117C00 LD DE, RENTER ; LOAD INTERPRETER DUMMY RETURN
209: 0084 D5 MINT0 PUSH DE ; SAVE DUMMY RETURN
210: 0085 4F LD C, A ; INDEX TO C
211: 0086 3012 JR NC, $(MINT2-$) ; JUMP IF NO LOAD WANTED
212: 0088 EB EX DE, HL
213: 0089 0600 LD B, 0
214: 008B 214B01 LD HL, MRARGT ; LOAD SYSTEM ARG TABLE
215: 008E CB77 BIT 6, A ; USE USER TABLE?
216: 0090 2803 JR Z, $(MINT1-$) ; JUMP IS NO
217: 0092 2AFB4F LD HL, (UMARGT)
218: 0095 09 MINT1 ADD HL, BC ; INDEX TABLE
219: 0096 46 LD B, (HL)
220: 0097 CDA800 CALL MSUCK1 ; CALL SUCK ROUTINE
221: 009A D1 MINT2 POP DE ; DUMMY RETURN TO DE, HL = PC
222: 009B 79 LD A, C ; GET CALL INDEX BACK

```

Home Video Game System Library - System Subroutines

```

223: 009C FD4607      LD    B,(IY+CBB)    ; RESTORE CLOBBERED REGISTERS
224: 009F FD4E06      LD    C,(IY+CBC)
225: 00A2 18AA        JR    $+(INTPE-$)  ; JOIN NORMAL UPI DISPATCH SEQU

227:                ; NAME:           SUCK INLINE ARGUMENTS
228:                ; PURPOSE:        TRANSFER OF INLINE ARGS INTO CONTEXT BLO
229:                ; INPUT:         B = ARG LOAD MASK (SEE INTERPRETER COMME
230:                ; OUTPUT:        HL = UPDATED PC
231:                ; EXPLANATION: THIS ROUTINE IMPLEMENTS A MACRO LOAD INST
232:                ; IT IS USED BY THE INTERPRETER AS WELL.  A ONE BIT IN T
233:                ; INLINE LOAD MASK MEANS TRANSFER THE NEXT INLINE BYTE I
234:                ; A ZERO BIT MEANS 'ADVANCE CONTEXT BLOCK POINTER'
235:                ; TWO ENTRY POINTS ARE DEFINED, ONE FOR THE SUCK MACRO I
236:                ; THE OTHER FOR THE INTERPRETER TO USE
237:                ; SUCK MACRO ENTRY
238: 00A4 E1          MSUCK  POP HL          ; RETURN ADDRESS TO HL
239: 00A5 D1          MSUCK  POP DE          ; POP OFF PC
240:                ; BYTE SAVING TRICK *** REPLACE WITH LD HL,REENTRY
241: 00A6 23          MSUCK  INC HL           ; ADVANCE TO REENTRY (MINT0)
242: 00A7 E5          MSUCK  PUSH HL
243:                ; CALL INTO...
244: 00A8 CB60        MSUCK1 BIT 4,B        ; IX LOAD WANTED?
245: 00AA 280A        MSUCK1 JR Z,$+(MSUCK2-$) ; MSUCK2 IF NOT
246: 00AC 1A          MSUCK1 LD A,(DE)
247: 00AD 13          MSUCK1 INC DE
248: 00AE FD7702      MSUCK1 LD (IY+CBIXL),A
249: 00B1 1A          MSUCK1 LD A,(DE)
250: 00B2 13          MSUCK1 INC DE
251: 00B3 FD7703      MSUCK1 LD (IY+CBIXH),A
252: 00B6 FDE5        MSUCK2 PUSH IY          ; LET HL = IY
253: 00B8 E1          MSUCK2 POP HL
254: 00B9 23          MSUCK2 INC HL           ; + 4
255: 00BA 23          MSUCK2 INC HL
256: 00BB 23          MSUCK2 INC HL
257: 00BC 23          MSUCK2 INC HL
258: 00BD CBA0        MSUCK2 RES 4,B        ; KILL IX BIT
259:                ; SUCK IN LOOP
260: 00BF CB38        MSUCK3 SRL B
261: 00C1 3003        MSUCK3 JR NC,$+(MSUCK5-$) ; MSUCK5 IF NOT THIS TIME
262: 00C3 1A          MSUCK3 LD A,(DE)        ; GET INLINE BYTE
263: 00C4 13          MSUCK3 INC DE
264: 00C5 77          MSUCK3 LD (HL),A        ; STUFF INTO CB
265: 00C6 23          MSUCK5 INC HL          ; BUMP CB POINTER
266:                ; ** THIS CODE ASSUMES THAT STATUS OF 'SRL' IS PRESERVED
267: 00C7 20F6        MSUCK5 JR NZ,$+(MSUCK3-$) ; JUMP BACK IF MORE TO DO
268: 00C9 EB          MSUCK5 EX DE,HL        ; HL = PC
269: 00CA C9          MSUCK5 RET           ; THEN QUIT
271:                ; *****
272:                ; * UPI ROUTINE ADDRESS TABLE *
273:                ; *****
274: 00CB 7B00        SYSDPT DW MINTPC
275: 00CD 7902        SYSDPT DW MXINTC
276: 00CF 3206        SYSDPT DW MRCALL
277: 00D1 7D00        SYSDPT DW MMCALL
278: 00D3 730B        SYSDPT DW MMRET
279: 00D5 C40A        SYSDPT DW MMJUMP
280: 00D7 A400        SYSDPT DW MSUCK

```

Home Video Game System Library - System Subroutines

```

281: 00D9 8B01          DW  MACTIN
282: 00DB 7E04          DW  TIMEY
283: 00DD 0805          DW  MUZSET
284: 00DF FC05          DW  MUZSTP
285: 00E1 CF03          DW  MSETUP
286: 00E3 DB01          DW  MCOLOR
287: 00E5 EE0A          DW  MFILL
288: 00E7 B206          DW  MPAINTE
289: 00E9 FE06          DW  MVWRIT
290: 00EB 0B07          DW  MWRITR
291: 00ED 1507          DW  MWRITP
292: 00EF 1907          DW  MWRIT
293: 00F1 1C07          DW  MWRITA
294: 00F3 7D07          DW  MVBLAN
295: 00F5 9E07          DW  MBLANK
296: 00F7 B903          DW  MSAVE
297: 00F9 AD07          DW  MREST
298: 00FB 6A02          DW  MSCROL
299: 00FD E107          DW  DISPCH
300: 00FF C407          DW  STRNEW
301: 0101 EB0B          DW  BCDISP
302: 0103 F60A          DW  MRELAB
303: 0105 FB0A          DW  MRELA1      ; RELAB1
304: 0107 5606          DW  MVECTC
305: 0109 3306          DW  MVECT
306: 010B C90A          DW  MKCTAS
307: 010D AC01          DW  MENTRY      ; SENTRY
308: 010F 0C06          DW  MDOIT       ; DOIT
309: 0111 0B06          DW  MDOITB
310: 0113 BA01          DW  MPIZBK      ; PIZBRK
311: 0115 970C          DW  MMENU
312: 0117 FB0C          DW  MGETP
313: 0119 310D          DW  MGETN
314: 011B 1B00          DW  MPAUSE      ; PAUSE
315: 011D CC0B          DW  MDISTI      ; DISPLAY TIME
316: 011F 150C          DW  MINCSC      ; INC SCORE
317: 0121 760B          DW  INXNIB      ; INDEXN
318: 0123 900B          DW  PUTNIB      ; STOREN
319: 0125 AC0B          DW  MINDW       ; INDEXW
320: 0127 BD0B          DW  MINDB       ; INDEXB
321: 0129 4B0B          DW  MMOVE       ; MOVE
322: 012B AA0D          DW  MSHFTU
323: 012D 2103          DW  BCDAD
324: 012F 1F03          DW  BCDSB
325: 0131 DE02          DW  BCDML
326: 0133 8402          DW  BCDDV
327: 0135 6403          DW  BCDCS
328: 0137 4103          DW  BCDNG
329: 0139 6E03          DW  SDADD
330: 013B 2903          DW  SDSMG
331: 013D 5603          DW  SDABS
332: 013F 4C03          DW  SNEGT
333: 0141 7F03          DW  MRANGE
334: 0143 410C          DW  MQUIT
335: 0145 6C03          DW  MSETB
336: 0147 2300          DW  MSETW
337: 0149 4002          DW  MMTD

```

Home Video Game System Library - System Subroutines

```

339:          ; MACRO ROUTINES ARGUMENT MASK TABLE
340:          ; FORMAT:
341:          ; *****
342:          ; * 7 * 6 * 5 * 4 * 3 * 2 * 1 * 0 *
343:          ; *****
344:          ; * H * L * A * IX* B * C * D * E *
345:          ; *****
346:          ; ARGUMENTS MUST FOLLOW THE CALL INDEX IN THE FOLLOWING
347:          ; (OMITTING UNUSED ARGUMENTS, OF COURSE)
348:          ; (INDEX), IXL, IXH, E,D,C,B,A,L,H
349: 014B 00          MRARGT  DB    0          ; INTPC
350: 014C 00          DB    0          ; XINTC
351: 014D C0          DB   11000000B       ; RCALL
352: 014E C0          DB   11000000B       ; MCALL
353: 014F 00          DB    0          ; MRET
354: 0150 C0          DB   11000000B       ; MJUMP
355: 0151 08          DB   00001000B       ; SUCK
356: 0152 00          DB    0          ; ACTINT
357: 0153 04          DB   00000100B       ; DECCTS
358: 0154 F0          DB   11110000B       ; BMUSIC
359: 0155 00          DB    0          ; EMUSIC
360: 0156 2A          DB   00101010B       ; SETOUT
361: 0157 C0          DB   11000000B       ; COLSET
362: 0158 2F          DB   00101111B       ; FILL
363: 0159 2F          DB   00101111B       ; RECTAN
364: 015A D0          DB   11010000B       ; VWRITR
365: 015B E3          DB   11100011B       ; WRITR
366: 015C E3          DB   11100011B       ; WRITP
367: 015D EF          DB   11101111B       ; WRIT
368: 015E EF          DB   11101111B       ; WRITA
369: 015F 13          DB   00010011B       ; VBLANK
370: 0160 CB          DB   11001011B       ; BLANK
371: 0161 CF          DB   11001111B       ; SAVE
372: 0162 C3          DB   11000011B       ; RESTORE
373: 0163 CF          DB   11001111B       ; SCROLL
374: 0164 27          DB   00100111B       ; NEW DISCHR
375: 0165 C7          DB   11000111B       ; NEW DISSTR
376: 0166 CF          DB   11001111B       ; DISNUM
377: 0167 20          DB   00100000B       ; RELABS
378: 0168 20          DB   00100000B       ; RELAB1
379: 0169 D4          DB   11010100B       ; VECTC
380: 016A D0          DB   11010000B       ; VECT
381: 016B 00          DB    0          ; KCTASC
382: 016C 03          DB   00000011B       ; SENTRY
383: 016D C0          DB   11000000B       ; DOIT
384: 016E C0          DB   11000000B       ; DOITB
385: 016F 00          DB    0          ; PIZBRK
386: 0170 C3          DB   11000011B       ; MENU
387: 0171 EC          DB   11101100B       ; GET PARAMETER
388: 0172 CF          DB   11001111B       ; GET NUMBER
389: 0173 08          DB   00001000B       ; PAUSE
390: 0174 07          DB   00000111B       ; DISTIM
391: 0175 C0          DB   11000000B       ; INCSCR

```

Home Video Game System Library - System Subroutines

```

392: 0176 C0          DB 11000000B    ; INDEXN
393: 0177 C0          DB 11000000B    ; STOREN
394: 0178 C0          DB 11000000B    ; INDEXW
395: 0179 C0          DB 11000000B    ; INDEXB
396: 017A CF          DB 11001111B    ; MOVE
397: 017B C8          DB 11001000B    ; SHIFTU
398: 017C CB          DB 11001011B    ; BCDADD
399: 017D CB          DB 11001011B    ; BCDSUB
400: 017E CB          DB 11001011B    ; BCDMUL
401: 017F CB          DB 11001011B    ; BCDDIV
402: 0180 C8          DB 11001000B    ; BCDCHS
403: 0181 0B          DB 00001011B    ; BCDNEG
404: 0182 CB          DB 11001011B    ; DADD
405: 0183 0B          DB 00001011B    ; DSMG
406: 0184 0B          DB 00001011B    ; DABS
407: 0185 C8          DB 11001000B    ; NEG
408: 0186 20          DB 00100000B    ; RANGED
409: 0187 00          DB 00000000B    ; QUIT
410: 0188 E0          DB 11100000B    ; SET BYTE
411: 0189 C3          DB 11000011B    ; SET WORD
412: 018A C7          DB 11000111B    ; MASK TO DELTAS

```

```

414:                ; DOES 4 60TH SEC COUNTERS IN CT0-3
415: 018B F3          MACTIN DI          ; MAKE SURE INTERRUPT IS DISABL
416: 018C F5          PUSH AF
417: 018D C5          PUSH BC
418: 018E D5          PUSH DE
419: 018F E5          PUSH HL
420: 0190 ED5E        IM 2
421: 0192 3E00        LD A,ITAB SHR 8
422: 0194 ED47        LD I,A
423: 0196 3EC8        LD A,200
424: 0198 D30F        OUT (INLIN),A
425: 019A 3E34        LD A,ITAB & 0FFH
426: 019C D30D        OUT (INFBK),A
427: 019E CDA004      CALL TIMEZ          ; UPDATE TIMEOUT,MUSIC AND SECON
428: 01A1 0E0F        LD C,0FH          ; USE CT0-3
429: 01A3 CD7E04      CALL TIMEY          ; DEC CT0-3
430: 01A6 E1          POP HL
431: 01A7 D1          POP DE
432: 01A8 C1          POP BC
433: 01A9 F1          POP AF
434: 01AA FB          EI
435: 01AB C9          RET

```

```

437:                ; ROUTINE: SENTRY
438:                ; PURPOSE: TO WAIT FOR CHANGE OF PROGRAM STATUS
439:                ; IN EITHER THE PORTS OR THE TIMER-COUNTERS.
440:                ; IN ADDITION IT CHECKS TIMOUT FOR LONG, PERIODS OF IN-
441:                ; ACTIVITY.
442:                ; ** IS VECTOR OUT FLAG SET??

```

Home Video Game System Library - System Subroutines

```

443: 01AC 3AFA4F      MENTRY LD  A,(SENFLG)
444: 01AF FEAA          CP  0AAH
445: 01B1 CA1920          JP  Z,2019H      ; YES - JUMP OUT
446: 01B4 3AEC4F      LD  A,(TIMOUT)  ; CHECK IF TIME TO BLAKOUT
447: 01B7 B7              OR  A
448: 01B8 202B          JR  NZ,$+(TTEST-$)
449: 01BA AF            MPIZBK XOR A          ; TIME TO SHUT DOWN
450: 01BB F3              DI
451: 01BC D315          OUT (VOLC),A    ; TURN OFF SOUNDS
452: 01BE D316          OUT (VOLAB),A
453: 01C0 010B08       LD  BC,COLBX+8*256
454: 01C3 ED79          OUT (C),A      ; PAINT IT BLACK
455: 01C5 10FC          DJNZ ($-2)
456: 01C7 111402       PBLP LD  DE,AKEYS
457: 01CA CDF40C       CALL FINDL3     ; CALL STORE DE INTO CONTEXT RO
458: 01CD CDE501       CALL TTEST      ; WAIT FOR SOMETHING TO HAPPEN
459: 01D0 3C           INC  A
460: 01D1 20E7          JR  NZ,$+(MPIZBK-$)
461: 01D3 FD360900      LD  (IY+CBA),0
462: 01D7 FB           EI
463: 01D8 2AE84F       LD  HL,(COLLST) ; GET SAVED COLORS
464: 01DB 22E84F       MCOLOR LD  (COLLST),HL ; SAVE COLORS FOR FUTURE
465: 01DE 010B08       LD  BC,800H+COLBX
466: 01E1 EDB3          OTIR           ; RESET THE COLORS
467: 01E3 AF           XOR  A
468: 01E4 C9           RET
469: 01E5 CDEC03       TTEST CALL TRCHK
470: 01E8 FD7709       LD  (IY+CBA),A
471: 01EB FD7007       LD  (IY+CBB),B
472: 01EE FE13          CP  SKYD
473: 01F0 D8           RET  C
474: 01F1 FE1C          CP  POT0
475: 01F3 D0           RET  NC
476: 01F4 3EFF          LD  A,$FF
477: 01F6 32EC4F       LD  (TIMOUT),A
478: 01F9 C9           RET

```

```

480: 01FA C40D      CALCL  DW  SCBL
481: 01FC DD0D      DW  PNCALC
482: 01FE 2010     DW  CALCST      ; START OF CALCULATOR

```

```

484:          ; SYSTEM ROUTINES JUMP VECTOR
485:          ;          ORG 200H
486: 0200 C3A004     JP  TIMEZ      ; DO TIMER & MUSIC
487: 0203 C37B04     JP  TIMEX      ; DECTMR
488:
489: 0206 20        SYSFNT DB  20H
490: 0207 08        DB  8
491: 0208 08        DB  8
492: 0209 01        DB  1
493: 020A 07        DB  7
494: 020B E408      DW  LRGCHR
495:

```

Home Video Game System Library - System Subroutines

```

496: 020D A0          SMLFNT DB 0A0H
497: 020E 04          DB 4
498: 020F 06          DB 6
499: 0210 01          DB 1
500: 0211 05          DB 5
501: 0212 BF0A        DW SMLCHR
502:
503:                ; ALL KEYS MASK
504: 0214 3F          AKEYS DB 3FH
505: 0215 3F          DB 3FH
506: 0216 3F          DB 3FH
507: 0217 3F          DB 3FH
508:
509:                ; HEAD OF ONBOARD MENU
510: 0218 BE0D        GUNLNK DW CML
511: 021A CA0D        DW PNGF
512: 021C DE17        DW GFSTRT
513: 021E 4D415820    DEFM 'MAX SCORE'
514:                53434F52
515:                45
516: 0227 00          DB 0
517: 0228 23204F46    DEFM '# OF PLAYERS'
518:                20504C41
519:                59455253
520: 0234 00          DB 0
521: 0235 23204F46    DEFM '# OF GAMES'
522:                2047414D
523:                4553
524: 023F 00          DB 0
525:                ; NAME: CONVERT MASK TO DELTAS
526:                ; INPUT: B = JOYSTICK MASK
527:                ; C = FLOP STATUS (MR FLOP BIT SET IF FLOP
528:                ; DE = X POSITIVE DELTA
529:                ; HL = Y POSITIVE DELTA
530: 0240 CD5602      MMTD CALL CONCPL ; HANDLE Y
531: 0243 EB          EX DE,HL
532: 0244 CB71        BIT MRFLOP,C ; FLOP SET?
533: 0246 2807        JR Z,$+(MMTD2-$) ; YES - DOIT
534: 0248 78          LD A,B ; NO - GET MASK
535: 0249 E603        AND 3
536: 024B 2801        JR Z,$+(MMTD1-$) ; INVERT IF NOT ZERO
537: 024D 2F          CPL
538: 024E 47          MMTD1 LD B,A
539: 024F CD5602      MMTD2 CALL CONCPL ; PROCESS X
540: 0252 EB          EX DE,HL
541: 0253 C3B80B      JP STHLDE ; STORE HL,DE AND QUIT
542:
543:                ; SUBROUTINE TO CONDITIONALLY COMPLEMENT OR ZERO HL
544: 0256 CB08        CONCPL RRC B
545: 0258 300A        JR NC,$+(CONC1-$) ; JUMP IF NOT UP
546: 025A 7D          LD A,L
547: 025B 2F          CPL
548: 025C 6F          LD L,A
549: 025D 7C          LD A,H
550: 025E 2F          CPL
551: 025F 67          LD H,A

```

Home Video Game System Library - System Subroutines

```

547: 0260 23          INC  HL
548: 0261 CB08        RRC  B
549: 0263 C9          RET
550: 0264 CB08        RRC  B          ; DOWN SET?
551: 0266 D8          RET  C          ; QUIT IF SO
552: 0267 C32B00      JP   CONC2      ; JUMP TO ZERO OUT

554:                ; NAME:          SCROLL MEMORY BLOCK
555:                ; INPUT:          B = NUMBER OF LINES TO SCROLL
556:                ;                C = NUMBER OF BYTES ON LINE TO SCROLL
557:                ;                DE = LINE INCREMENT
558:                ;                HL = FIRST LINE TO SCROLL
559: 026A AF          MSCROL XOR  A
560: 026B C5          MSCRL1 PUSH BC          ; SAVE COUNTERS
561: 026C D5          MSCRL1 PUSH DE
562: 026D 47          MSCRL1 LD   B,A
563: 026E EB          MSCRL1 EX   DE,HL
564: 026F 19          MSCRL1 ADD  HL,DE          ; ADD INCREMENT TO LINE
565: 0270 E5          MSCRL1 PUSH HL
566: 0271 EDB0        MSCRL1 LDIR
567: 0273 E1          MSCRL1 POP  HL
568: 0274 D1          MSCRL1 POP  DE
569: 0275 C1          MSCRL1 POP  BC
570: 0276 10F3       MSCRL1 DJNZ $+(MSCRL1-$)
571: 0278 C9          MSCRL1 RET

573:                ; NAME:          MACRO INTERPRETER EXIT WITH CONTEXT REST
574:                ; PURPOSE:        QUIT INTERPRETING AND GO HOME
575: 0279 E1          MXINTC: POP  HL          ; THROW OUT DUMMY RETURN
576:                ; NAME:          RETURN FROM SYSTEM CALL
577:                ; PURPOSE:        RETURNING TO USER AND RESTORATION OF REG
578: 027A E1          RETN_  POP  HL          ; RETURN ADDRESS TO HL
579: 027B FDE1        RETN_  POP  IY
580: 027D DDE1        RETN_  POP  IX
581: 027F D1          RETN_  POP  DE
582: 0280 C1          RETN_  POP  BC
583: 0281 F1          RETN_  POP  AF
584: 0282 E3          RETN_  EX   (SP),HL          ;STK=RETURN, HL=OLD HL
585: 0283 C9          RETN_  RET

587:                ; NAME:          BCD DIVIDE
588:                ;
589: 0284 CDC002      BCDDV  CALL GNACC          ; GENERATE ACCUMULATOR
590: 0287 E3          BCDDV  EX   (SP),HL          ; HL = ACC, TOP = ARG2
591: 0288 C5          BCDDV  PUSH BC
592: 0289 0600       BCDDV  LD   B,0
593: 028B 79          BCDDV  LD   A,C
594: 028C CB39        BCDDV  SRL  C
595: 028E 09          BCDDV  ADD  HL,BC
596: 028F 4F          BCDDV  LD   C,A
597: 0290 EB          BCDDV  EX   DE,HL          ; HL = ARG1, DE = ACC
598: 0291 EDB0        BCDDV  LDIR          ; HL=ARG1 FLAG+1
599: 0293 C1          BCDDV  POP  BC
600: 0294 D1          BCDDV  POP  DE

```

Home Video Game System Library - System Subroutines

```

601: 0295 2B          DEC  HL          ;
602: 0296 E3          EX   (SP),HL     ; HL =ARG2, TOP = ARG1 FLAG
603: 0297 C5          PUSH BC
604: 0298 0600        LD   B,0
605: 029A 09          ADD  HL,BC       ; HL = ACC+SIZE/2
606: 029B C1          POP  BC
607: 029C 0D          DEC  C          ; DECREMENT SIZE
608: 029D EB          EX   DE,HL      ; HL = ARG2, DE=ACC, TOP=ARG
609: 029E 1B          DEC  DE
610: 029F 1B          DIV1 DEC  DE
611: 02A0 AF          XOR  A
612: 02A1          SYSTEM NEGTL ; ARG2 = -ARG2 (10S COMP)
612: 02A1 FF          RST   56
612: 02A2 74          DB   NEGTL
612: 0000          IF   NEGTL = INTPC
612:          ENDIF
612: 02A3          ENDM
612:
613: 02A3          DIV2 SYSTEM DADD ; SUBTRACT UNTIL BORROW
613: 02A3 FF          RST   56
613: 02A4 6E          DB   DADD
613: 0000          IF   DADD = INTPC
613:          ENDIF
613: 02A5          ENDM
613:
614: 02A5 380A         JR   C,$+(DIV3-$)
615: 02A7 3C          INC  A          ; OR UNTIL LOOP COUNT > 99
616: 02A8 27          DAA
617: 02A9 20F8        JR   NZ,$+(DIV2-$)
618: 02AB E1          POP  HL
619: 02AC 36FF        LD   (HL),$FF
620: 02AE C1          POP  BC
621: 02AF 186A         JR   $+(MULT6-$)
622: 02B1          DIV3 SYSTEM NEGTL
622: 02B1 FF          RST   56
622: 02B2 74          DB   NEGTL
622: 0000          IF   NEGTL = INTPC
622:          ENDIF
622: 02B3          ENDM
622:
623: 02B3          SYSTEM DADD
623: 02B3 FF          RST   56
623: 02B4 6E          DB   DADD
623: 0000          IF   DADD = INTPC
623:          ENDIF
623: 02B5          ENDM
623:
624: 02B5 E3          EX   (SP),HL     ; HL = ARG1
625: 02B6 2B          DEC  HL
626: 02B7 77          LD   (HL),A     ; SAVE ANSWER IN ARG1
627: 02B8 E3          EX   (SP),HL
628: 02B9 0D          DEC  C
629: 02BA 20E3        JR   NZ,$+(DIV1-$)
630: 02BC E1          POP  HL
631: 02BD C1          POP  BC
632: 02BE 1855         JR   $+(DIV4-$)
633:          ; SUBROUTINE TO GENERATE ACCUMULATOR ON THE STACK

```

Home Video Game System Library - System Subroutines

```

634: 02C0 DDE1      GNACC  POP  IX
635: 02C2 AF          XOR  A
636: 02C3 4F          LD   C,A
637: 02C4          SYSTEM DABS      ; ARG1= ABS VALUE
637: 02C4 FF          RST  56
637: 02C5 72          DB   DABS
637: 0000          IF   DABS = INTPC
637:          ENDIF
637: 02C6          ENDM
637:
638: 02C6 EB          EX  DE,HL
639: 02C7          SYSTEM DABS      ; ARG2= ABS VALUE
639: 02C7 FF          RST  56
639: 02C8 72          DB   DABS
639: 0000          IF   DABS = INTPC
639:          ENDIF
639: 02C9          ENDM
639:
640: 02C9 EB          EX  DE,HL      ; FLAG=1 IF NEG ANS, ELSE POS
641: 02CA 67          LD  H,A
642: 02CB 6F          LD  L,A
643: 02CC 78          LD  A,B
644: 02CD E5          MULT1 PUSH HL      ; GENERATE ACC ON STACK
645: 02CE 10FD        DJNZ $+(MULT1-$)
646: 02D0 47          LD  B,A      ; RESTORE SIZE
647: 02D1 39          ADD HL,SP
648: 02D2 C5          PUSH BC      ; SAVE SIGN
649: 02D3 E5          PUSH HL      ; SAVE STACK POINTER
650: 02D4 E5          PUSH HL      ; SAVE ACC POINTER
651: 02D5 FD660B     LD  H,(IY+CBH) ; RESTORE ARG2 POINTER
652: 02D8 FD6E0A     LD  L,(IY+CBL)
653: 02DB 48          LD  C,B
654: 02DC DDE9          JP  (IX)
655:          ; DECIMAL MULTIPLY
656:          ; GIVEN:   DE>ARG1, HL>ARG2, B=SIZE/2
657:          ;          (SIZE/2-1 ASSUMED EVEN)
658:          ; RETURNED: ARG1=ANSWER, C>0 ON OVERFLOW
659:          ;
660:          ;
661: 02DE CDC002     BCDML CALL GNACC      ; GENERATE ACCUM
662: 02E1 7E          MULT2 LD  A,(HL)      ; A=MULT LOOP COUNT
663: 02E2 23          INC  HL
664: 02E3 E3          EX  (SP),HL    ; HL>DEC ACC
665: 02E4 A7          AND  A          ; IF A=0, SKIP MULT LOOP
666: 02E5 2809        JR  Z,$+(MULT4-$)
667: 02E7 EB          EX  DE,HL
668: 02E8          MULT3 SYSTEM DADD      ; ELSE MULTIPLY
668: 02E8 FF          RST  56
668: 02E9 6E          DB   DADD
668: 0000          IF   DADD = INTPC
668:          ENDIF
668: 02EA          ENDM
668:
669: 02EA A7          AND  A          ; CLEAR THE CARRY BIT
670: 02EB 3D          DEC  A          ; DECIMAL DECREMENT
671: 02EC 27          DAA
672: 02ED 20F9        JR  NZ,$+(MULT3-$)

```

Home Video Game System Library - System Subroutines

```

673: 02EF EB          EX   DE,HL
674: 02F0 23          MULT4 INC  HL          ; INCREMENT DECIMAL ACC
675: 02F1 E3          EX   (SP),HL      ; HL>ARG2
676: 02F2 0D          DEC  C
677: 02F3 20EC        JR   NZ,$+(MULT2-$)
678: 02F5 E1          POP  HL          ; RESTORE STACK POINTER
679: 02F6 E1          POP  HL          ; RESTORE SIGN
680: 02F7 C1          POP  BC
681: 02F8 D5          PUSH DE
682: 02F9 C5          PUSH BC
683: 02FA 48          LD   C,B
684: 02FB 0600        LD   B,0
685: 02FD CB39        SRL  C
686: 02FF 09          ADD  HL,BC
687: 0300 CB21        SLA  C
688: 0302 EDB0        LDIR
689: 0304 C1          POP  BC
690: 0305 C5          PUSH BC          ; CHECK FOR OVERFLOW
691: 0306 CB38        SRL  B
692: 0308 AF          XOR  A
693: 0309 B6          MULT5 OR   (HL)
694: 030A 23          INC  HL
695: 030B 10FC        DJNZ $+(MULT5-$)
696: 030D A7          AND  A          ; SET FLAGS
697: 030E 2803        JR   Z,$+(MULT7-$)
698: 0310 3EFF        LD   A,$FF
699: 0312 12          LD   (DE),A
700: 0313 C1          MULT7 POP  BC          ; CHECK SIGN AND
701: 0314 E1          POP  HL
702: 0315 CB41        DIV4 BIT  0,C          ; NEGATE ARG1 IF NECESSARY
703: 0317 2802        JR   Z,$+(MULT6-$)
704: 0319          SYSTEM BCDCHS
704: 0319 FF          RST   56
704: 031A 6A          DB   BCDCHS
704: 0000          IF   BCDCHS = INTPC
704:          ENDIF
704: 031B          ENDM
704:
705: 031B E1          MULT6 POP  HL          ; RESTORE ORIGINAL STACK POINTER
706: 031C 10FD        DJNZ $+(MULT6-$)
707: 031E C9          RET
708:          ; BCD SUBTRACT & ADD
709:          ;
710:          ; GIVEN:      DE>ARG1, HL>ARG2
711:          ;          B=SIZE/2+1
712:          ; RETURNED: ARG1=ANSWER
713: 031F BCDSB SYSTEM BCDCHS
713: 031F FF          RST   56
713: 0320 6A          DB   BCDCHS
713: 0000          IF   BCDCHS = INTPC
713:          ENDIF
713: 0321          ENDM
713:
714: 0321 BCDAD SYSTEM BCDNEG
714: 0321 FF          RST   56
714: 0322 6C          DB   BCDNEG
714: 0000          IF   BCDNEG = INTPC

```

Home Video Game System Library - System Subroutines

```

714:                                ENDIF
714: 0323                                ENDM
714:
715: 0323  EB                                EX  DE,HL
716: 0324                                SYSTEM BCDNEG
716: 0324  FF                                RST  56
716: 0325  6C                                DB   BCDNEG
716: 0000                                IF   BCDNEG = INTPC
716:                                ENDIF
716: 0326                                ENDM
716:
717: 0326  EB                                EX  DE,HL
718: 0327                                SYSTEM DADD
718: 0327  FF                                RST  56
718: 0328  6E                                DB   DADD
718: 0000                                IF   DADD = INTPC
718:                                ENDIF
718: 0329                                ENDM
718:
719:                                ; AND FALL INTO
720:                                ;
721:                                ;
722:                                ; DECIMAL SIGNED MAGNITUDE
723:                                ;
724:                                ; GIVEN:      DE>ARG (10'S COMPLEMENT)
725:                                ;                B=SIZE/2+1
726:                                ; RETURN:     ARG (SIGNED MAGNITUDE)
727:                                ;
728: 0329  68  SDSMG  LD  L,B                ; HL>ARG3-1 (SIGN BYTE)
729: 032A  2D                                DEC  L
730: 032B  2600                               LD  H,0
731: 032D  19                                ADD  HL,DE
732: 032E  7E                                LD  A,(HL)                ; IF POS (SIGN NIBBLE<5)
733: 032F  FE50                               CP  $50
734: 0331  D8                                RET  C                    ; EXIT
735: 0332  EB                                EX  DE,HL
736: 0333  3E00  SDSMG1 LD  A,0                ; ELSE 10'S COMPLIMENT
737: 0335  9E                                SBC  A,(HL)
738: 0336  27                                DAA
739: 0337  77                                LD  (HL),A
740: 0338  23                                INC  HL
741: 0339  10F8                               DJNZ $+(SDSMG1-$)
742: 033B  2B                                DEC  HL                    ; AND SET SIGN BIT
743: 033C  7E                                LD  A,(HL)
744: 033D  F680                               OR  $80
745: 033F  77                                LD  (HL),A
746: 0340  C9                                RET
747:                                ;
748:                                ;
749:                                ; BCD NEGATE
750:                                ;
751:                                ; GIVEN:      DE>ARG (SIGNED MAGNITUDE)
752:                                ;                B=SIZE/2+1
753:                                ; RETURNED:  ARG (10'S COMPLEMENT)
754:                                ;
755: 0341  68  BCDNG  LD  L,B                ; HL>ARG+B-1 (SIGN BYTE)
756: 0342  2D                                DEC  L

```

Home Video Game System Library - System Subroutines

```

757: 0343 2600          LD    H,0
758: 0345 19           ADD   HL,DE
759: 0346 CB7E          BIT   7,(HL)          ; EXIT IF POS
760: 0348 C8           RET   Z
761: 0349 3600          LD    (HL),0          ; CLEAR SIGN BYTE
762: 034B EB           EX   DE,HL
763: 034C AF           SNEGT XOR A          ; CLEAR CARRY
764: 034D 3E00          BCDNG1 LD A,0          ; ELSE 10'S COMPLEMENT
765: 034F 9E           SBC  A,(HL)
766: 0350 27           DAA
767: 0351 77           LD    (HL),A
768: 0352 23           INC  HL
769: 0353 10F8          DJNZ  $+(BCDNG1-$)
770: 0355 C9           RET
771:                ;
772:                ;
773:                ; DECIMAL ABSOLUTE
774:                ;
775:                ; GIVEN:    DE>ARG (SIGNED MAGNITUDE)
776:                ;          B=SIZE/2+1
777:                ; RETURNED: C=C+1 IF SIGN BIT CLEARED
778:                ;
779: 0356 68           SDABS LD  L,B
780: 0357 2600          LD    H,0
781: 0359 2D           DEC  L
782: 035A 19           ADD  HL,DE
783: 035B CB7E          BIT   7,(HL)
784: 035D C8           RET   Z
785: 035E 3600          LD    (HL),0
786: 0360 FD3406        INC  (IY+CBC)
787: 0363 C9           RET
788:                ;
789:                ;
790:                ; BCD CHANGE SIGN
791:                ;
792:                ; GIVEN:    HL>ARG B=SIZE/2+1
793:                ;          (SIGNED MAGNITUDE)
794:                ; RETURNED: ARG SIGN BIT COMPLEMENTED
795:                ;
796: 0364 48           BCDCS LD  C,B
797: 0365 0600          LD    B,0
798: 0367 0D           DEC  C
799: 0368 09           ADD  HL,BC
800: 0369 7E           LD   A,(HL)
801: 036A EE80          XOR  $80
802:                ; NAME:    SET BYTE
803: 036C 77           MSETB LD (HL),A
804: 036D C9           RET
805:                ;
806:                ;
807:                ; DECIMAL ADD
808:                ;
809:                ;
810:                ; GIVEN:    DE>ARG1 HL>ARG2 (10'S COMPLEMENT)
811:                ;          B=SIZE/2+1
812:                ; RETURNED: ARG1=ANSWER (10'S COMPLEMENT)
813:                ;

```

Home Video Game System Library - System Subroutines

```

814: 036E AF          SDADD  XOR  A
815: 036F 1A          SDADD1 LD  A,(DE)
816: 0370 8E          ADC   A,(HL)
817: 0371 27          DAA
818: 0372 12          LD   (DE),A
819: 0373 13          INC  DE
820: 0374 23          INC  HL
821: 0375 10F8        DJNZ  $+(SDADD1-$)
822: 0377 FE99        CP   99H
823: 0379 17          RLA
824: 037A 2F          CPL
825: 037B FD7708      LD   (IY+CBFLAG),A ; SEND BACK STATUS FROM DADD
826: 037E C9          RET

```

```

828:           ; NAME:           RANGED RANDOM NUMBER
829:           ; INPUT:          A = RANGE
830:           ; OUTPUT:        A = RANDOM NUMBER (0 TO RANGE-1)
831: 037F F5          MRANGE PUSH AF
832: 0380 2AEF4F      LD   HL,(RANSHT)
833: 0383 CDAC03      CALL SHIFTR
834: 0386 011700      LD   BC,23
835: 0389 09          ADD  HL,BC
836: 038A 8A          ADC  A,D
837: 038B 22EF4F      LD   (RANSHT),HL
838: 038E 2AF14F      LD   HL,(RANSHT+2)
839: 0391 5F          LD   E,A
840: 0392 CDAC03      CALL SHIFTR
841: 0395 19          ADD  HL,DE
842: 0396 22F14F      LD   (RANSHT+2),HL
843: 0399 5A          LD   E,D
844: 039A EB          EX  DE,HL
845: 039B F1          POP  AF
846: 039C A7          AND  A
847: 039D 4F          LD   C,A
848: 039E 7A          LD   A,D
849: 039F 2808        JR   Z,$+(R3-$)
850: 03A1 AF          XOR  A
851: 03A2 19          R1   ADD  HL,DE
852: 03A3 3001        JR   NC,$+(R2-$)
853: 03A5 3C          INC  A
854: 03A6 0D          R2   DEC  C
855: 03A7 20F9        JR   NZ,$+(R1-$)
856: 03A9 C3D10A        R3   JP   QFROG
857: 03AC 44          SHIFTR LD  B,H
858: 03AD 4D          LD  C,L
859: 03AE AF          XOR  A
860: 03AF 1607        LD  D,7
861: 03B1 29          SH1  ADD  HL,HL
862: 03B2 17          RLA
863: 03B3 15          DEC  D
864: 03B4 20FB        JR   NZ,$+(SH1-$)
865: 03B6 09          ADD  HL,BC
866: 03B7 8A          ADC  A,D
867: 03B8 C9          RET

```

Home Video Game System Library - System Subroutines

```

869:          ; NAME:          SAVE AREA
870:          ; INPUT:        HL = SCREEN ADDRESS
871:          ;                DE = SAVE AREA ADDRESS
872:          ;                BC = Y,X SIZE OF AREA TO SAVE
873:          ; NOTES:        THE SIZES OF THE OBJECT ARE SAVED IN THE
874:          ;                FIRST TWO BYTES OF THE SAVE AREA
875: 03B9 EB          MSAVE EX DE,HL
876: 03BA 71          LD (HL),C          ; SET X SIZE
877: 03BB 23          INC HL
878: 03BC 70          LD (HL),B          ; SET Y SIZE
879: 03BD 23          INC HL
880: 03BE AF          XOR A
881: 03BF EB          EX DE,HL
882: 03C0 CBF4        SET 6,H          ; SET NON-MAGIC ADDRESS
883: 03C2 C5          MSAVE1 PUSH BC
884: 03C3 E5          PUSH HL
885: 03C4 47          LD B,A
886: 03C5 EDB0        LDIR
887: 03C7 E1          POP HL
888: 03C8 0E28        LD C,BYTEPL
889: 03CA 09          ADD HL,BC
890: 03CB C1          POP BC
891: 03CC 10F4        DJNZ $+(MSAVE1-$)
892: 03CE C9          RET

894:          ; NAME:          PREGAME OUTPUT PORT SETUP
895:          ; PURPOSE:       TO SET CONCOM, VERBL, ETC
896:          ; INPUTS:       B=HORCB, D=VERBL, A=INMOD
897: 03CF 0E09        MSETUP: LD C,HORCB          ; GET BASE PORT NUMBER
898: 03D1 ED41        OUT (C),B          ; HORBL
899: 03D3 0C          INC C          ;
900: 03D4 ED51        OUT (C),D          ; VERBL
901: 03D6 D30E        OUT (INMOD),A
902: 03D8 C9          RET

904:          ; NAME:          TEST FOR TRANSITIONS
905:          ; FUNCTION:     TO LOOK FOR CHANGES IN THE PORTS %TC.
906:          ; RETURNS:     A=0 NO CHANGE
907:          ; 1-8 COUNTER TIMER #N HIT 0
908:          ; 9-C = POT0-3 CHANGED
909:          ; D = A SECONDS UP
910:          ; E= KEYBOARD CHANGED (B=0-24)
911:          ; F-16 : TRIG0!JOY0 - T3!J3
912:          ; RETURNS NEW VALUE IN B
913: 03D9 5E          CTLP LD E,(HL)
914: 03DA 010108      LD BC,$801
915: 03DD 79          CCTLP LD A,C          ; GET MASK
916: 03DE 0F          RRCA
917: 03DF 4F          LD C,A
918: 03E0 A3          AND E          ; CHECK IF CT BIT =1
919: 03E1 2003        JR NZ,$+(CCT1-$)
920: 03E3 10F8        DJNZ $+(CCTLP-$)
921: 03E5 C9          RET
922: 03E6 AB          CCT1 XOR E          ; MASK OUT BIT IN QUESTION
923: 03E7 77          LD (HL),A          ; PUT BACK THE CTFLAGS OR SEMI4

```

Home Video Game System Library - System Subroutines

```

924: 03E8 78          LD    A,B
925: 03E9 82          ADD   A,D
926: 03EA E1          POP   HL          ; OLD RETURN ADDRESS
927: 03EB C9          RET
928: 03EC 2825        TRCHK JR    Z,$+(TSEX-$) ; SKIP COUNTER-TIMERS AND POTS?
929: 03EE 21DD4F      LD    HL,CNT      ; GET COUNTER TIMERS STATUS
930: 03F1 1600        LD    D,0
931: 03F3 CDD903      CALL CTLP        ; COUNTER TIMERS
932: 03F6 1608        LD    D,8
933: 03F8 23          INC   HL
934: 03F9 CDD903      CALL CTLP        ; SEMI4S
935: 03FC 011C04      LD    BC,$400+POT0
936: 03FF 23          TPLOP INC   HL          ; -> MPOT0
937: 0400 ED78        IN    A,(C)
938: 0402 5E          LD    E,(HL)     ; GET 0POT
939: 0403 93          SUB   E
940: 0404 3805        JR    C,$+(PHOT-$) ; NEW LINE LESS THAN OLD
941: 0406 D608        SUB   PFUG        ; FUDGE BOUNCE FACTOR
942: 0408 3806        JR    C,$+(EPLOP-$) ; NEW MORE THAN OLD+4
943: 040A 3C          INC   A
944: 040B 83          PHOT  ADD   A,E
945: 040C 77          LD    (HL),A
946: 040D 47          LD    B,A
947: 040E 79          LD    A,C
948: 040F C9          RET
949: 0410 0C          EPLOP INC   C
950: 0411 10EC        DJNZ  $+(TPLOP-$)
951:                ; NOW TEST SECONDS
952: 0413 21E34F      TSEX: LD    HL,KEYSEX ; HL = KEYSEX
953: 0416 7E          LD    A,(HL)
954: 0417 CB7F        BIT   7,A
955: 0419 2806        JR    Z,$+(TKEYS-$)
956: 041B CBBF        RES   7,A
957: 041D 77          LD    (HL),A
958: 041E 3E11        LD    A,SSEC     ; SECS
959: 0420 C9          RET
960:                ; NOW TEST KEYBOARD
961: 0421 E5          TKEYS PUSH HL
962: 0422 CD7400      CALL DELOAD
963: 0425 EB          EX    DE,HL
964: 0426 011704      LD    BC,$400+KEY3
965: 0429 1100FF      LD    DE,$FF00   ; SET BIT COUNTER+COLUMNN
966: 042C ED78        MSK1  IN    A,(C)
967: 042E A6          AND   (HL)
968: 042F 200A        JR    NZ,$+(MSENK2-$)
969: 0431 0D          DEC   C          ; NEXT PORT
970: 0432 1C          INC   E          ; AND COLUMN
971: 0433 23          INC   HL         ; AND MASK
972: 0434 10F6        DJNZ  $+(MSK1-$)
973: 0436 78          LD    A,B        ; NOTHING DOWN
974: 0437 1E12        LD    E,SKYU
975: 0439 180B        JR    $+(MSENKE-$)
976: 043B 14          MSENK2 INC   D          ; BIT COUNTER
977: 043C 0F          RRCA
978: 043D 30FC        JR    NC,$+(MSENK2-$)
979: 043F 7A          LD    A,D
980: 0440 07          RLCA          ; KEY=BIT*4

```

Home Video Game System Library - System Subroutines

```

981: 0441 07          RLCA
982: 0442 83          ADD  A,E           ; + COLUMN
983: 0443 3C          INC  A             ; PLUS 1
984: 0444 1E13        LD   E,SKYD
985: 0446 E1          MSENKE POP  HL
986: 0447 AE          XOR  (HL)          ; KEY=0KEY?
987: 0448 E67F        AND  $7F
988: 044A 2807        JR   Z,$+(HANDLE-$)
989: 044C AE          XOR  (HL)
990: 044D 77          LD   (HL),A
991: 044E E67F        AND  $7F
992: 0450 47          LD   B,A
993: 0451 7B          LD   A,E           ; KEYBOARD RETURN CODE
994: 0452 C9          RET
995:                ; NOW TEST HANDLES
996: 0453 011004       HANDLE LD  BC,$400+SW0
997: 0456 23          SWLOP INC  HL       ; -> OSW0
998: 0457 ED78        IN   A,(C)
999: 0459 AE          XOR  (HL)          ; COMPARE THE TWO
1000: 045A 2005        JR   NZ,$+(SWHIT-$) ; NO CHANGE
1001: 045C 0C          INC  C
1002: 045D 10F7        DJNZ $+(SWLOP-$)
1003: 045F 78          LD   A,B           ; RETURN 0
1004: 0460 C9          RET
1005: 0461 CB67        SWHIT BIT  4,A       ; TEST TRIGGER
1006: 0463 280C        JR   Z,$+(JOYS-$) ; NO TRIG MUST BE JOYSTICK
1007: 0465 E610        AND  $10           ; FILTER OUT TRIGGER
1008: 0467 AE          XOR  (HL)          ; UPDATE VALUE
1009: 0468 77          LD   (HL),A
1010: 0469 E610        AND  $10
1011: 046B 47          LD   B,A
1012: 046C 79          LD   A,C           ; GET PORT NUMBER
1013: 046D 07          RLCA              ; *2
1014: 046E D60C        SUB  $C
1015: 0470 C9          RET
1016: 0471 AE          JOYS XOR  (HL)
1017: 0472 77          LD   (HL),A       ; NO CHANGE IN TRIG SO STORE ST
1018: 0473 E60F        AND  $F           ; TAKE OFF TRIGGER
1019: 0475 47          LD   B,A
1020: 0476 79          LD   A,C
1021: 0477 07          RLCA              ; *2
1022: 0478 D60B        SUB  $B
1023: 047A C9          RET

1025:                ; TIMEX
1026:                ; INPUTS HL-> TIME BASE IN RAM
1027:                ; B=TIME BASE MODULUS
1028:                ; C=MASK AS IN DECCTS
1029:                ; PURPOSE: TO DECR TIMEBASE AND IF 0 RESET IF AND DECR
1030:                ; COUNTER          TIMERS
1031: 047B 35          TIMEX DEC  (HL)       ; DEC TIMEBASE
1032: 047C C0          RET  NZ
1033: 047D 70          LD   (HL),B       ; RESET TIMEBASE

1035:                ; NAME: DECREMENT COUNTER          TIMERS

```

Home Video Game System Library - System Subroutines

```

1036:           ; INPUTS: C=MASK
1037:           ; USED BY ACTINT AND DECCTS TO DECREASEMENTS CTS UNDER MASK
1038:           ; MASK = *76543210* , IF BIT=1 THEN DEC CORRESPONDING
1039:           ; CT# , IF BIT=0 LEAVE CT# ALONE
1040:           ; NOTE: ALL COUNTERS ARE RUN IN BCD FOR EASY DISPLAY
1041: 047E 0608           TIMEY  LD   B,8           ; NO OF BITS
1042: 0480 21D54F        LD   HL,CT0        ; -> TO COUNTER TIMERS
1043: 0483 1600           LD   D,0           ; RESULTS
1044: 0485 CB39           TIMLP  SRL  C           ; CHANGE THIS TIMER?
1045: 0487 300A           JR   NC,$+(ETLP-$)
1046: 0489 7E             LD   A,(HL)        ; GET THE TIMER
1047: 048A B7             OR   A           ; IS IT ZERO ALREADY?
1048: 048B 2806           JR   Z,$+(ETLP-$)
1049: 048D 3D             DEC  A
1050: 048E 27             DAA
1051: 048F 2001           JR   NZ,($+3)
1052: 0491 37             SCF
1053: 0492 77             LD   (HL),A        ; STORE NEW VALUE
1054: 0493 23           ETLP  INC  HL
1055: 0494 CB1A           RR   D           ; ROTATES IN CARRY FLAG
1056: 0496 10ED           DJNZ $+(TIMLP-$)
1057: 0498 3ADD4F        LD   A,(CNT)        ; COUNTER UPDATE & NUMBER TRACKER
1058: 049B B2             OR   D
1059: 049C 32DD4F        LD   (CNT),A
1060: 049F C9             RET

```

```

1062:           ; NAME: TIMER ROUTINE
1063:           ; PURPOSE: TO UPDATE GAME TIME, TIMEOUT AND MUSIC
1064:           ; INPUTS:           OUTPUTS: NONE
1065:           ; NOTE: PUSH YOUR REGISTERS (AF,BC,DE,HL)
1066: 04A0           TIMEZ
1067: 04A0 21F94F        LD   HL,PRIOR        ; ASSUMES YOU PUSH DA REGS
1068: 04A3 CB4E           BIT  1,(HL)        ; CHECK IF TICKS OVERRUN
1069: 04A5 C0             RET  NZ
1070: 04A6 CBCE           SET  1,(HL)
1071: 04A8 EB             EX  DE,HL
1072:           ; *SIXTIETH OF A SECOND INTERRUPT*
1073: 04A9 21EA4F        LD   HL,DURAT        ; NOTE TIMER
1074: 04AC 7E             LD   A,(HL)        ; =0 SKIP
1075: 04AD B7             OR   A
1076: 04AE 281C           JR   Z,$+(SIXY-$)
1077: 04B0 35             DEC  (HL)
1078: 04B1 200B           JR   NZ,$+(STAK0-$)
1079: 04B3 E5             PUSH HL
1080: 04B4 DDE5           PUSH IX
1081: 04B6 CD1405        CALL MUZCPU        ; =0 DO NEXT NOTE
1082: 04B9 DDE1           POP  IX
1083: 04BB E1             POP  HL
1084: 04BC 180E           JR   $+(SIXY-$)
1085: 04BE EB             STAK0 EX  DE,HL
1086: 04BF CB7E           BIT  7,(HL)
1087: 04C1 EB             EX  DE,HL
1088: 04C2 2008           JR   NZ,$+(SIXY-$)
1089: 04C4 3D             DEC  A
1090: 04C5 3D             DEC  A           ; =1 QUIET NOTE
1091: 04C6 2004           JR   NZ,$+(SIXY-$)

```

Home Video Game System Library - System Subroutines

```

1092:                ; A=0
1093: 04C8 D316                OUT (VOLAB),A
1094: 04CA D315                OUT (VOLC),A
1095: 04CC 23                SIXY INC HL
1096: 04CD 35                DEC (HL) ; IF(--TMR60<0)
1097: 04CE F20205            JP P,GOUT ; ELZ ONWARD
1098: 04D1 363B            LD (HL),59 ; THEN TMR60=59
1099: 04D3 23                INC HL ; -> TIMOUT
1100: 04D4 EB                EX DE,HL
1101: 04D5 21E34F            LD HL,KEYSEX ; SET SECONDS UP
1102: 04D8 CBFE            SET 7,(HL)
1103: 04DA EB                EX DE,HL
1104: 04DB 7E                LD A,(HL) ; CHECK IF ZERO
1105: 04DC B7                OR A
1106: 04DD 2801            JR Z,$+(GTIMER-$)
1107: 04DF 35                DEC (HL) ; DEC TIMOUT
1108:                ; *GAME TIMER ONCE A SECOND ROUTINE*
1109:                ; IF (SEC != 0 & MIN !=0)
1110:                ; IF (SEC ==0)
1111:                ; SEC=59;--MIN
1112:                ; ELSE --SEC
1113:                ; ELSE GAMETIMEUP=1
1114: 04E0 23                GTIMER INC HL ; ->GTSECS
1115: 04E1 7E                LD A,(HL) ; IF (SEC=0
1116: 04E2 23                INC HL ; ->GTMINS
1117: 04E3 B6                OR (HL) ; & MIN!=0
1118: 04E4 2813            JR Z,$+(GT02-$)
1119: 04E6 2B                DEC HL ; ->GTSECS AGAIN
1120: 04E7 7E                LD A,(HL) ; IF (SEC ==0)
1121: 04E8 B7                OR A
1122: 04E9 2009            JR NZ,$+(GT01-$)
1123: 04EB 3659            LD (HL),$59 ; THEN SEC=59BCD
1124: 04ED 23                INC HL ; ->GTMINS AGAIN
1125: 04EE 7E                LD A,(HL) ; --MIN
1126: 04EF 3D                DEC A
1127: 04F0 27                DAA
1128: 04F1 77                LD (HL),A
1129: 04F2 180E            JR $+(GOUT-$)
1130: 04F4 3D                GT01 DEC A ; ELSE --SEC
1131: 04F5 27                DAA
1132: 04F6 77                LD (HL),A
1133: 04F7 1809            JR $+(GOUT-$)
1134: 04F9 21F84F            GT02 LD HL,GAMSTB ; ELSE GAMETIMEUP=1
1135: 04FC CB46                BIT GSBTIM,(HL)
1136: 04FE 2802            JR Z,$+(GOUT-$)
1137: 0500 CBFE            SET GSBEND,(HL)
1138: 0502 21F94F            GOUT LD HL,PRIOR
1139: 0505 CB8E            RES 1,(HL)
1140: 0507 C9                RET ; RETURN TO BAKGRND OR LO LEVEL
1141:                ;
1142:                ; NAME: START MUZCPU
1143:                ; PURPOSE: TO START MUSIC PLAYING (ALSO NOISES)
1144:                ; INPUTS: HL -> SCORE
1145:                ; A=VOICES
1146:                ; NOTE: YOU SHOULD LOAD MUZSP IF YOU DO CALLS
1147: 0508 32D44F            MUZSET LD (VOICES),A
1148: 050B DD22D04F        LD (MUZSP),IX

```

Home Video Game System Library - System Subroutines

```

1149: 050F CDFC05          CALL MUZSTP
1150: 0512 1803          JR    $+(MUZCP1-$)
1151:                   ; NAME: MUZCPU
1152:                   ; PURPOSE: PLAYING MUSIC AND NOISES
1153:                   ; NOTE: DURAT=0 WHEN CALLED
1154:                   ; OUTPUT: NONE
1155:                   ; *MUSIC PROCESSOR*
1156:                   ; FETCH OPCODE
1157:                   ; IF (OPCODE < 80H)
1158:                   ;     SET NOTE DURATION ETC
1159:                   ; ELSE
1160:                   ; SWITCH (OPCODE < 80H)
1161:                   ; CASE 80H:
1162:                   ;     IF (MASK=8 STUFF SNDBK PC=PC+9
1163:                   ;     ELSE OUTPUT(MASK)=DATA
1164:                   ; CASE 90H:
1165:                   ;     VOICES=DATA
1166:                   ; CASE A0H:
1167:                   ;     (--SP)=DATA IN NIBBLE OF OP + 1
1168:                   ; CASE B0H:
1169:                   ;     SET VOLUMES = DATA,DATA
1170:                   ; CASE C0H:
1171:                   ;     SWITCH (MASK)
1172:                   ;     CASE 9: MPCL=(MSP++); MPCH=(MSP++); BREAK
1173:                   ;     CASE D: (--MSP)=MPCH; (--MSP)=MPCL
1174:                   ;     CASE 0: IF --(SP)==0; THEN SP++
1175:                   ;     CASE 3: MPC=DATA16
1176:                   ; CASE D0H: CALL RELATIVE
1177:                   ; CASE E0: DURAT=DATA
1178:                   ; CASE F0: VOICES=0,PORTS=0
1179: 0514 2ACE4F          MUZCPU LD  HL,(MUZPC)   ; LOOK LIKE NORMAL LOOP RETURN
1180: 0517 DD2AD04F        MUZCP1 LD  IX,(MUZSP)   ; FETCH STACK POINTER
1181: 051B 7E              OPLOOP LD  A,(HL)      ; OPCODE FETCH
1182: 051C 23              INC  HL              ;->OPERAND,DATA
1183: 051D B7              OR   A              ; TEST FOR 80H OR MORE
1184: 051E FA5B05          JP   M,MOO
1185:                   ; NORMAL NOTE OPERATOR
1186: 0521 32EA4F          LD   (DURAT),A
1187: 0524 3AD44F          LD   A,(VOICES)
1188: 0527 011808          LD   BC,800H+SNDBX
1189: 052A CB3F           SRL  A              ; SET NOISE
1190: 052C 3002           JR   NC,($+4)
1191: 052E EDA3           OUTI
1192: 0530 0605           LD   B,5           ; -> VIBRATO
1193: 0532 CB3F           SRL  A
1194: 0534 3002           JR   NC,($+4)
1195: 0536 EDA3           OUTI              ; SET VIBRATO
1196: 0538 0604           LD   B,4           ; -> NOTEC
1197: 053A CB3F           M81: SRL  A         ; CHECK C,B,A
1198: 053C 3009           JR   NC,$+(M82-$)
1199: 053E EDA3           OUTI
1200: 0540 CB3F           M815 SRL  A         ; CHECK IF INC PC WAS ON
1201: 0542 3807           JR   C,$+(M83-$)
1202: 0544 2B             DEC  HL            ; RESTORE PC
1203: 0545 1804           JR   $+(M83-$)
1204: 0547 05             M82  DEC  B
1205: 0548 23             INC  HL

```

Home Video Game System Library - System Subroutines

```

1206: 0549 18F5          JR  $(M815-$)
1207: 054B B7             M83  OR  A
1208: 054C 20EC          JR  NZ, $(M81-$)
1209:                ; PLAY NOTE
1210: 054E 3AD24F        LD  A, (PVOLAB)
1211: 0551 D316          OUT (VOLAB), A
1212: 0553 3AD34F        LD  A, (PVOLMC)
1213: 0556 D315          OUT (VOLC), A
1214: 0558 C3F405        JP  MUZ999
1215: 055B FE90          MO0:  CP  90H
1216: 055D 3015          JR  NC, $(MO1-$)
1217:                ; STUFF PORT OR SOUND BLOCK
1218: 055F CB5F          BIT  3, A           ; IF (STUFF SNDBLK)
1219: 0561 2808          JR  Z, $(MO01-$)
1220: 0563 78            LD  A, B           ; SAVE B (VSN)
1221: 0564 011808        LD  BC, 8*256+SNDBX ; B=8 C=SNDBX
1222: 0567 EDB3          OTIR                ; HL->NEXT OPCODE WHEN DONE
1223: 0569 18B0          JR  $(OPLOOP-$)
1224: 056B E607          MO01 AND  7           ; ISOLATE PORT NUMBER
1225: 056D F610          OR  10H           ; PORTS 10H-17H
1226: 056F 4F            LD  C, A           ; SET PORT REGISTER
1227: 0570 EDA3          OUTI
1228: 0572 18A7          JR  $(OPLOOP-$)
1229: 0574 2007          MO1:  JR  NZ, $(MO2-$)
1230: 0576 7E            LD  A, (HL)       ; GET NEW VOICES
1231: 0577 23            INC  HL
1232: 0578 32D44F        LD  (VOICES), A
1233: 057B 189E          JR  $(OPLOOP-$)
1234: 057D FEB0          MO2:  CP  0B0H
1235: 057F 3006          JR  NC, $(MO3-$)
1236: 0581 E60F          AND  0FH
1237: 0583 5F            LD  E, A
1238: 0584 1C            INC  E
1239: 0585 183E          JR  $(MO45-$)
1240: 0587 FEC0          MO3:  CP  0C0H           ; SET VOL ETC
1241: 0589 3009          JR  NC, $(MO4-$)
1242:                ; LOAD PVOLS
1243: 058B 11D24F        LD  DE, PVOLAB
1244: 058E EDA0          LDI                ; DONT CARE ABOUT BC
1245: 0590 EDA0          LDI
1246: 0592 1887          OPLP2 JR  $(OPLOOP-$)
1247: 0594 200B          MO4  JR  NZ, $(MO40-$)
1248: 0596 DD3500        DEC  (IX+0)       ; DEC STACK TOP
1249: 0599 200A          JR  NZ, $(MO41-$)
1250: 059B DD23          INC  IX
1251: 059D 23            INC  HL
1252: 059E 23            INC  HL
1253: 059F 18F1          JR  $(OPLP2-$)
1254: 05A1 FED0          MO40 CP  0D0H           ; PC SP STUFF
1255: 05A3 3027          JR  NC, $(MO5-$)
1256: 05A5 E60F          MO41 AND  0FH           ; ISOLATE MASK
1257: 05A7 FE09          CP  9             ; RETURN
1258: 05A9 200C          JR  NZ, $(MO43-$)
1259: 05AB DD6E00        LD  L, (IX+0)
1260: 05AE DD23          INC  IX
1261: 05B0 DD6600        LD  H, (IX+0)
1262: 05B3 DD23          INC  IX

```

Home Video Game System Library - System Subroutines

```

1263: 05B5 18DB          JR  $+(OPLP2-$)
1264: 05B7 5E          MO43: LD  E,(HL)      ; PCL=
1265: 05B8 23          INC  HL
1266: 05B9 56          LD  D,(HL)      ; PCH=
1267: 05BA 23          INC  HL
1268: 05BB EB          EX  DE,HL      ; SET THE PC
1269: 05BC FE04        CP  4          ; IS IT A JMP?
1270: 05BE 38D2        JR  C,$+(OPLP2-$) ; IT IS
1271: 05C0 DD2B          MO44: DEC  IX      ; IT'S A CALL
1272: 05C2 DD7200       LD  (IX+0),D    ; (--SP)=PHL
1273: 05C5 DD2B          MO45: DEC  IX
1274: 05C7 DD7300       LD  (IX+0),E    ; (--SP)=PCL
1275: 05CA 18C6        JR  $+(OPLP2-$)
1276: 05CC FEE0          MO5  : CP  0E0H
1277: 05CE 300A        JR  NC,$+(M06-$)
1278: 05D0 E60F        AND  0FH
1279: 05D2 0600       LD  B,0
1280: 05D4 4F          LD  C,A
1281: 05D5 54          LD  D,H
1282: 05D6 5D          LD  E,L
1283: 05D7 09          ADD  HL,BC
1284: 05D8 18E6        JR  $+(M044-$)  ; CALL
1285: 05DA 200A        MO6  : JR  NZ,$+(M061-$)
1286: 05DC 3AF94F       LD  A,(PRIOR)  ; LEGSTA
1287: 05DF EE80          XOR  80H
1288: 05E1 32F94F       LD  (PRIOR),A
1289: 05E4 18AC        JR  $+(OPLP2-$)
1290: 05E6 FEF0          MO61: CP  0F0H    ; REST VOICE (OR SUSTAIN)
1291: 05E8 2812        JR  Z,$+(MUZSTP-$)
1292: 05EA 7E          LD  A,(HL)
1293: 05EB 32EA4F       LD  (DURAT),A  ; SET DURATION OR QUIET
1294: 05EE 23          INC  HL
1295: 05EF AF          XOR  A
1296: 05F0 D316        OUT  (VOLAB),A
1297: 05F2 D315        OUT  (VOLC),A
1298:                ; END OF MUSIC PROCESSOR
1299: 05F4 22CE4F       MUZ999 LD  (MUZPC),HL ; SAVE THE PC
1300: 05F7 DD22D04F      LD  (MUZSP),IX ; SAVE THE STACK POINTER
1301: 05FB C9          RET
1302:                ; NAME MUZSTP
1303:                ; PURPOSE: STOP MUZCPU, SET PORT TO 0
1304: 05FC AF          MUZSTP: XOR  A
1305: 05FD 32EA4F       LD  (DURAT),A
1306: 0600 32F94F       LD  (PRIOR),A
1307: 0603 011808      LD  BC,800H+SNDBX
1308: 0606 ED79        OUT  (C),A
1309: 0608 10FC        DJNZ ($-2)
1310: 060A C9          RET
1311:                ;
1312:                ; NAME: DO IT
1313:                ; PURPOSE: TRANSFER CONTROL TO USER STATE TRANSITION
1314:                ; INPUT: A = RETURN CODE FROM SENTRY ROUTINE
1315:                ; HL = DO IT TABLE ADDRESS
1316:                ; OUTPUT:
1317:                ; DESCRIPTION: THIS ROUTINE IS USED WITH THE SENTRY ROUT
1318:                ; IT IS USED FOR DISPATCHING TO A STATE TRANSITION
1319:                ; ROUTINE. THE RETURN CODE FROM SENTRY IS USED TO

```

Home Video Game System Library - System Subroutines

```

1320:          ;      SEARCH THE DOIT TABLE.  IF A MATCH IS FOUND, CONT
1321:          ;      TRANSFERRED.  IF NOT MATCH IS FOUND, THE ROUTINE RE
1322:          ;      THE DOIT TABLE IS MADE UP OF THREE BYTE ENTRYS:
1323:          ;      BYTE 0 BIT 7: IF SET - DO A MCALL TO THIS HANDLER
1324:          ;      BYTE 0 BIT 6: IF SET - DO A RCALL TO THIS HANDLER
1325:          ;      BYTE 0 BITS 5-0: RETURN CODE THIS ROUTINE IS TO PR
1326:          ;      BYTE 1 AND 2: THE ADDRESS TO TRANSFER TO.
1327:          ;      THE LIST IS TERMINATED BY A BYTE WHICH IS .GE.  OC
1328: 060B 78      MDOITB: LD   A,B
1329: 060C D5      MDOIT:  PUSH DE
1330: 060D 57              LD   D,A
1331: 060E 7E      MDOIT0: LD   A,(HL)      ; GET RETURN CODE FOR THIS ENTR
1332: 060F 4F              LD   C,A          ; C = CURRENT ENTRY
1333: 0610 FEC0      CP    0C0H          ; LIST TERMINATOR?
1334: 0612 3802      JR    C, $(MDOIT1-$) ; NO - JUMP
1335: 0614 D1              POP  DE          ; YES  RETURN
1336: 0615 C9              RET
1337: 0616 23      MDOIT1: INC  HL
1338: 0617 E63F      AND  3FH
1339: 0619 BA              CP    D          ; NORMAL MATCH?
1340: 061A 2804      JR    Z, $(MDOIT2-$) ; JUMP IF SO
1341: 061C 23      MDO1A:  INC  HL          ; NO MATCH - SKIP OVER
1342: 061D 23              INC  HL          ; GO TO ADDRESS
1343: 061E 18EE      JR    $(MDOIT0-$)
1344: 0620 D1      MDOIT2: POP  DE
1345: 0621 5E      MDOIT3: LD   E,(HL)      ; DE = GOTO ADDR
1346: 0622 23              INC  HL
1347: 0623 56              LD   D,(HL)
1348: 0624 EB              EX  DE,HL
1349: 0625 CB79      BIT  7,C          ; MCALL?
1350: 0627 C27D00     JP   NZ,MMCALL    ; JUMP IF SO
1351: 062A CB71      BIT  6,C          ; RCALL?
1352: 062C 2004      JR    NZ, $(MRCALL-$)
1353: 062E D1              POP  DE          ; MUST BE JUMP
1354: 062F F1              POP  AF
1355: 0630 E5              PUSH HL
1356: 0631 EB              EX  DE,HL
1357:          ; RCALL ROUTINE
1358: 0632 E9      MRCALL: JP   (HL)
1359:          ; *****
1360:          ; * VECTORING ROUTINES *
1361:          ; *****
1362:          ; NAME:          VECTOR X AND Y COORDINATES
1363:          ; PURPOSE:      UPDATE X,Y COORDINATES AND LIMIT CHECK
1364:          ; INPUT:        IX = VECTOR PACKET
1365:          ;              HL = LIMITS TABLE
1366:          ; OUTPUT:      C = TIME BASE USED
1367:          ;              NONZERO STATUS SET IF OBJECT MOVED
1368:          ; NOTES:
1369:          ;   THIS ROUTINE WORKS WITH A 'VECTOR PACKET', WHICH LOO
1370:          ; *****
1371:          ; *BYTE*      CONTENTS      * NAME      *
1372:          ; *****
1373:          ; * 00 * MAGIC REGISTER * VBMR      *
1374:          ; *****
1375:          ; * 01 * VECTOR STATUS * VBSTAT    *
1376:          ; *****

```

Home Video Game System Library - System Subroutines

```

1377:          ; * 02 * TIME BASE          * VBTIMB *
1378:          ; *****
1379:          ; * 03 * DELTA X            * VBDXL  *
1380:          ; * 04 *                      * VBDXH  *
1381:          ; *****
1382:          ; * 05 * X COORDINATE        * VBXL   *
1383:          ; * 06 *                      * VBXH   *
1384:          ; *****
1385:          ; * 07 * X CHECKS MASK        * VBXCHK *
1386:          ; *****
1387:          ; * 08 * DELTA Y            * VBDYL  *
1388:          ; * 09 *                      * VBDYH  *
1389:          ; *****
1390:          ; * 0A * Y COORDINATE        * VBYL   *
1391:          ; * 0B *                      * VBYH   *
1392:          ; *****
1393:          ; * 0C * Y CHECKS MASK        * VBYCHK *
1394:          ; *****
1395:          ;
1396:          ; OPTIONS BYTE:
1397:          ; BIT  MEANING
1398:          ; ---  -----
1399:          ;   7   VECTOR IS ACTIVE
1400:          ;
1401:          ; CHECKS BYTE:
1402:          ; BIT  MEANING
1403:          ; ---  -----
1404:          ;   0   DO LIMIT CHECKS
1405:          ;   1   REVERSE COORDINATES ON LIMIT ATTAINMENT
1406:          ;   3   TARGET ATTAINED (OUTPUT)
1407:          ;   IF THE VECTOR IS ACTIVE, AND THE TIME BASE IS NONZER
1408:          ; THEN UPDATE COORDINATE ROUTINE IS CALLED FOR THE X
1409:          ; AND Y PORTIONS OF THE PACKET.
1410: 0633  FDCB08F6      MVECT  SET  PSWZRO,(IY+CBFLAG) ; SET ZERO FLAG
1411: 0637  DDCB017E          BIT  VBSACT,(IX+VBSTAT) ; IS VECTOR ACTIVE?
1412: 063B  DD4E02          LD   C,(IX+VBTIMB)      ; TIME BASE TO C
1413: 063E  DD360200        LD   (IX+VBTIMB),0      ; ZERO TIME BASE
1414: 0642  FD7106          LD   (IY+CBC),C        ; PASS BACK TIME BASE
1415: 0645  C8              RET  Z
1416: 0646  79              LD   A,C
1417: 0647  A7              AND  A                ; IS TIME BASE ZERO?
1418: 0648  C8              RET  Z                ; QUIT IF SO
1419: 0649  110300          LD   DE,VBDXL        ; ADVANCE TO FIRST
1420: 064C  DD19            ADD  IX,DE
1421: 064E  CD5606          CALL MVECTC          ; UPDATE FIRST COORDINATE
1422: 0651  110500          LD   DE,VBDYL-VBDXL ; TO Y
1423: 0654  DD19            ADD  IX,DE
1424:          ; AND FALL INTO...
1425:          ; NAME:          VECTOR COORDINATE
1426:          ; PURPOSE:       UPDATE OF SINGLE COORDINATE
1427:          ; INPUT:         IX = POINTER TO L.O. DELTA BYTE OF VECTOR
1428:          ;                 C = TIME BASE
1429:          ;                 HL = LIMITS PACKET (IF USED)
1430:          ; OUTPUT:        NONZERO STATUS SET IF MOTION IS OCCURED
1431:          ;                 (SHOULD BE SET ON CALL, SINCE IT IS NOT S
1432:          ; NOTES:
1433:          ;   THIS ROUTINE OPERATES ON A SUBSET OF THE VECTOR PACK

```

Home Video Game System Library - System Subroutines

```

1434:                ; (BETWEEN L.O. DELTA BYTE AND CHECKS BYTE).
1435:                ; THE DELTA IS ADDED TO THE COORDINATE TIME-BASE TIMES
1436:                ; IF OPTIONED, LIMIT CHECKING IS DONE. IF HE CHECK FAI
1437:                ; THE COORDINATE IS SET TO THE LIMIT.
1438:                ; WHEN THIS HAPPENS, THE LIMIT ATTAINED BIT IS SET
1439: 0656 E5          MVECTC  PUSH HL
1440: 0657 DD5601      LD      D,(IX+VBDCH) ; LOAD DELTA
1441: 065A DD5E00      LD      E,(IX+VBDCL)
1442: 065D DD6603      LD      H,(IX+VBCH) ; LOAD COORDINATE
1443: 0660 DD6E02      LD      L,(IX+VBCL)
1444: 0663 7C          LD      A,H          ; SAVE OLD COORDINATE FOR MOTIO
1445: 0664 41          LD      B,C
1446: 0665 19          MVECT1  ADD  HL,DE          ; ADD DELTA TO COORDINATE
1447: 0666 10FD        DJNZ  $+(MVECT1-$) ; TIME-BASE TIMES
1448:                ; HAS MOTION OCCURRED?
1449: 0668 BC          CP      H
1450: 0669 2804        JR      Z,$+(MVCT1A-$) ; JUMP TO SKIP TEST IF SO
1451: 066B FDCB08B6    RES    PSWZRO,(IY+CBFLAG) ; SET MOVE STATUS
1452:                ; IS LIMIT CHECK WANTED?
1453: 066F DDCB0446    MVCT1A BIT  VBCLMT,(IX+VBCCHK)
1454: 0673 2831        JR      Z,$+(MVECT6-$) ; MVECT6 IF NOT
1455:                ; PERFORM LIMIT CHECK
1456: 0675 7C          LD      A,H
1457: 0676 E3          EX      (SP),HL
1458: 0677 46          LD      B,(HL)          ; LIMIT TO B
1459: 0678 23          INC    HL
1460:                ; HANDLE SLIGHTLY LESS THAN ZERO CASE
1461: 0679 FECF        CP      207          ; MIDPOINT BETWEEN 160 AND 0
1462: 067B 3007        JR      NC,$+(MVECT2-$) ; JUMP TO FAIL IF >207
1463: 067D B8          CP      B          ; DO COMPARE
1464: 067E 3804        JR      C,$+(MVECT2-$) ; JUMP ON FAIL
1465: 0680 46          LD      B,(HL)          ; UPPER LIMIT CHECK
1466: 0681 B8          CP      B
1467: 0682 3820        JR      C,$+(MVECT3-$) ; JUMP ON PASS
1468: 0684 23          MVECT2  INC  HL
1469:                ; A LIMIT WAS EXCEEDED - SET COORDINATE AT LIMIT
1470: 0685 DD7003      LD      (IX+VBCH),B
1471: 0688 DD360200     LD      (IX+VBCL),0
1472: 068C DDCB04DE     SET    VBCLAT,(IX+VBCCHK) ; SET LIMIT ATTAINED
1473:                ; IS REVERSE DELTA SET
1474: 0690 F1          POP    AF          ; CLEAN UP STACK
1475: 0691 DDCB044E     BIT    VBCREV,(IX+VBCCHK)
1476: 0695 C8          RET    Z          ; QUIT IF NOT
1477:                ; REVERSE THE BIMBO
1478: 0696 7A          LD      A,D
1479: 0697 2F          CPL
1480: 0698 57          LD      D,A
1481: 0699 7B          LD      A,E
1482: 069A 2F          CPL
1483: 069B 5F          LD      E,A
1484: 069C 13          INC    DE
1485: 069D DD7300      LD      (IX+VBDCL),E ; STORE STACK
1486: 06A0 DD7201      LD      (IX+VBDCH),D
1487: 06A3 C9          RET
1488: 06A4 23          MVECT3  INC  HL          ; STEP PAST LIMIT
1489: 06A5 E3          EX      (SP),HL      ; HL = COORDINATE AGAIN
1490: 06A6 DD7502      MVECT6  LD      (IX+VBCL),L ; STORE BACK COORDINATES

```

Home Video Game System Library - System Subroutines

```

1491: 06A9 DD7403          LD    (IX+VBCH),H
1492: 06AC E1              POP   HL                ; RESTORE LIMITS POINTER
1493: 06AD DDCB049E        RES   VBCLAT,(IX+VBCCHK) ; CLEAR ATTAINED BIT
1494: 06B1 C9              RET
1495:                ;
1496:                ; *****
1497:                ; * PAINT RECTANGLE ROUTINE *
1498:                ; *****
1499:                ; NAME:          PAINT RECTANGLE
1500:                ; INPUT:         A = COLOR MASK TO WRITE
1501:                ;                B = Y SIZE
1502:                ;                C = X SIZE
1503:                ;                D = Y COORDINATE
1504:                ;                E = X COORDINATE
1505: 06B2 AF              MPAINT XOR   A
1506: 06B3 CD4E0B         CALL  RELTAL
1507: 06B6 EB              EX    DE,HL
1508: 06B7 CBF4           SET   6,H                ; UNMAGIC THE ADDRESS
1509: 06B9 D30C           OUT   (MAGIC),A
1510:                ;          XOR   A
1511: 06BB FD5E09         LD    E,(IY+CBA)
1512: 06BE 79              LD    A,C
1513: 06BF 0F              RRCA
1514: 06C0 0F              RRCA
1515: 06C1 E63F           AND   3FH
1516: 06C3 3C              INC   A
1517: 06C4 57              LD    D,A
1518: 06C5 15              MPT1  DEC   D
1519: 06C6 2807           JR    Z,$+(MPT2-$)
1520: 06C8 3EFF           LD    A,0FFH
1521: 06CA CDE206         CALL  STRIPE
1522: 06CD 18F6           JR    $+(MPT1-$)
1523: 06CF 79              MPT2  LD    A,C
1524: 06D0 E603           AND   03H
1525: 06D2 3C              INC   A
1526: 06D3 4F              LD    C,A
1527: 06D4 AF              XOR   A
1528: 06D5 0D              MPT3  DEC   C
1529: 06D6 2806           JR    Z,$+(MPT4-$)
1530: 06D8 0F              RRCA
1531: 06D9 0F              RRCA
1532: 06DA C6C0           ADD   A,11000000B
1533: 06DC 18F7           JR    $+(MPT3-$)
1534: 06DE CDE206         MPT4  CALL  STRIPE
1535: 06E1 AF              XOR   A
1536:                ; AND FALL INTO...
1537:                ; STRIPE PAINTER
1538:                ; HL = ADDRESS OF STRIPE A = DATA E = MASK B = ITERATIONS
1539:                ; OUT HL=HL+1 A = CLOBBERED
1540: 06E2 E5              STRIPE PUSH HL
1541: 06E3 C5              PUSH  BC
1542: 06E4 32FF0F         LD    (WASTE),A
1543: 06E7 3AFF4F         LD    A,(WASTE+4000H)
1544: 06EA 4F              LD    C,A
1545: 06EB 7B              STRP1 LD    A,E
1546: 06EC AE              XOR   (HL)
1547: 06ED A1              AND   C

```

Home Video Game System Library - System Subroutines

```

1548: 06EE AE          XOR   (HL)
1549: 06EF 77          LD    (HL),A
1550: 06F0 7D          LD    A,L
1551: 06F1 C628         ADD   A,BYTEPL
1552: 06F3 6F          LD    L,A
1553: 06F4 7C          LD    A,H
1554: 06F5 CE00         ADC   A,0
1555: 06F7 67          LD    H,A
1556: 06F8 10F1         DJNZ  $(STRP1-$)
1557: 06FA C1          POP   BC
1558: 06FB E1          POP   HL
1559: 06FC 23          INC   HL
1560: 06FD C9          RET

1561:                ;
1562:                ; *****
1563:                ; * WRITE ROUTINES *
1564:                ; *****
1565:                ; NOTES:          THE GENERAL CALLING SEQUENCE FOR THE WRI
1566:                ; INPUT:          HL = PATTERN ADDRESS
1567:                ;                  D = Y COORDINATE
1568:                ;                  E = X COORDINATE
1569:                ;                  B = Y SIZE
1570:                ;                  C = X SIZE
1571:                ;                  A = MAGIC REGISTER
1572:                ; OUTPUT:         DE = SCREEN ADDRESS USED
1573:                ;                  THESE ROUTINES ARE NESTED, FOR EXAMPLE
1574:                ;                  WRITP, WHICH FALLS INTO WRIT, WHICH FALL
1575:                ; ENTRY:          WRITE FROM VECTOR
1576:                ; INPUT:          HL = PATTERN ADDRESS
1577:                ;                  IX = VECTOR ADDRESS
1578:                ; OUTPUT:         DE,A
1579:                ; SIDE EFFECTS: BLANK BIT SET IN VECTOR STATUS BYTE
1580: 06FE DD7E00        MVWRIT LD    A,(IX+VBMR) ; LOAD MR
1581: 0701 DD560B        LD    D,(IX+VBYH) ; LOAD Y
1582: 0704 DD5E06        LD    E,(IX+VBXH) ; LOAD X
1583: 0707 DDCB01F6      SET   VBBLNK,(IX+VBSTAT) ; SET BLANK BIT
1584:                ; ENTRY:          WRITE RELATIVE
1585:                ; PURPOSE:        WRITING RELATIVE PATTERNS
1586:                ; INPUT:          HL,DE,A
1587:                ; OUTPUT:         DE
1588:                ; NOTES:          PATTERN IS PRECEDED BY RELATIVE DISPLAC
1589:                ;                  (X FIRST, THEN Y, AND PATTERN SIZE
1590: 070B F5          MWRITR PUSH  AF          ; SAVE MR
1591: 070C 7E          LD    A,(HL)      ; GET REL X
1592: 070D 23          INC   HL
1593: 070E 83          ADD   A,E          ; ADD TO SUPERIOR X
1594: 070F 5F          LD    E,A
1595: 0710 7E          LD    A,(HL)      ; SAME STORY FOR Y
1596: 0711 23          INC   HL
1597: 0712 82          ADD   A,D
1598: 0713 57          LD    D,A
1599: 0714 F1          POP   AF
1600:                ; ENTRY:          WRITE WITH PATTERN SIZE SCARE-UP
1601:                ; PURPOSE:        WRITING VARIABLE SIZED PATTERNS
1602:                ; INPUT:          HL,DE,A
1603:                ; OUTPUT:         DE
1604:                ; NOTES:          FIRST TWO BYTES POINTED AT BY HL ARE TAK

```

Home Video Game System Library - System Subroutines

```

1605:                ;                TO BE PATTERN SIZES (X SIZE FIRST)
1606: 0715 4E          MWRITP LD  C,(HL)          ; GET X SIZE
1607: 0716 23                INC  HL
1608: 0717 46                LD   B,(HL)          ; AND Y
1609: 0718 23                INC  HL
1610:                ; ENTRY:          WRITE WITH COORDINATE CONVERSION
1611:                ; INPUT:          HL,DE,BC,A
1612:                ; OUTPUT:         DE
1613: 0719 CDF60A        MWRIT  CALL MRELAB          ; DO CONVERSION
1614:                ; ENTRY:          WRITE ABSOLUTE
1615:                ; INPUT:          HL,BC,A AS ABOVE
1616:                ;                DE = ABSOLUTE SCREEN ADDRESS
1617: 071C CB77          MWRITA BIT  MRFLOP,A          ; FLOP WRITE WANTED?
1618: 071E 202C          JR   NZ,$+(MWRTFL-$) ; MWRTFL IS SO
1619: 0720 CB5F          BIT  MRXPND,A          ; EXPAND WANTED?
1620: 0722 2011          JR   NZ,$+(MWX-$)   ; JUMP IF SO
1621:                ; DO NORMAL? WRITE
1622: 0724 AF                XOR  A
1623: 0725 C5                MWRIT  PUSH BC
1624: 0726 D5                PUSH DE
1625: 0727 47                LD   B,A            ; ZERO REGISTER B
1626: 0728 EDB0          LDIR                ; WRITE A LINE
1627: 072A 12                LD   (DE),A        ; CLEAR THE SHIFTER
1628: 072B D1                POP  DE
1629: 072C EB                EX   DE,HL          ; ADVANCE TO NEXT LINE
1630: 072D 0E28          LD   C,BYTEPL
1631: 072F 09                ADD  HL,BC
1632: 0730 EB                EX   DE,HL
1633: 0731 C1                POP  BC
1634: 0732 10F1          DJNZ $+(MWRT-$)   ; LOOP IF MORE GOODIES
1635: 0734 C9                RET
1636:                ; WRITE EXPANDED
1637: 0735 EB                MWX   EX   DE,HL
1638: 0736 C5                MWX1  PUSH BC
1639: 0737 E5                PUSH HL
1640: 0738 41                LD   B,C
1641: 0739 1A                MWX2  LD   A,(DE)
1642: 073A 13                INC  DE
1643: 073B 77                LD   (HL),A
1644: 073C 23                INC  HL
1645: 073D 77                LD   (HL),A
1646: 073E 23                INC  HL
1647: 073F 10F8          DJNZ $+(MWX2-$)
1648: 0741 70                LD   (HL),B
1649: 0742 23                INC  HL
1650: 0743 70                LD   (HL),B
1651: 0744 E1                POP  HL
1652: 0745 0E28          LD   C,BYTEPL
1653: 0747 09                ADD  HL,BC
1654: 0748 C1                POP  BC
1655: 0749 10EB          DJNZ $+(MWX1-$)
1656: 074B C9                RET
1657:                ; ROUTINE TO HANDLE FLOPPED CASE
1658: 074C CB5F          MWRTFL BIT  MRXPND,A          ; EXPANDED FLOPPED WRITE WANTED
1659: 074E 2016          JR   NZ,$+(MWXF-$) ; JUMP IF YEP
1660: 0750 AF                XOR  A
1661: 0751 C5                WRFL1  PUSH BC

```

Home Video Game System Library - System Subroutines

```

1662: 0752 D5          PUSH DE
1663: 0753 47          LD B,A
1664: 0754 EDA0          WRFL2 LDI
1665: 0756 1B          DEC DE
1666: 0757 1B          DEC DE
1667: 0758 EA5407        JP PE,WRFL2
1668: 075B 12          LD (DE),A
1669: 075C D1          POP DE
1670: 075D EB          EX DE,HL ; SAME AS NORMAL NOW ON
1671: 075E 0E28        LD C,BYTEPL
1672: 0760 09          ADD HL,BC
1673: 0761 EB          EX DE,HL
1674: 0762 C1          POP BC
1675: 0763 10EC        DJNZ $+(WRFL1-$)
1676: 0765 C9          RET
1677:                ; WRITE EXPANDED FLOPPED ROUTINE
1678: 0766 EB          MWXF EX DE,HL
1679: 0767 C5          MWXF1 PUSH BC
1680: 0768 E5          PUSH HL
1681: 0769 41          LD B,C
1682: 076A 1A          MWXF2 LD A,(DE)
1683: 076B 13          INC DE
1684: 076C 77          LD (HL),A
1685: 076D 2B          DEC HL
1686: 076E 77          LD (HL),A
1687: 076F 2B          DEC HL
1688: 0770 10F8        DJNZ $+(MWXF2-$)
1689: 0772 70          LD (HL),B
1690: 0773 2B          DEC HL
1691: 0774 70          LD (HL),B
1692: 0775 E1          POP HL
1693: 0776 0E28        LD C,BYTEPL
1694: 0778 09          ADD HL,BC
1695: 0779 C1          POP BC
1696: 077A 10EB        DJNZ $+(MWXF1-$)
1697: 077C C9          RET
1698:                ; NAME: BLANK FROM VECTOR
1699:                ; PURPOSE: BLANK WITH INFO LOAD FROM VECTOR
1700:                ; INPUT: IX = VECTOR
1701:                ; E = X SIZE
1702:                ; D = Y SIZE
1703:                ; NOTES: THIS ROUTINE BLANKS TO 00
1704:                ; THIS ROUTINE INTERROGATES THE BLANK BIT
1705:                ; AND REFRAINS FROM BLANKING IF NOT SET
1706:                ; IF IT WAS SET, IT IS THEN RESET
1707: 077D DDCB0176        MVBLAN BIT VBBLNK,(IX+VBSTAT) ; IS BLANK BIT SET?
1708: 0781 C8          RET Z ; QUIT IF NOT
1709: 0782 DDCB01B6        RES VBBLNK,(IX+VBSTAT) ; KILL BLANK BIT
1710: 0786 DD660E        LD H,(IX+VBOAH) ; LOAD BLANK ADDRESS
1711: 0789 DD6E0D        LD L,(IX+VBOAL)
1712: 078C DDCB0076        BIT MRFLOP,(IX+VBMR) ; IS FLOP SET?
1713: 0790 2808        JR Z,$+(MVBLA1-$) ; JUMP IF NOT
1714: 0792 7B          LD A,E ; X SIZE TO A
1715: 0793 ED44        NEG ; TWOS COMPLEMENT AND ADD 1
1716: 0795 3C          INC A
1717: 0796 4F          LD C,A
1718: 0797 06FF        LD B,0FFH

```

Home Video Game System Library - System Subroutines

```

1719: 0799 09          ADD HL,BC          ; USE TO BACK UP SCREEN ADDRESS
1720:                ; UNMAGIC THE BLANK ADDRESS
1721: 079A          MVBLA1
1722: 079A CBF4          SET 6,H
1723: 079C 0600          LD B,0          ; ASSUME BLANK TO ZERO
1724:                ; NAME:          BLANK AREA
1725:                ; PURPOSE:       SETTING N X M REGION TO CONSTANT
1726:                ; INPUT:         HL = BLANK ADDRESS
1727:                ;                E = X SIZE
1728:                ;                D = Y SIZE
1729:                ;                B = DATA TO FILL WITH
1730: 079E 3E28          MBLANK LD A,BYTEPL    ; COMPUTE LINE INCREMENT
1731: 07A0 93           SUB E
1732: 07A1 4F           LD C,A
1733: 07A2 78           LD A,B          ; A = DATA TO FILL WITH
1734: 07A3 43           MBLAN1 LD B,E
1735: 07A4 77           MBLAN2 LD (HL),A
1736: 07A5 23           INC HL
1737: 07A6 10FC          DJNZ $+(MBLAN2-$)
1738: 07A8 09           ADD HL,BC
1739: 07A9 15           DEC D
1740: 07AA 20F7          JR NZ,$+(MBLAN1-$)
1741: 07AC C9           RET
1742:                ; NAME:          RESTORE AREA
1743:                ; INPUT:         HL = SCREEN ADDRESS TO RESTORE TO
1744:                ;                DE = SAVE AREA ADDRESS
1745:                ; NOTE:         SIZES ARE LOADED FROM THE SAVE AREA
1746: 07AD EB           MREST EX DE,HL
1747: 07AE 4E           LD C,(HL)
1748: 07AF 23           INC HL
1749: 07B0 46           LD B,(HL)
1750: 07B1 23           INC HL
1751: 07B2 CBF2          SET 6,D          ; MAKE SURE WE ARE NONMAGIC
1752: 07B4 AF           XOR A
1753: 07B5 C5           MREST1 PUSH BC
1754: 07B6 D5           PUSH DE
1755: 07B7 47           LD B,A
1756: 07B8 EDB0          LDIR
1757: 07BA EB           EX DE,HL
1758: 07BB E1           POP HL
1759: 07BC 0E28          LD C,BYTEPL
1760: 07BE 09           ADD HL,BC
1761: 07BF EB           EX DE,HL
1762: 07C0 C1           POP BC
1763: 07C1 10F2          DJNZ $+(MREST1-$)
1764: 07C3 C9           RET
1765:                ;
1766:                ; *****
1767:                ; * CHARACTER DISPLAY ROUTINES *
1768:                ; *****
1769:                ; NAME:          DISPLAY STRING
1770:                ; PURPOSE:       MESSAGE DISPLAY
1771:                ; INPUT:         E,D = X, Y COORDINATES
1772:                ;                HL = STRING ADDRESS
1773:                ;                IX = FONT DESCRIPTOR
1774:                ; OUTPUT:        D,E ALTERED AS IN DISPLAY CHARACTER
1775:                ; STACK USE:     4 BYTES (EXCLUDING USE BY SYSPCH)

```

Home Video Game System Library - System Subroutines

```

1776:          ; EXPLANATION: AS EACH CHARACTER IS BROUGHT IN, IT
1777:          ; IS TESTED FOR BEING A LIST TERMINATOR ( CHAR = 0)
1778:          ; IF IT ISN'T, DISPLAY CHARACTER IS CALLED AND THE
1779:          ; TEST IS REPEATED FOR THE NEXT CHARACTER.  THUS
1780:          ; A NULL STRING IS HANDLED PROPERLY.
1781: 07C4 7E          STRNEW LD  A,(HL)          ; GET CHARACTER
1782: 07C5 A7          AND  A                    ; BE IT A TERMINATOR
1783: 07C6 C8          RET  Z                    ; QUIT IF SO
1784: 07C7 FACE07      JP  M,STRD1        ; DISPLAY IF ALT FONT
1785: 07CA FE64        CP  64H                    ; SUCK IN STRING?
1786: 07CC 3006        JR  NC,$+(STRD2-$) ; JUMP IF YES
1787: 07CE CDE107      STRD1 CALL DISPCH        ; SHOW CHAR
1788: 07D1 23          INC  HL                    ; ADVANCE TO NEXT CHAR
1789: 07D2 18F0        JR  $+(STRNEW-$) ; AND LOOP
1790: 07D4 E617        STRD2 AND  10111B        ; MAKE SUCK MASK
1791: 07D6 47          LD  B,A
1792: 07D7 23          INC  HL
1793: 07D8 EB          EX  DE,HL
1794: 07D9 CDA800      CALL MSUCK1
1795: 07DC CD6800      CALL RELD
1796: 07DF 18E3        JR  $+(STRNEW-$) ; GO AFTER NEXT CHARACTER
1797:          ; *****
1798:          ; * CHARACTER DISPLAY ROUTINE *
1799:          ; *****
1800:          ; INPUT:          A = CHARACTER
1801:          ;                  C = OPTIONS
1802:          ;                  D = Y COORDINATE
1803:          ;                  E = X COORDINATE
1804:          ;                  IX = FONT DESCRIPTOR
1805:          ;                  (ONLY IF ALTERNATE FONT SET USED)
1806:          ; OUTPUT:        DE UPDATED TO POINT AT NEXT CHARACTER FRA
1807:          ; NOTES:        THE OPTION BYTE IS FORMATTED AS FOLLOWS:
1808:          ;          BITS  CONTENTS
1809:          ;          ----  -
1810:          ;          0-1   OFF COLOR FOR EXPANSION
1811:          ;          2-3   ON COLOR FOR EXPANSION
1812:          ;          4     OR OPTION
1813:          ;          5     XOR OPTION
1814:          ;          6-7   ENLARGMENT FACTOR (N+1)X
1815:          ;
1816:          ; CHARACTERS BETWEEN 1 AND 1FH, AND BETWEEN 81H AND 9FH
1817:          ; ARE INTERPRETED AS TAB CHARACTERS.  THEY CAUSE THE
1818:          ; CURSOR REPRESENTED BY D AND E TO BE SPACED OVER N
1819:          ; CHARACTER POSITIONS, WHERE N = CHAR.AND.7FH
1820:          ; CHARACTERS BETWEEN 20H AND 7FH ARE TAKEN AS REFERENCES
1821:          ; THE SYSTEM STANDARD 5 X 7 CHARACTER FONT.  CHARACTERS
1822:          ; BETWEEN 0A0H AND 0FFH REFER TO THE USER SUPPLIED ALTERN
1823:          ; CHARACTER FONT.  THIS FONT IS DESCRIBED BY A FONT
1824:          ; DESCRIPTOR TABLE OF THE FOLLOWING FORMAT:
1825:          ; *****
1826:          ; * 0 * BASE CHARACTER VALUE      *
1827:          ; *****
1828:          ; * 1 * X FRAME SIZE              *
1829:          ; *****
1830:          ; * 2 * Y FRAME SIZE              *
1831:          ; *****
1832:          ; * 3 * X PATTERN SIZE (BYTES) *

```

Home Video Game System Library - System Subroutines

```

1833:          ; *****
1834:          ; * 4 * Y PATTERN SIZE          *
1835:          ; *****
1836:          ; * 5 *   PATTERN TABLE          *
1837:          ; * 6 *     ADDRESS              *
1838:          ; *****
1839: 07E1  C5          DISPCH  PUSH BC
1840: 07E2  E5          PUSH HL
1841: 07E3  DDE5       PUSH IX
1842: 07E5  A7          AND  A
1843: 07E6  FAED07     JP  M,DISCH1      ; JUMP IF YES
1844: 07E9  DD210602   LD  IX,SYSFNT
1845: 07ED  FE20       DISCH1  CP   20H          ; IS CHAR < 20H?
1846: 07EF  300D       JR  NC,$+(DISC1B-$) ; JUMP IF NOT
1847: 07F1  F5          DISC1A  PUSH AF          ; LOOP TO SPACE OVER
1848: 07F2  CD4E08     CALL NXTFRM
1849: 07F5  CDF40C     CALL FINDL3      ; STORE IT BACK
1850: 07F8  F1          POP  AF
1851: 07F9  3D          DEC  A
1852: 07FA  20F5       JR  NZ,$+(DISC1A-$)
1853: 07FC  183B       JR  $+(DISCH5-$) ; JUMP TO EXIT
1854: 07FE  DD9600     DISC1B  SUB  (IX+FTBASE) ; SUBTRACT BASE CHAR
1855: 0801  5F          LD  E,A
1856: 0802  1600       LD  D,0
1857: 0804  210000     LD  HL,0
1858: 0807  DD4E03     LD  C,(IX+FTBYTE) ; MULTIPLY CHARACTER
1859: 080A  DD4604     DISCH2  LD  B,(IX+FTYSIZ) ; BY PATTERN SIZE
1860: 080D  19          DISCH3  ADD  HL,DE
1861: 080E  10FD       DJNZ $+(DISCH3-$)
1862: 0810  0D          DEC  C
1863: 0811  20F7       JR  NZ,$+(DISCH2-$)
1864: 0813  DD5606     LD  D,(IX+FTPTH) ; ADD TO TABLE START
1865: 0816  DD5E05     LD  E,(IX+FTPTL)
1866: 0819  19          ADD  HL,DE
1867:          ; COMPUTE POSITION WHERE NEXT CHARACTER WOULD GO
1868:          ; AND SAVE
1869: 081A  CD4E08     CALL NXTFRM      ; STEP COORDINATES TO NEXT FRAM
1870: 081D  D5          PUSH DE          ; SAVE
1871: 081E  DD4604     LD  B,(IX+FTYSIZ)
1872: 0821  C5          DISCH4  PUSH BC
1873: 0822  E5          PUSH HL
1874: 0823  CD6C08     CALL WRTLIN
1875: 0826  E1          POP  HL
1876: 0827  DD4E03     LD  C,(IX+FTBYTE) ; STEP TO NEXT LINE OF PATTERN
1877: 082A  09          ADD  HL,BC
1878: 082B  C1          POP  BC
1879: 082C  FD7E05     LD  A,(IY+CBD)   ; ADVANCE Y COORDINATE
1880: 082F  81          ADD  A,C
1881: 0830  FD7705     LD  (IY+CBD),A
1882: 0833  10EC       DJNZ $+(DISCH4-$)
1883: 0835  D1          POP  DE          ; RESTORE NEW POSITION
1884: 0836  CDF40C     CALL FINDL3      ; STUFF DE BACK INTO CONTEXT
1885: 0839  DDE1       DISCH5  POP  IX
1886: 083B  E1          POP  HL
1887: 083C  C1          POP  BC
1888: 083D  C9          RET
1889:          ; SUBROUTINE TO CONVERT ENLARGEMENT FACTOR TO ITERATION C

```

Home Video Game System Library - System Subroutines

```

1890:           ; INPUT:           MODE BYTE FROM CONTEXT SAVE AREA
1891:           ; OUTPUT:          B,A = ITERATION COUNT
1892: 083E FD7E06      DCLCTB LD  A,(IY+CBC)   ; GET MODE BYTE
1893: 0841 07          RLCA
1894: 0842 07          RLCA
1895: 0843 E603          AND  03           ; ISOLATE ENLARGEMENT FACTOR
1896: 0845 3C          INC  A
1897: 0846 47          LD  B,A
1898: 0847 AF          XOR  A
1899: 0848 37          SCF
1900: 0849 8F          DCLCT1 ADC  A,A
1901: 084A 10FD         DJNZ $+(DCLCT1-$)
1902: 084C 47          LD  B,A
1903: 084D C9          RET
1904:           ; SUBROUTINE TO UPDATE COORDINATES TO POINT AT NEXT CHARA
1905:           ; FRAME:
1906:           ; INPUT:           COORDINATES TAKEN FROM CBD, CBE IN CONTEXT
1907:           ; OUTPUT:          UPDATED COORDINATES RETURNED IN D AND E
1908:           ; A,B = CLOBBERED C= ENLARGE FACTOR CONVERT
1909: 084E CD3E08      NXTFRM CALL DCLCTB   ; GET ITERATION COUNT
1910: 0851 48          LD  C,B           ; SAVE
1911: 0852 FD5605      LD  D,(IY+CBD)   ; GET Y COORD
1912: 0855 FD7E04      LD  A,(IY+CBE)   ; GET X COORD
1913: 0858 DD8601      NXTFR1 ADD  A,(IX+FTFSX) ; ADD X FRAME SIZE
1914: 085B 10FB         DJNZ $+(NXTFR1-$) ; 2**ENLARGE TIMES
1915: 085D FEA0          CP   160          ; PAST RIGHT EDGE OF SCREEN?
1916: 085F 3809         JR  C,$+(NXTFR3-$)
1917: 0861 7A          LD  A,D
1918: 0862 41          LD  B,C
1919: 0863 DD8602      NXTFR2 ADD  A,(IX+FTFSY) ; YEP - ADVANCE VERTICAL
1920: 0866 10FB         DJNZ $+(NXTFR2-$)
1921: 0868 57          LD  D,A
1922: 0869 AF          XOR  A
1923: 086A 5F          NXTFR3 LD  E,A
1924: 086B C9          RET
1925:           ; SUBROUTINE TO WRITE ONE LINE OF A PATTERN WITH ENLARGE
1926:           ; AND EXPAND
1927:           ; ENTRY: HL = SOURCE IX = FONT TABLE
1928: 086C DD4E03      WRTLIN LD  C,(IX+FTBYTE)
1929: 086F 0600          LD  B,0
1930: 0871 DDE5          PUSH IX           ; CAPTURE STACK POINTER
1931: 0873 DD210000     LD  IX,0
1932: 0877 DD39          ADD  IX,SP
1933: 0879 DDE5          PUSH IX           ; SAVE CAPTURED STACK
1934: 087B D1          POP  DE           ; DE = CAPTURED STACK
1935: 087C 3E0C         LD  A,0CH         ; SET EXPAND TO 00,11
1936: 087E D319         OUT  (XPAND),A
1937: 0880 3E08         LD  A,08H         ; SET EXPAND BIT
1938: 0882 D30C         OUT  (MAGIC),A
1939: 0884 FD7E06      LD  A,(IY+CBC)   ; GET CONTROL BYTE
1940: 0887 E6C0         AND  0C0H         ; ISOLATE ENLARGE AMOUNT
1941: 0889 2821         JR  Z,$+(WRTL3-$) ; JUMP IF ZERO
1942: 088B 07          RLCA
1943: 088C 07          RLCA
1944: 088D EB          WRTL1  EX  DE,HL
1945: 088E A7          AND  A           ; CLEAR CARRY BIT
1946: 088F ED42         SBC  HL,BC       ; COMPUTE STOCK FRAME SIZE

```

Home Video Game System Library - System Subroutines

```

1947: 0891 ED42          SBC  HL,BC
1948: 0893 F9           LD   SP,HL           ; SEIZE STACK SPACE
1949: 0894 CBB4          RES  6,H             ; MAGNIFY THE ADDRESS
1950: 0896 F5           PUSH AF
1951: 0897 41           LD   B,C
1952: 0898 1A           WRTL2 LD  A,(DE)         ; GET SOURCE BYTE
1953: 0899 13           INC  DE
1954: 089A 77           LD   (HL),A         ; EXPAND IT
1955: 089B 23           INC  HL
1956: 089C 77           LD   (HL),A
1957: 089D 23           INC  HL
1958: 089E 10F8          DJNZ $+(WRTL2-$)
1959: 08A0 CB21          SLA  C
1960: 08A2 F1           POP  AF
1961: 08A3 210000        LD   HL,0           ; CAPTURE STACK TOP AGAIN
1962: 08A6 39           ADD  HL,SP
1963: 08A7 54           LD   D,H            ; SET DE=HL
1964: 08A8 5D           LD   E,L            ; FOR NEXT DEST COMBO
1965: 08A9 3D           DEC  A
1966: 08AA 20E1          JR   NZ,$+(WRTL1-$)
1967:                ; NOW DO WRITE TO SCREEN
1968: 08AC CD3E08        WRTL3 CALL DCLCTB         ; GET ITERATION COUNTER
1969: 08AF CD7400          CALL DELOAD
1970: 08B2 FD7E06          LD   A,(IY+CBC)
1971: 08B5 D319          OUT  (XPAND),A
1972: 08B7 E630          AND  030H
1973: 08B9 F608          OR   8
1974: 08BB CD080B          CALL RELTA
1975: 08BE EB           EX  DE,HL
1976: 08BF F5           WRTL4 PUSH AF
1977: 08C0 C5           PUSH BC
1978: 08C1 D5           PUSH DE
1979: 08C2 E5           PUSH HL
1980: 08C3 41           LD   B,C
1981: 08C4 1A           WRTL5 LD  A,(DE)
1982: 08C5 13           INC  DE
1983: 08C6 77           LD   (HL),A
1984: 08C7 23           INC  HL
1985: 08C8 77           LD   (HL),A
1986: 08C9 23           INC  HL
1987: 08CA 10F8          DJNZ $+(WRTL5-$)
1988: 08CC FD7E04          LD   A,(IY+CBE)
1989: 08CF E603          AND  03
1990: 08D1 2801          JR   Z,$+(WRTL6-$)
1991: 08D3 70           LD   (HL),B
1992: 08D4 E1           WRTL6 POP  HL             ; STEP TO NEXT LINE
1993: 08D5 0E28          LD  C,BYTEPL
1994: 08D7 09           ADD  HL,BC
1995: 08D8 D1           POP  DE
1996: 08D9 C1           POP  BC
1997: 08DA F1           POP  AF
1998: 08DB D30C          OUT  (MAGIC),A
1999: 08DD 10E0          DJNZ $+(WRTL4-$)
2000: 08DF DDF9          LD   SP,IX          ; RESTORE STACK
2001: 08E1 DDE1          POP  IX
2002: 08E3 C9           RET
2003:                ;

```

Home Video Game System Library - System Subroutines

```

2004:          ; MACRO TO GENERATE CHARACTER PATTERN TABLE ENTRY
2005:          DEFCHR  MACRO AA, BB, CC, DD, EE, FF, GG
2006:              DB  AA
2007:              DB  BB
2008:              DB  CC
2009:              DB  DD
2010:              DB  EE
2011:              DB  FF
2012:              DB  GG
2013:              ENDM
2014:          ;
2015:          ; LARGE CHARACTER SET (8 X 8)
2016: 08E4          LRGCHR
2017: 08E4          DEFCHR 000H, 000H, 000H, 000H, 000H, 000H, 000H ;
SPACE
2017: 08E4 00          DB  000H
2017: 08E5 00          DB  000H
2017: 08E6 00          DB  000H
2017: 08E7 00          DB  000H
2017: 08E8 00          DB  000H
2017: 08E9 00          DB  000H
2017: 08EA 00          DB  000H
2017: 08EB          ENDM
2017:
2018: 08EB          DEFCHR 020H, 020H, 020H, 020H, 020H, 000H, 020H ;
!
2018: 08EB 20          DB  020H
2018: 08EC 20          DB  020H
2018: 08ED 20          DB  020H
2018: 08EE 20          DB  020H
2018: 08EF 20          DB  020H
2018: 08F0 00          DB  000H
2018: 08F1 20          DB  020H
2018: 08F2          ENDM
2018:
2019: 08F2          DEFCHR 050H, 050H, 050H, 000H, 000H, 000H, 000H ;
"
2019: 08F2 50          DB  050H
2019: 08F3 50          DB  050H
2019: 08F4 50          DB  050H
2019: 08F5 00          DB  000H
2019: 08F6 00          DB  000H
2019: 08F7 00          DB  000H
2019: 08F8 00          DB  000H
2019: 08F9          ENDM
2019:
2020: 08F9          DEFCHR 048H, 048H, 0FCH, 048H, 0FCH, 048H, 048H ;
#
2020: 08F9 48          DB  048H
2020: 08FA 48          DB  048H
2020: 08FB FC          DB  0FCH
2020: 08FC 48          DB  048H
2020: 08FD FC          DB  0FCH
2020: 08FE 48          DB  048H
2020: 08FF 48          DB  048H
2020: 0900          ENDM
2020:

```

Home Video Game System Library - System Subroutines

```

2021: 0900          DEFCHR 020H, 078H, 080H, 070H, 008H, 0F0H, 020H ;
$
2021: 0900  20      DB    020H
2021: 0901  78      DB    078H
2021: 0902  80      DB    080H
2021: 0903  70      DB    070H
2021: 0904  08      DB    008H
2021: 0905  F0      DB    0F0H
2021: 0906  20      DB    020H
2021: 0907          ENDM
2021:
2022: 0907          DEFCHR 0C0H, 0C8H, 010H, 020H, 040H, 098H, 018H ;
%
2022: 0907  C0      DB    0C0H
2022: 0908  C8      DB    0C8H
2022: 0909  10      DB    010H
2022: 090A  20      DB    020H
2022: 090B  40      DB    040H
2022: 090C  98      DB    098H
2022: 090D  18      DB    018H
2022: 090E          ENDM
2022:
2023: 090E          DEFCHR 060H, 090H, 0A0H, 040H, 0A8H, 090H, 068H ;
&
2023: 090E  60      DB    060H
2023: 090F  90      DB    090H
2023: 0910  A0      DB    0A0H
2023: 0911  40      DB    040H
2023: 0912  A8      DB    0A8H
2023: 0913  90      DB    090H
2023: 0914  68      DB    068H
2023: 0915          ENDM
2023:
2024: 0915          DEFCHR 060H, 060H, 060H, 000H, 000H, 000H, 000H ;
,
2024: 0915  60      DB    060H
2024: 0916  60      DB    060H
2024: 0917  60      DB    060H
2024: 0918  00      DB    000H
2024: 0919  00      DB    000H
2024: 091A  00      DB    000H
2024: 091B  00      DB    000H
2024: 091C          ENDM
2024:
2025: 091C          DEFCHR 010H, 020H, 020H, 020H, 020H, 020H, 010H ;
(
2025: 091C  10      DB    010H
2025: 091D  20      DB    020H
2025: 091E  20      DB    020H
2025: 091F  20      DB    020H
2025: 0920  20      DB    020H
2025: 0921  20      DB    020H
2025: 0922  10      DB    010H
2025: 0923          ENDM
2025:
2026: 0923          DEFCHR 040H, 020H, 020H, 020H, 020H, 020H, 040H ;
)

```

Home Video Game System Library - System Subroutines

```

2026: 0923 40          DB 040H
2026: 0924 20          DB 020H
2026: 0925 20          DB 020H
2026: 0926 20          DB 020H
2026: 0927 20          DB 020H
2026: 0928 20          DB 020H
2026: 0929 40          DB 040H
2026: 092A          ENDM
2026:
2027: 092A          DEFCHR 000H, 0A8H, 070H, 0D8H, 070H, 0A8H, 000H ;
*
2027: 092A 00          DB 000H
2027: 092B A8          DB 0A8H
2027: 092C 70          DB 070H
2027: 092D D8          DB 0D8H
2027: 092E 70          DB 070H
2027: 092F A8          DB 0A8H
2027: 0930 00          DB 000H
2027: 0931          ENDM
2027:
2028: 0931          DEFCHR 000H, 020H, 020H, 0F8H, 020H, 020H, 000H ;
+
2028: 0931 00          DB 000H
2028: 0932 20          DB 020H
2028: 0933 20          DB 020H
2028: 0934 F8          DB 0F8H
2028: 0935 20          DB 020H
2028: 0936 20          DB 020H
2028: 0937 00          DB 000H
2028: 0938          ENDM
2028:
2029: 0938          DEFCHR 000H, 000H, 000H, 060H, 060H, 020H, 040H ;
,
2029: 0938 00          DB 000H
2029: 0939 00          DB 000H
2029: 093A 00          DB 000H
2029: 093B 60          DB 060H
2029: 093C 60          DB 060H
2029: 093D 20          DB 020H
2029: 093E 40          DB 040H
2029: 093F          ENDM
2029:
2030: 093F          DEFCHR 000H, 000H, 000H, 0F8H, 000H, 000H, 000H ;
-
2030: 093F 00          DB 000H
2030: 0940 00          DB 000H
2030: 0941 00          DB 000H
2030: 0942 F8          DB 0F8H
2030: 0943 00          DB 000H
2030: 0944 00          DB 000H
2030: 0945 00          DB 000H
2030: 0946          ENDM
2030:
2031: 0946          DEFCHR 000H, 000H, 000H, 000H, 000H, 060H, 060H ;
.
2031: 0946 00          DB 000H
2031: 0947 00          DB 000H

```

Home Video Game System Library - System Subroutines

```

2031: 0948 00          DB  000H
2031: 0949 00          DB  000H
2031: 094A 00          DB  000H
2031: 094B 60          DB  060H
2031: 094C 60          DB  060H
2031: 094D          ENDM
2031:
2032: 094D          DEFCHR 000H, 008H, 010H, 020H, 040H, 080H, 000H ;
/
2032: 094D 00          DB  000H
2032: 094E 08          DB  008H
2032: 094F 10          DB  010H
2032: 0950 20          DB  020H
2032: 0951 40          DB  040H
2032: 0952 80          DB  080H
2032: 0953 00          DB  000H
2032: 0954          ENDM
2032:
2033: 0954          DEFCHR 070H, 088H, 088H, 088H, 088H, 088H, 070H ;
0
2033: 0954 70          DB  070H
2033: 0955 88          DB  088H
2033: 0956 88          DB  088H
2033: 0957 88          DB  088H
2033: 0958 88          DB  088H
2033: 0959 88          DB  088H
2033: 095A 70          DB  070H
2033: 095B          ENDM
2033:
2034: 095B          DEFCHR 020H, 060H, 020H, 020H, 020H, 020H, 070H ;
1
2034: 095B 20          DB  020H
2034: 095C 60          DB  060H
2034: 095D 20          DB  020H
2034: 095E 20          DB  020H
2034: 095F 20          DB  020H
2034: 0960 20          DB  020H
2034: 0961 70          DB  070H
2034: 0962          ENDM
2034:
2035: 0962          DEFCHR 070H, 088H, 008H, 070H, 080H, 080H, 0F8H ;
2
2035: 0962 70          DB  070H
2035: 0963 88          DB  088H
2035: 0964 08          DB  008H
2035: 0965 70          DB  070H
2035: 0966 80          DB  080H
2035: 0967 80          DB  080H
2035: 0968 F8          DB  0F8H
2035: 0969          ENDM
2035:
2036: 0969          DEFCHR 070H, 088H, 008H, 030H, 008H, 088H, 070H ;
3
2036: 0969 70          DB  070H
2036: 096A 88          DB  088H
2036: 096B 08          DB  008H
2036: 096C 30          DB  030H

```

Home Video Game System Library - System Subroutines

```

2036: 096D 08          DB  008H
2036: 096E 88          DB  088H
2036: 096F 70          DB  070H
2036: 0970          ENDM
2036:
2037: 0970          DEFCHR 010H, 030H, 050H, 090H, 0F8H, 010H, 010H ;
4
2037: 0970 10          DB  010H
2037: 0971 30          DB  030H
2037: 0972 50          DB  050H
2037: 0973 90          DB  090H
2037: 0974 F8          DB  0F8H
2037: 0975 10          DB  010H
2037: 0976 10          DB  010H
2037: 0977          ENDM
2037:
2038: 0977          DEFCHR 0F8H, 080H, 0F0H, 008H, 008H, 088H, 070H ;
5
2038: 0977 F8          DB  0F8H
2038: 0978 80          DB  080H
2038: 0979 F0          DB  0F0H
2038: 097A 08          DB  008H
2038: 097B 08          DB  008H
2038: 097C 88          DB  088H
2038: 097D 70          DB  070H
2038: 097E          ENDM
2038:
2039: 097E          DEFCHR 030H, 040H, 080H, 0F0H, 088H, 088H, 070H ;
6
2039: 097E 30          DB  030H
2039: 097F 40          DB  040H
2039: 0980 80          DB  080H
2039: 0981 F0          DB  0F0H
2039: 0982 88          DB  088H
2039: 0983 88          DB  088H
2039: 0984 70          DB  070H
2039: 0985          ENDM
2039:
2040: 0985          DEFCHR 0F8H, 008H, 010H, 020H, 040H, 040H, 040H ;
7
2040: 0985 F8          DB  0F8H
2040: 0986 08          DB  008H
2040: 0987 10          DB  010H
2040: 0988 20          DB  020H
2040: 0989 40          DB  040H
2040: 098A 40          DB  040H
2040: 098B 40          DB  040H
2040: 098C          ENDM
2040:
2041: 098C          DEFCHR 070H, 088H, 088H, 070H, 088H, 088H, 070H ;
8
2041: 098C 70          DB  070H
2041: 098D 88          DB  088H
2041: 098E 88          DB  088H
2041: 098F 70          DB  070H
2041: 0990 88          DB  088H
2041: 0991 88          DB  088H

```

Home Video Game System Library - System Subroutines

```

2041: 0992 70          DB  070H
2041: 0993          ENDM
2041:
2042: 0993          DEFCHR 070H, 088H, 088H, 078H, 008H, 010H, 060H ;
9
2042: 0993 70          DB  070H
2042: 0994 88          DB  088H
2042: 0995 88          DB  088H
2042: 0996 78          DB  078H
2042: 0997 08          DB  008H
2042: 0998 10          DB  010H
2042: 0999 60          DB  060H
2042: 099A          ENDM
2042:
2043: 099A          DEFCHR 000H, 060H, 060H, 000H, 060H, 060H, 000H ;
:
2043: 099A 00          DB  000H
2043: 099B 60          DB  060H
2043: 099C 60          DB  060H
2043: 099D 00          DB  000H
2043: 099E 60          DB  060H
2043: 099F 60          DB  060H
2043: 09A0 00          DB  000H
2043: 09A1          ENDM
2043:
2044: 09A1          DEFCHR 060H, 060H, 000H, 060H, 060H, 020H, 040H ;
;
2044: 09A1 60          DB  060H
2044: 09A2 60          DB  060H
2044: 09A3 00          DB  000H
2044: 09A4 60          DB  060H
2044: 09A5 60          DB  060H
2044: 09A6 20          DB  020H
2044: 09A7 40          DB  040H
2044: 09A8          ENDM
2044:
2045: 09A8          DEFCHR 010H, 020H, 040H, 080H, 040H, 020H, 010H ;
<
2045: 09A8 10          DB  010H
2045: 09A9 20          DB  020H
2045: 09AA 40          DB  040H
2045: 09AB 80          DB  080H
2045: 09AC 40          DB  040H
2045: 09AD 20          DB  020H
2045: 09AE 10          DB  010H
2045: 09AF          ENDM
2045:
2046: 09AF          DEFCHR 000H, 000H, 0F8H, 000H, 0F8H, 000H, 000H ;
=
2046: 09AF 00          DB  000H
2046: 09B0 00          DB  000H
2046: 09B1 F8          DB  0F8H
2046: 09B2 00          DB  000H
2046: 09B3 F8          DB  0F8H
2046: 09B4 00          DB  000H
2046: 09B5 00          DB  000H
2046: 09B6          ENDM

```

Home Video Game System Library - System Subroutines

```

2046:
2047: 09B6          DEFCHR 040H, 020H, 010H, 008H, 010H, 020H, 040H ;
>
2047: 09B6  40      DB    040H
2047: 09B7  20      DB    020H
2047: 09B8  10      DB    010H
2047: 09B9  08      DB    008H
2047: 09BA  10      DB    010H
2047: 09BB  20      DB    020H
2047: 09BC  40      DB    040H
2047: 09BD          ENDM
2047:
2048: 09BD          DEFCHR 070H, 088H, 008H, 010H, 020H, 000H, 020H ;
?
2048: 09BD  70      DB    070H
2048: 09BE  88      DB    088H
2048: 09BF  08      DB    008H
2048: 09C0  10      DB    010H
2048: 09C1  20      DB    020H
2048: 09C2  00      DB    000H
2048: 09C3  20      DB    020H
2048: 09C4          ENDM
2048:
2049: 09C4          DEFCHR 070H, 088H, 0B8H, 0A8H, 0B8H, 080H, 078H ;
@
2049: 09C4  70      DB    070H
2049: 09C5  88      DB    088H
2049: 09C6  B8      DB    0B8H
2049: 09C7  A8      DB    0A8H
2049: 09C8  B8      DB    0B8H
2049: 09C9  80      DB    080H
2049: 09CA  78      DB    078H
2049: 09CB          ENDM
2049:
2050: 09CB          DEFCHR 070H, 088H, 088H, 0F8H, 088H, 088H, 088H ;
A
2050: 09CB  70      DB    070H
2050: 09CC  88      DB    088H
2050: 09CD  88      DB    088H
2050: 09CE  F8      DB    0F8H
2050: 09CF  88      DB    088H
2050: 09D0  88      DB    088H
2050: 09D1  88      DB    088H
2050: 09D2          ENDM
2050:
2051: 09D2          DEFCHR 0F0H, 088H, 088H, 0F0H, 088H, 088H, 0F0H ;
B
2051: 09D2  F0      DB    0F0H
2051: 09D3  88      DB    088H
2051: 09D4  88      DB    088H
2051: 09D5  F0      DB    0F0H
2051: 09D6  88      DB    088H
2051: 09D7  88      DB    088H
2051: 09D8  F0      DB    0F0H
2051: 09D9          ENDM
2051:

```

Home Video Game System Library - System Subroutines

```

2052: 09D9          DEFCHR 070H, 088H, 080H, 080H, 080H, 088H, 070H ;
C
2052: 09D9  70      DB  070H
2052: 09DA  88      DB  088H
2052: 09DB  80      DB  080H
2052: 09DC  80      DB  080H
2052: 09DD  80      DB  080H
2052: 09DE  88      DB  088H
2052: 09DF  70      DB  070H
2052: 09E0          ENDM
2052:
2053: 09E0          DEFCHR 0F0H, 088H, 088H, 088H, 088H, 088H, 0F0H ;
D
2053: 09E0  F0      DB  0F0H
2053: 09E1  88      DB  088H
2053: 09E2  88      DB  088H
2053: 09E3  88      DB  088H
2053: 09E4  88      DB  088H
2053: 09E5  88      DB  088H
2053: 09E6  F0      DB  0F0H
2053: 09E7          ENDM
2053:
2054: 09E7          DEFCHR 0F8H, 080H, 080H, 0E0H, 080H, 080H, 0F8H ;
E
2054: 09E7  F8      DB  0F8H
2054: 09E8  80      DB  080H
2054: 09E9  80      DB  080H
2054: 09EA  E0      DB  0E0H
2054: 09EB  80      DB  080H
2054: 09EC  80      DB  080H
2054: 09ED  F8      DB  0F8H
2054: 09EE          ENDM
2054:
2055: 09EE          DEFCHR 0F8H, 080H, 080H, 0E0H, 080H, 080H, 080H ;
F
2055: 09EE  F8      DB  0F8H
2055: 09EF  80      DB  080H
2055: 09F0  80      DB  080H
2055: 09F1  E0      DB  0E0H
2055: 09F2  80      DB  080H
2055: 09F3  80      DB  080H
2055: 09F4  80      DB  080H
2055: 09F5          ENDM
2055:
2056: 09F5          DEFCHR 070H, 088H, 080H, 080H, 098H, 088H, 078H ;
G
2056: 09F5  70      DB  070H
2056: 09F6  88      DB  088H
2056: 09F7  80      DB  080H
2056: 09F8  80      DB  080H
2056: 09F9  98      DB  098H
2056: 09FA  88      DB  088H
2056: 09FB  78      DB  078H
2056: 09FC          ENDM
2056:
2057: 09FC          DEFCHR 088H, 088H, 088H, 0F8H, 088H, 088H, 088H ;
H

```

Home Video Game System Library - System Subroutines

```

2057: 09FC 88          DB 088H
2057: 09FD 88          DB 088H
2057: 09FE 88          DB 088H
2057: 09FF F8          DB 0F8H
2057: 0A00 88          DB 088H
2057: 0A01 88          DB 088H
2057: 0A02 88          DB 088H
2057: 0A03          ENDM
2057:
2058: 0A03          DEFCHR 070H, 020H, 020H, 020H, 020H, 020H, 070H ;
I
2058: 0A03 70          DB 070H
2058: 0A04 20          DB 020H
2058: 0A05 20          DB 020H
2058: 0A06 20          DB 020H
2058: 0A07 20          DB 020H
2058: 0A08 20          DB 020H
2058: 0A09 70          DB 070H
2058: 0A0A          ENDM
2058:
2059: 0A0A          DEFCHR 008H, 008H, 008H, 008H, 008H, 088H, 070H ;
J
2059: 0A0A 08          DB 008H
2059: 0A0B 08          DB 008H
2059: 0A0C 08          DB 008H
2059: 0A0D 08          DB 008H
2059: 0A0E 08          DB 008H
2059: 0A0F 88          DB 088H
2059: 0A10 70          DB 070H
2059: 0A11          ENDM
2059:
2060: 0A11          DEFCHR 088H, 090H, 0A0H, 0C0H, 0A0H, 090H, 088H ;
K
2060: 0A11 88          DB 088H
2060: 0A12 90          DB 090H
2060: 0A13 A0          DB 0A0H
2060: 0A14 C0          DB 0C0H
2060: 0A15 A0          DB 0A0H
2060: 0A16 90          DB 090H
2060: 0A17 88          DB 088H
2060: 0A18          ENDM
2060:
2061: 0A18          DEFCHR 080H, 080H, 080H, 080H, 080H, 080H, 0F8H ;
L
2061: 0A18 80          DB 080H
2061: 0A19 80          DB 080H
2061: 0A1A 80          DB 080H
2061: 0A1B 80          DB 080H
2061: 0A1C 80          DB 080H
2061: 0A1D 80          DB 080H
2061: 0A1E F8          DB 0F8H
2061: 0A1F          ENDM
2061:
2062: 0A1F          DEFCHR 088H, 0D8H, 0A8H, 0A8H, 088H, 088H, 088H ;
M
2062: 0A1F 88          DB 088H
2062: 0A20 D8          DB 0D8H

```

Home Video Game System Library - System Subroutines

```

2062: 0A21  A8          DB   0A8H
2062: 0A22  A8          DB   0A8H
2062: 0A23  88          DB   088H
2062: 0A24  88          DB   088H
2062: 0A25  88          DB   088H
2062: 0A26                ENDM
2062:
2063: 0A26                DEFCHR 088H, 0C8H, 0A8H, 098H, 088H, 088H, 088H ;
N
2063: 0A26  88          DB   088H
2063: 0A27  C8          DB   0C8H
2063: 0A28  A8          DB   0A8H
2063: 0A29  98          DB   098H
2063: 0A2A  88          DB   088H
2063: 0A2B  88          DB   088H
2063: 0A2C  88          DB   088H
2063: 0A2D                ENDM
2063:
2064: 0A2D                DEFCHR 0F8H, 088H, 088H, 088H, 088H, 088H, 0F8H ;
O
2064: 0A2D  F8          DB   0F8H
2064: 0A2E  88          DB   088H
2064: 0A2F  88          DB   088H
2064: 0A30  88          DB   088H
2064: 0A31  88          DB   088H
2064: 0A32  88          DB   088H
2064: 0A33  F8          DB   0F8H
2064: 0A34                ENDM
2064:
2065: 0A34                DEFCHR 0F0H, 088H, 088H, 0F0H, 080H, 080H, 080H ;
P
2065: 0A34  F0          DB   0F0H
2065: 0A35  88          DB   088H
2065: 0A36  88          DB   088H
2065: 0A37  F0          DB   0F0H
2065: 0A38  80          DB   080H
2065: 0A39  80          DB   080H
2065: 0A3A  80          DB   080H
2065: 0A3B                ENDM
2065:
2066: 0A3B                DEFCHR 070H, 088H, 088H, 088H, 0A8H, 090H, 068H ;
Q
2066: 0A3B  70          DB   070H
2066: 0A3C  88          DB   088H
2066: 0A3D  88          DB   088H
2066: 0A3E  88          DB   088H
2066: 0A3F  A8          DB   0A8H
2066: 0A40  90          DB   090H
2066: 0A41  68          DB   068H
2066: 0A42                ENDM
2066:
2067: 0A42                DEFCHR 0F0H, 088H, 088H, 0F0H, 0A0H, 090H, 088H ;
R
2067: 0A42  F0          DB   0F0H
2067: 0A43  88          DB   088H
2067: 0A44  88          DB   088H
2067: 0A45  F0          DB   0F0H

```

Home Video Game System Library - System Subroutines

```

2067: 0A46 A0          DB 0A0H
2067: 0A47 90          DB 090H
2067: 0A48 88          DB 088H
2067: 0A49             ENDM
2067:
2068: 0A49             DEFCHR 070H, 088H, 080H, 070H, 008H, 088H, 070H ;
S
2068: 0A49 70          DB 070H
2068: 0A4A 88          DB 088H
2068: 0A4B 80          DB 080H
2068: 0A4C 70          DB 070H
2068: 0A4D 08          DB 008H
2068: 0A4E 88          DB 088H
2068: 0A4F 70          DB 070H
2068: 0A50             ENDM
2068:
2069: 0A50             DEFCHR 0F8H, 020H, 020H, 020H, 020H, 020H, 020H ;
T
2069: 0A50 F8          DB 0F8H
2069: 0A51 20          DB 020H
2069: 0A52 20          DB 020H
2069: 0A53 20          DB 020H
2069: 0A54 20          DB 020H
2069: 0A55 20          DB 020H
2069: 0A56 20          DB 020H
2069: 0A57             ENDM
2069:
2070: 0A57             DEFCHR 088H, 088H, 088H, 088H, 088H, 088H, 070H ;
U
2070: 0A57 88          DB 088H
2070: 0A58 88          DB 088H
2070: 0A59 88          DB 088H
2070: 0A5A 88          DB 088H
2070: 0A5B 88          DB 088H
2070: 0A5C 88          DB 088H
2070: 0A5D 70          DB 070H
2070: 0A5E             ENDM
2070:
2071: 0A5E             DEFCHR 088H, 088H, 088H, 050H, 050H, 020H, 020H ;
V
2071: 0A5E 88          DB 088H
2071: 0A5F 88          DB 088H
2071: 0A60 88          DB 088H
2071: 0A61 50          DB 050H
2071: 0A62 50          DB 050H
2071: 0A63 20          DB 020H
2071: 0A64 20          DB 020H
2071: 0A65             ENDM
2071:
2072: 0A65             DEFCHR 088H, 088H, 088H, 0A8H, 0A8H, 0D8H, 088H ;
W
2072: 0A65 88          DB 088H
2072: 0A66 88          DB 088H
2072: 0A67 88          DB 088H
2072: 0A68 A8          DB 0A8H
2072: 0A69 A8          DB 0A8H
2072: 0A6A D8          DB 0D8H

```

Home Video Game System Library - System Subroutines

```

2072: 0A6B 88          DB  088H
2072: 0A6C          ENDM
2072:
2073: 0A6C          DEFCHR 088H, 088H, 050H, 020H, 050H, 088H, 088H ;
X
2073: 0A6C 88          DB  088H
2073: 0A6D 88          DB  088H
2073: 0A6E 50          DB  050H
2073: 0A6F 20          DB  020H
2073: 0A70 50          DB  050H
2073: 0A71 88          DB  088H
2073: 0A72 88          DB  088H
2073: 0A73          ENDM
2073:
2074: 0A73          DEFCHR 088H, 088H, 050H, 020H, 020H, 020H, 020H ;
Y
2074: 0A73 88          DB  088H
2074: 0A74 88          DB  088H
2074: 0A75 50          DB  050H
2074: 0A76 20          DB  020H
2074: 0A77 20          DB  020H
2074: 0A78 20          DB  020H
2074: 0A79 20          DB  020H
2074: 0A7A          ENDM
2074:
2075: 0A7A          DEFCHR 0F8H, 008H, 010H, 020H, 040H, 080H, 0F8H ;
Z
2075: 0A7A F8          DB  0F8H
2075: 0A7B 08          DB  008H
2075: 0A7C 10          DB  010H
2075: 0A7D 20          DB  020H
2075: 0A7E 40          DB  040H
2075: 0A7F 80          DB  080H
2075: 0A80 F8          DB  0F8H
2075: 0A81          ENDM
2075:
2076: 0A81          DEFCHR 070H, 040H, 040H, 040H, 040H, 040H, 070H ;
[
2076: 0A81 70          DB  070H
2076: 0A82 40          DB  040H
2076: 0A83 40          DB  040H
2076: 0A84 40          DB  040H
2076: 0A85 40          DB  040H
2076: 0A86 40          DB  040H
2076: 0A87 70          DB  070H
2076: 0A88          ENDM
2076:
2077: 0A88          DEFCHR 000H, 080H, 040H, 020H, 010H, 008H, 000H ;
\
2077: 0A88 00          DB  000H
2077: 0A89 80          DB  080H
2077: 0A8A 40          DB  040H
2077: 0A8B 20          DB  020H
2077: 0A8C 10          DB  010H
2077: 0A8D 08          DB  008H
2077: 0A8E 00          DB  000H
2077: 0A8F          ENDM

```

Home Video Game System Library - System Subroutines

```

2077:
2078: 0A8F          DEFCHR 070H, 010H, 010H, 010H, 010H, 010H, 070H ;
]
2078: 0A8F 70      DB 070H
2078: 0A90 10      DB 010H
2078: 0A91 10      DB 010H
2078: 0A92 10      DB 010H
2078: 0A93 10      DB 010H
2078: 0A94 10      DB 010H
2078: 0A95 70      DB 070H
2078: 0A96          ENDM
2078:
2079: 0A96          DEFCHR 020H, 070H, 0A8H, 020H, 020H, 020H, 020H ;
^
2079: 0A96 20      DB 020H
2079: 0A97 70      DB 070H
2079: 0A98 A8      DB 0A8H
2079: 0A99 20      DB 020H
2079: 0A9A 20      DB 020H
2079: 0A9B 20      DB 020H
2079: 0A9C 20      DB 020H
2079: 0A9D          ENDM
2079:
2080: 0A9D          DEFCHR 000H, 020H, 040H, 0F8H, 040H, 020H, 000H ;
<-
2080: 0A9D 00      DB 000H
2080: 0A9E 20      DB 020H
2080: 0A9F 40      DB 040H
2080: 0AA0 F8      DB 0F8H
2080: 0AA1 40      DB 040H
2080: 0AA2 20      DB 020H
2080: 0AA3 00      DB 000H
2080: 0AA4          ENDM
2080:
2081: 0AA4          DEFCHR 020H, 020H, 020H, 020H, 0A8H, 070H, 020H ;
DOWN
2081: 0AA4 20      DB 020H
2081: 0AA5 20      DB 020H
2081: 0AA6 20      DB 020H
2081: 0AA7 20      DB 020H
2081: 0AA8 A8      DB 0A8H
2081: 0AA9 70      DB 070H
2081: 0AAA 20      DB 020H
2081: 0AAB          ENDM
2081:
2082: 0AAB          DEFCHR 000H, 020H, 010H, 0F8H, 010H, 020H, 000H ;
RIGHT
2082: 0AAB 00      DB 000H
2082: 0AAC 20      DB 020H
2082: 0AAD 10      DB 010H
2082: 0AAE F8      DB 0F8H
2082: 0AAF 10      DB 010H
2082: 0AB0 20      DB 020H
2082: 0AB1 00      DB 000H
2082: 0AB2          ENDM
2082:

```

Home Video Game System Library - System Subroutines

```

2083: 0AB2          DEFCHR 000H, 088H, 050H, 020H, 050H, 088H, 000H ;
MULTI
2083: 0AB2  00          DB  000H
2083: 0AB3  88          DB  088H
2083: 0AB4  50          DB  050H
2083: 0AB5  20          DB  020H
2083: 0AB6  50          DB  050H
2083: 0AB7  88          DB  088H
2083: 0AB8  00          DB  000H
2083: 0AB9          ENDM
2083:
2084: 0AB9  00          DB  0
2085: 0ABA  20          DB  20H
2086: 0ABB  00          DB  0
2087: 0ABC  F8          DB  0F8H
2088: 0ABD  00          DB  0
2089: 0ABE  20          DB  20H
2090:                ; ** LAST BYTE OF DIVIDE IS ZERO, WHICH HAPPENS TO BE FIR
2091:                ;   BYTE OF...
2092:                ; SMALL CHARACTERS (4 X 6)
2093: 0ABF          SMLCHR
2094: 0ABF          DEF5 000H, 000H, 000H, 000H, 000H ; SPACE
2094: 0ABF  00          DB  000H
2094: 0AC0  00          DB  000H
2094: 0AC1  00          DB  000H
2094: 0AC2  00          DB  000H
2094: 0AC3  00          DB  000H
2094: 0AC4          ENDM
2094:
2095:                ;
2096: 0AC4  DDE1          MMJUMP POP IX
2097: 0AC6  E3           EX  (SP),HL
2098: 0AC7  DDE9          JP  (IX)
2099:                ;
2100:                ; NAME:   CONVERT KEY CODE TO ASCII
2101:                ; PURPOSE: SAME
2102:                ; INPUT:   A=KEY CODE
2103:                ; OUTPUT:  A=ASCII EQUIVALENT
2104:                ; HOW:    TABLE LOOKUP
2105: 0AC9          MKCTAS
2106: 0AC9  48           LD  C,B
2107: 0ACA  0600         LD  B,0
2108: 0ACC  21D50A       LD  HL,KCTATB
2109: 0ACF  09           ADD HL,BC
2110: 0AD0  7E           LD  A,(HL)
2111: 0AD1  FD7709       QFROG LD  (IY+CBA),A
2112: 0AD4  C9           RET
2113:                ;
2114: 0AD5          KCTATB
2115: 0AD5  20           DB  ' '          ; SPACE
2116: 0AD6  43           DB  'C'          ; BULLET
2117: 0AD7  5E           DB  5EH          ; UP ARROW
2118: 0AD8  5C           DB  5CH          ; DOWN ARROW
2119: 0AD9  25           DB  '%'          ;
2120: 0ADA  52           DB  'R'          ; RECALL
2121: 0ADB  53           DB  'S'          ; STORE
2122: 0ADC  3B           DB  ';'          ; PLUS MINUS

```

Home Video Game System Library - System Subroutines

```

2123: 0ADD 2F          DB  '/'          ; DIVIDE
2124: 0ADE 37          DB  '7'
2125: 0ADF 38          DB  '8'
2126: 0AE0 39          DB  '9'
2127: 0AE1 2A          DB  '*'          ; TIMES
2128: 0AE2 34          DB  '4'
2129: 0AE3 35          DB  '5'
2130: 0AE4 36          DB  '6'
2131: 0AE5 2D          DB  '-'          ; MINUS
2132: 0AE6 31          DB  '1'
2133: 0AE7 32          DB  '2'
2134: 0AE8 33          DB  '3'
2135: 0AE9 2B          DB  '+'          ; PLUS
2136: 0AEA 26          DB  '&'          ; CE
2137: 0AEB 30          DB  '0'
2138: 0AEC 2E          DB  '.'          ; POINT
2139: 0AED 3D          DB  '='          ; EQUALS
2140:                   ;
2141:                   ; NAME:          FILL AREA
2142:                   ; PURPOSE:       SET REGION OF SCREEN TO CONSTANT VALUE
2143:                   ; INPUT:         A = DATA TO FILL WITH
2144:                   ;                   BC = NUMBER OF BYTES TO FILL
2145:                   ;                   DE = STARTING ADDRESS OF REGION TO FILL
2146: 0AEE EB          MFILL EX  DE,HL
2147: 0AEF 77          MFILL1 LD  (HL),A      ; STUFF BYTE
2148: 0AF0 ED1         CPI                    ; BUMP HL, DEC BC
2149: 0AF2 EAEF0A      JP  PE,MFILL1
2150: 0AF5 C9          RET
2151:                   ;
2152:                   ; NAME:          RELATIVE TO ABSOLUTE
2153:                   ; PURPOSE:       COORDINATE CONVERSION
2154:                   ; INPUT:         E = X COORDINATE
2155:                   ;                   D = Y COORDINATE
2156:                   ;                   A = MAGIC REGISTER
2157:                   ; OUTPUT:       DE = ABSOLUTE ADDRESS
2158:                   ;                   A = MAGIC REGISTER TO USE
2159:                   ; MAGIC ENTRY POINT
2160: 0AF6 CD080B      MRELAB CALL RELTA
2161: 0AF9 1805        JR   $+(MRELA2-$)
2162:                   ; NONMAGIC ENTRY POINT
2163: 0AFB CD4E0B      MRELA1 CALL RELTA1
2164: 0AFE CBF2        SET  6,D          ; NONMAGIC THE ADDRESS
2165: 0B00 FD7304      MRELA2 LD  (IY+CBE),E      ; UPDATE CB DE
2166: 0B03 FD7205      LD  (IY+CBD),D
2167: 0B06 18C9        MFROG JR   $+(QFROG-$)
2168:                   ; MAGIC ENTRY POINT
2169: 0B08 CD4E0B      RELTA  CALL RELTA1
2170: 0B0B D30C        OUT  (MAGIC),A
2171: 0B0D C9          RET
2172: 0B0E 00          CKSUM2 DB  0          ; *** CHECKSUM ***
2173: 0B0F             DEF5 0E0H, 0A0H, 0A0H, 0A0H, 0E0H ; 0
2173: 0B0F E0          DB  0E0H
2173: 0B10 A0          DB  0A0H
2173: 0B11 A0          DB  0A0H
2173: 0B12 A0          DB  0A0H
2173: 0B13 E0          DB  0E0H
2173: 0B14             ENDM

```

Home Video Game System Library - System Subroutines

```

2173:
2174: 0B14          DEF5 040H, 040H, 040H, 040H, 040H ; 1
2174: 0B14 40       DB    040H
2174: 0B15 40       DB    040H
2174: 0B16 40       DB    040H
2174: 0B17 40       DB    040H
2174: 0B18 40       DB    040H
2174: 0B19          ENDM
2174:
2175: 0B19          DEF5 0E0H, 020H, 0E0H, 080H, 0E0H ; 2
2175: 0B19 E0       DB    0E0H
2175: 0B1A 20       DB    020H
2175: 0B1B E0       DB    0E0H
2175: 0B1C 80       DB    080H
2175: 0B1D E0       DB    0E0H
2175: 0B1E          ENDM
2175:
2176: 0B1E          DEF5 0E0H, 020H, 060H, 020H, 0E0H ; 3
2176: 0B1E E0       DB    0E0H
2176: 0B1F 20       DB    020H
2176: 0B20 60       DB    060H
2176: 0B21 20       DB    020H
2176: 0B22 E0       DB    0E0H
2176: 0B23          ENDM
2176:
2177: 0B23          DEF5 0A0H, 0A0H, 0E0H, 020H, 020H ; 4
2177: 0B23 A0       DB    0A0H
2177: 0B24 A0       DB    0A0H
2177: 0B25 E0       DB    0E0H
2177: 0B26 20       DB    020H
2177: 0B27 20       DB    020H
2177: 0B28          ENDM
2177:
2178: 0B28          DEF5 0E0H, 080H, 0E0H, 020H, 0E0H ; 5
2178: 0B28 E0       DB    0E0H
2178: 0B29 80       DB    080H
2178: 0B2A E0       DB    0E0H
2178: 0B2B 20       DB    020H
2178: 0B2C E0       DB    0E0H
2178: 0B2D          ENDM
2178:
2179: 0B2D          DEF5 0E0H, 080H, 0E0H, 0A0H, 0E0H ; 6
2179: 0B2D E0       DB    0E0H
2179: 0B2E 80       DB    080H
2179: 0B2F E0       DB    0E0H
2179: 0B30 A0       DB    0A0H
2179: 0B31 E0       DB    0E0H
2179: 0B32          ENDM
2179:
2180: 0B32          DEF5 0E0H, 020H, 020H, 020H, 020H ; 7
2180: 0B32 E0       DB    0E0H
2180: 0B33 20       DB    020H
2180: 0B34 20       DB    020H
2180: 0B35 20       DB    020H
2180: 0B36 20       DB    020H
2180: 0B37          ENDM
2180:

```

Home Video Game System Library - System Subroutines

```

2181: 0B37          DEF5 0E0H, 0A0H, 0E0H, 0A0H, 0E0H ; 8
2181: 0B37 E0      DB    0E0H
2181: 0B38 A0      DB    0A0H
2181: 0B39 E0      DB    0E0H
2181: 0B3A A0      DB    0A0H
2181: 0B3B E0      DB    0E0H
2181: 0B3C          ENDM
2181:
2182: 0B3C          DEF5 0E0H, 0A0H, 0E0H, 020H, 0E0H ; 9
2182: 0B3C E0      DB    0E0H
2182: 0B3D A0      DB    0A0H
2182: 0B3E E0      DB    0E0H
2182: 0B3F 20      DB    020H
2182: 0B40 E0      DB    0E0H
2182: 0B41          ENDM
2182:
2183: 0B41          DEF5 000H, 040H, 000H, 040H, 000H ; :
2183: 0B41 00      DB    000H
2183: 0B42 40      DB    040H
2183: 0B43 00      DB    000H
2183: 0B44 40      DB    040H
2183: 0B45 00      DB    000H
2183: 0B46          ENDM
2183:
2184: 0B46          DEF5 040H, 0E0H, 0E0H, 0E0H, 0E0H ; BULLET
2184: 0B46 40      DB    040H
2184: 0B47 E0      DB    0E0H
2184: 0B48 E0      DB    0E0H
2184: 0B49 E0      DB    0E0H
2184: 0B4A E0      DB    0E0H
2184: 0B4B          ENDM
2184:
2185:                ;
2186:                ; MOVE ROUTINE
2187: 0B4B EDB0      MMOVE  LDIR
2188: 0B4D C9        RET
2189:                ;
2190:                ; SYSTEM ENTRY POINT FOR NONMAGIC ADDRESS
2191: 0B4E E5        RELTA1 PUSH HL
2192: 0B4F E6FC      AND    0FCH          ; TOSS OUT SHIFT AMOUNT
2193: 0B51 6F        LD     L,A          ; SAVE
2194: 0B52 7B        LD     A,E          ; GET X
2195: 0B53 E603      AND    03H          ; ISOLATE SHIFT AMOUNT
2196: 0B55 B5        OR     L          ; COMBINE WITH MR
2197: 0B56 F5        RELTA2 PUSH AF
2198: 0B57 E640      AND    040H         ; IS FLOPPED BIT SET?
2199: 0B59 7B        LD     A,E
2200: 0B5A 2803      JR     Z,$+(RELTA3-$) ; JUMP IF NOT
2201: 0B5C 2F        CPL          ; YEP - UNFLOP THE COORDINATE
2202: 0B5D C6A0      ADD    A,160
2203: 0B5F 6A        RELTA3 LD     L,D          ; HL = Y
2204: 0B60 2600      LD     H,0
2205: 0B62 29        ADD    HL,HL        ; SET HL = Y * 8
2206: 0B63 29        ADD    HL,HL
2207: 0B64 29        ADD    HL,HL
2208: 0B65 54        LD     D,H
2209: 0B66 5D        LD     E,L

```

Home Video Game System Library - System Subroutines

```

2210: 0B67 29          ADD HL,HL          ; SET HL = Y * 32
2211: 0B68 29          ADD HL,HL
2212: 0B69 19          ADD HL,DE          ; SET HL = Y * 40
2213: 0B6A CB3F        SRL A              ; A = X 4
2214: 0B6C CB3F        SRL A
2215: 0B6E 5F          LD E,A
2216: 0B6F 1600        LD D,0
2217: 0B71 19          ADD HL,DE          ; HL = Y * 40 + X      4
2218: 0000             IF NWDWR-1
2219:                ENDIF
2220: 0B72 EB          EX DE,HL
2221:                ;
2222:                ; NAME:          RETURN FROM MACRO SUBROUTINE
2223:                ; PURPOSE:       RETURN CONTROL TO CALLER
2224:                ; THIS CODE WAS 'STOLEN' FROM RELABS SINCE
2225:                ; IT DOES THE STACK CLEANUP THAT MRET DOES
2226: 0B73 F1          MMRET POP AF
2227: 0B74 E1          POP HL
2228: 0B75 C9          RET
2229:                ;
2230:                ; ENTRY FOR USER
2231: 0B76 CD7B0B      INXNIB CALL XNIB
2232: 0B79 188B          JR $(MFROG-$)
2233:                ;
2234:                ; NAME:          INDEX NIBBLE
2235:                ; PURPOSE:       LOAD OF SPECIFIED NIBBLE RELATIVE TO BASE
2236:                ;                C = NIBBLE NUMBER
2237:                ;                HL = BASE ADDRESS
2238:                ; OUTPUT:        NIBBLE RETURNED RIGHT JUSTIFIED IN A
2239:                ; DESCRIPTION:    BYTE = NIBBLE#
2240:                ; THE LOW ORDER NIBBLE OF A GIVEN BYTE IS ADDRESSED
2241:                ; BY AN EVEN NIBBLE NUMBER.
2242: 0B7B E5          XNIB PUSH HL
2243: 0B7C C5          PUSH BC
2244: 0B7D 0600        LD B,0
2245: 0B7F CB39        SRL C
2246: 0B81 09          ADD HL,BC
2247: 0B82 7E          LD A,(HL)
2248: 0B83 C1          POP BC
2249: 0B84 CB41        BIT 0,C
2250: 0B86 2804        JR Z,$+(XNIB1-$)
2251: 0B88 0F          RRCA
2252: 0B89 0F          RRCA
2253: 0B8A 0F          RRCA
2254: 0B8B 0F          RRCA
2255: 0B8C E60F        XNIB1 AND 0FH
2256: 0B8E E1          POP HL
2257: 0B8F C9          RET
2258:                ;
2259:                ; NAME:          STORE NIBBLE
2260:                ; PURPOSE:       NIBBLE STORING (!)
2261:                ; INPUT:         A = NIBBLE TO STORE
2262:                ;                C = NIBBLE NUMBER (AS IN XNIB)
2263:                ;                HL = BASE ADDRESS
2264: 0B90 E5          PUTNIB PUSH HL
2265: 0B91 C5          PUSH BC
2266: 0B92 0600        LD B,0

```

Home Video Game System Library - System Subroutines

```

2267: 0B94 CB39          SRL  C
2268: 0B96 09           ADD  HL,BC
2269: 0B97 C1            POP  BC
2270: 0B98 CB41          BIT  0,C
2271: 0B9A 2809          JR   Z,$+(PUTNB1-$)
2272:                ; L.O. CASE - SHIFT IT
2273: 0B9C 07            RLCA
2274: 0B9D 07            RLCA
2275: 0B9E 07            RLCA
2276: 0B9F 07            RLCA
2277: 0BA0 AE            XOR  (HL)
2278: 0BA1 E6F0          AND  0F0H
2279: 0BA3 1803          JR   $+(PUTNB2-$)
2280: 0BA5 AE            PUTNB1 XOR  (HL)          ; L.O. CASE
2281: 0BA6 E60F          AND  0FH
2282: 0BA8 AE            PUTNB2 XOR  (HL)
2283: 0BA9 77           LD   (HL),A
2284: 0BAA E1           POP  HL
2285: 0BAB C9           RET
2286:                ;
2287:                ; NAME:          INDEX WORD TABLE (WORD INDEX)
2288:                ; PURPOSE:      TO INDEX AN ARRAY OF DEFW'S
2289:                ; INPUTS:       A = INDEX NUMBER (0-255)
2290:                ;                HL -> TABLE ENTRY 0
2291:                ; OUTPUTS:      DE = ENTRY LOOKED UP
2292:                ;                HL = POINTER TO ENTRY IN TABLE
2293: 0BAC 5F           MINDW LD   E,A
2294: 0BAD 1600          LD   D,0
2295: 0BAF CB23          SLA  E
2296: 0BB1 CB12          RL   D                ; DE*2
2297: 0BB3 19           ADD  HL,DE
2298: 0BB4 5E           LD   E,(HL)
2299: 0BB5 23           INC  HL
2300: 0BB6 56           LD   D,(HL)
2301: 0BB7 2B           DEC  HL
2302: 0BB8 CDF40C        STHLDE CALL FINDL3
2303: 0BBB 1808          JR   $+(MINDB1-$) ; JOIN STORE IN INDEX BYTE
2304:                ;
2305:                ; NAME:          INDEX BYTE TABLE
2306:                ; PURPOSE:      TABLE LOOKUP
2307:                ; INPUTS:       A = INDEX NUMBER
2308:                ; OUTPUT:      A = VALUE OF BYTE
2309:                ;                HL = POINTER TO TABLE ENTRY
2310: 0BBD 5F           MINDB LD   E,A
2311: 0BBE 1600          LD   D,0
2312: 0BC0 19           ADD  HL,DE
2313: 0BC1 7E           LD   A,(HL)
2314: 0BC2 FD7709        LD   (IY+CBA),A
2315: 0BC5 FD740B        MINDB1 LD   (IY+CBH),H
2316: 0BC8 FD750A        LD   (IY+CBL),L
2317: 0BCB C9           RET
2318:                ;
2319:                ; NAME:      DISPLAY TIME
2320:                ; PURPOSE:  DISPLAY TIME ON SCREEN
2321:                ; INPUTS:   E = X COORD
2322:                ;                D = Y COORD
2323:                ;                C = SAME AS DISCHR OPTIOS EXCEPT BIT 7 = 1

```

Home Video Game System Library - System Subroutines

```

2324:          ;          TO DISPLAY COLON AND SECONDS
2325:          ; OUTPUTS: NONE
2326: 0BCC          MDISTI
2327: 0BCC DD210D02      LD IX,SMLFNT
2328: 0BD0 0642          LD B,42H
2329: 0BD2 21EE4F          LD HL,GTMIN5
2330: 0BD5 C5              PUSH BC
2331: 0BD6 FDCB06BE        RES 7,(IY+CBC)
2332: 0BDA CDEB0B          CALL BCDISP
2333: 0BDD C1              POP BC
2334: 0BDE CB79           BIT 7,C
2335: 0BE0 C8              RET Z
2336: 0BE1 3EBA           LD A,80H+3AH
2337: 0BE3 CDE107          CALL DISPCH
2338: 0BE6 0642          LD B,42H
2339: 0BE8 21ED4F          LD HL,GTSECS
2340:          ; AND FALL INTO...
2341:          ;
2342:          ; NAME:          DISPLAY BCD NUMBER
2343:          ; INPUT:          B = NUMBER DISPLAY OPTIONS
2344:          ;                  C = CHARACTER DISPLAY OPTIONS
2345:          ;                  DE = Y,X COORDINATES
2346:          ;                  HL = NUMBER ADDRESS (POINTS AT LO BYTE)
2347:          ;                  IX = ALTERNATE FONT (IF USED)
2348:          ; OUTPUT:          DE UPDATED
2349:          ; DESCRIPTION: THIS ROUTINE CONVERTS EACH NIBBLE INTO
2350:          ; ASCII AND DISPLAYS IT. THE NORMALLY ILLEGAL BCD
2351:          ; VALUES ARE DISPLAYED AS CODES 2A THRU 2F RESPECTIVELY.
2352:          ; THE NUMBER DISPLAY OPTIONS BYTE IS FORMATED AS FOLLOWS:
2353:          ; BIT 7          SET IF LEADING ZERO SUPPRESSION WANTED
2354:          ; BIT 6          SET IF USE OF ALTERNATE FONT WANTED
2355:          ; BITS 5-0      NUMBER OF DIGITS TO DISPLAY (NOT NUMBER 0)
2356: 0BEB 78              BCDISP LD A,B          ; GET OPTIONS BYTE
2357: 0BEC E63F           AND 3FH          ; ISOLATE NUMBER OF DIGITS
2358: 0BEE 3D              BCDD0 DEC A
2359: 0BEF F8              RET M          ; QUIT IF NULL OR NO MORE
2360: 0BF0 4F              LD C,A          ; SAVE
2361: 0BF1 CD7B0B          CALL XNIB        ; GET NEXT DIGIT
2362: 0BF4 2007           JR NZ,$+(BCDD1-$) ; JUMP IF NONZERO
2363: 0BF6 CB78           BIT 7,B          ; IS ZERO SUPPRESS ON?
2364: 0BF8 2803           JR Z,$+(BCDD1-$) ; JUMP IF NOT
2365: 0BFA B1              OR C            ; LAST DIGIT?
2366: 0BFB 2014           JR NZ,$+(BCDD4-$)
2367: 0BFD CBB8           BCDD1 RES 7,B          ; CLEAR LEADING ZERO FLAG
2368: 0BFF C606           ADD A,6
2369: 0C01 E60F           AND 0FH
2370: 0C03 C62A           ADD A,2AH
2371: 0C05 CB70           BCDD2 BIT 6,B          ; ALTERNATE FONT?
2372: 0C07 2802           JR Z,$+(BCDD3-$) ; JUMP IF SO
2373: 0C09 F680           OR 80H          ; YEA, SET THE BIT
2374: 0C0B CDE107          BCDD3 CALL DISPCH        ; DISPLAY THE CHAR
2375: 0C0E 79              LD A,C          ; GET LOOP COUNTER IN A
2376: 0C0F 18DD           JR $+(BCDD0-$) ; AND GO FOR NEXT
2377: 0C11 3E20           BCDD4 LD A,' '        ; LEADING ZERO - WRITE A SPACE
2378: 0C13 18F0           JR $+(BCDD2-$)
2379:          ;
2380:          ; NAME:          INCREMENT SCORE

```

Home Video Game System Library - System Subroutines

```

2381:                ; PURPOSE: INCREMENT SCORE AND COMPARE TO END SCORE
2382:                ; INPUTS:  HL -> PLAYER SCORE LOW ADDR OF 3 BYTES
2383:                ; OUTPUTS: GSBEND OF GAMSTB SET IF MAX SCORE REACHED
2384: 0C15 0603      MINCSC LD  B,3
2385: 0C17 E5        PUSH HL
2386: 0C18 7E        INCLOP LD  A,(HL)
2387: 0C19 C601      ADD  A,1
2388: 0C1B 27        DAA
2389: 0C1C 77        LD   (HL),A
2390: 0C1D 2003     JR   NZ,$+(CMPIT-$)
2391: 0C1F 23        INC  HL
2392: 0C20 10F6     DJNZ $+(INCLOP-$)
2393: 0C22 E1        CMPIT POP HL
2394: 0C23 23        INC  HL
2395: 0C24 23        INC  HL
2396: 0C25 3AF84F    LD   A,(GAMSTB)
2397: 0C28 CB4F      BIT  GSBSCR,A
2398: 0C2A C8        RET  Z
2399: 0C2B 11F64F    LD  DE,ENDSCR+2
2400: 0C2E 0603      LD  B,3
2401: 0C30 1A        CMPLOP LD A,(DE)
2402: 0C31 BE        CP   (HL)
2403: 0C32 2807     JR   Z,$+(REPEAT-$) ; ENDSCLR = SCORE
2404: 0C34 D0        RET  NC ; ENDSCLR > SCORE
2405: 0C35 21F84F    SETEND LD HL,GAMSTB ; ENDSCLR < SCORE
2406: 0C38 CBF8     SET  GSBEND,(HL)
2407: 0C3A C9        RET
2408: 0C3B 1B        REPEAT DEC DE
2409: 0C3C 2B        DEC  HL
2410: 0C3D 10F1     DJNZ $+(CMPLOP-$)
2411: 0C3F 18F4     JR   $+(SETEND-$)
2412:                ;
2413:                ; NAME:          QUIT
2414:                ; PURPOSE:      HOLD PRESENT GAME SCORE UNTIL KEY HIT OR
2415:                ; SAY GAME OVER
2416: 0C41          MQUIT SYSSUK STRDIS
2416: 0C41 FF      RST  56
2416: 0C42 35      DB   STRDIS+1
2416: 0000          IF   STRDIS = INTPC
2416:                ENDIF
2416: 0C43          ENDM
2416:                ;
2417: 0C43 30      DB   48
2418: 0C44 18      DB   24
2419: 0C45 4C      DB   01001100B
2420: 0C46 570C    DW   GMOV
2421: 0C48          SYSTEM ACTINT ; ACTIVATE INTERRUPTS
2421: 0C48 FF      RST  56
2421: 0C49 0E      DB   ACTINT
2421: 0000          IF   ACTINT = INTPC
2421:                ENDIF
2421: 0C4A          ENDM
2421:                ;
2422: 0C4A          MQUIT1 SYSSUK SENTRY ; WAIT FOR SOMETHING TO HAPPEN
2422: 0C4A FF      RST  56
2422: 0C4B 43      DB   SENTRY+1
2422: 0000          IF   SENTRY = INTPC

```

Home Video Game System Library - System Subroutines

```

2422:                                ENDIF
2422: 0C4C                                ENDM
2422:
2423: 0C4C 1402                            DW  AKEYS
2424: 0C4E FE14                            CP  ST0
2425: 0C50 2804                            JR  Z,$+(MQUIT2-$) ; TRIGGER CHANGE?
2426: 0C52 FE13                            CP  SKYD ; KEY HIT?
2427: 0C54 20F4                            JR  NZ,$+(MQUIT1-$) ; NO - KEEP GOING
2428: 0C56 C7                                MQUIT2 RST 0 ; YES, RESET
2429: 0C57 47414D45                        GMOVR DB 'G','A','M','E'
2430: 0C5B 06                                DB 6
2431: 0C5C 4F564552                        DB 'O','V','E','R'
2432: 0C60 00                                DB 0
2433: ;
2434: ; *****
2435: ; * MENU ROUTINES *
2436: ; *****
2437: 0060                                NOLINE EQU 96 ; NUMBER OF DISPLAYED LINES
2438: 0000                                MNNL EQU 0 ; NEXT FIELD
2439: 0001                                MNNH EQU 1
2440: 0002                                MNSAL EQU 2 ; STRING ADDRESS
2441: 0003                                MNSAH EQU 3
2442: 0004                                MNGL EQU 4 ; GO TO ADDRESS
2443: 0005                                MNGH EQU 5
2444: ;
2445: ; SYSTEM POWER UP ROUTINE
2446: 0C61 3A0020                            PWRUP LD A,(FIRSTC) ; GET FIRST CASSETTE LOCATION
2447: 0C64 FEC3                            CP 0C3H ; IS IT A JUMP?
2448: 0C66 CA0020                            JP Z,FIRSTC ; JUMP TO IT IF SO
2449: 0C69 31CE4F                            LD SP,BEGRAM
2450: 0C6C                                SYSSUK FILL ; CLEAR SYSTEM RAM
2450: 0C6C FF                                RST 56
2450: 0C6D 1B                                DB FILL+1
2450: 0000                                IF FILL = INTPC
2450:                                ENDIF
2450: 0C6E                                ENDM
2450:
2451: 0C6E CE4F                                DW BEGRAM
2452: 0C70 3200                                DW 50
2453: 0C72 00                                DB 0
2454: 0C73 32FF0F                            LD (WASTE),A ; CLEAR SHIFTER
2455: 0C76 3D                                DEC A
2456: 0C77 32EC4F                            LD (TIMEOUT),A ; CLEAR TIMEOUT WATCHDOG
2457: 0C7A                                SYSTEM INTPC
2457: 0C7A FF                                RST 56
2457: 0C7B 00                                DB INTPC
2457: 0001                                IF INTPC = INTPC
2457: 0001                                INTPCC DEFL 1
2457:                                ENDIF
2457: 0C7C                                ENDM
2457:
2458: 0C7C                                DO EMUSIC
2458: 0C7C 15                                DB EMUSIC+1
2458: 0C7D                                ENDM
2458:
2459: 0C7D                                DO SETOUT
2459: 0C7D 17                                DB SETOUT+1

```

Home Video Game System Library - System Subroutines

```

2459: 0C7E          ENDM
2459:
2460: 0C7E BF          DB  (NOLINE*2)-1
2461: 0C7F 29          DB  41
2462: 0C80 08          DB  8
2463: 0C81          DO  COLSET
2463: 0C81 19          DB  COLSET+1
2463: 0C82          ENDM
2463:
2464: 0C82 1300         DW  MENUCL
2465: 0C84          DO  ACTINT
2465: 0C84 0F          DB  ACTINT+1
2465: 0C85          ENDM
2465:
2466: 0C85          EXIT
2466: 0C85 02          DB  XINTC
2466: 0000          INTPCC DEF  0
2466: 0C86          ENDM
2466:
2467: 0C86 11F30D       LD  DE,GAMSTR    ; 'SELECT GAME' AS TITLE
2468: 0C89 210020       LD  HL,FIRSTC   ; ASSUME MENU STARTS IN CASSETT
2469: 0C8C 7E          LD  A,(HL)      ; GET FIRST CASSETTE BYTE
2470: 0C8D 23          INC  HL
2471: 0C8E FE55        CP  55H         ; IS SENTINEL THERE?
2472: 0C90 2803        JR  Z,$+(PWRUP1-$) ; YEP - JUMP
2473: 0C92 211802       LD  HL,GUNLNK   ; WRONG - USE ONBOARD ONLY
2474: 0C95          PWRUP1 SYSTEM MENU ; DISPLAY THE MENU
2474: 0C95 FF          RST  56
2474: 0C96 4A          DB  MENU
2474: 0000          IF  MENU = INTPC
2474:          ENDF
2474: 0C97          ENDM
2474:
2475:          ;
2476:          ; NAME:          DISPLAY MENU AND BRANCH ON CHOICE
2477:          ; INPUT:         HL = MENU LIST
2478:          ;              DE = MENU TITLE
2479:          ; OUTPUT:        DE = TITLE OF SELECTION MADE
2480:          ; DESCRIPTION:
2481:          ;           THE MENU LIST IS A LINKED LIST OF THE FOLLOWING F
2482:          ; *****
2483:          ; * 0 * NEXT ENTRY          *
2484:          ; * 1 *                      *
2485:          ; *****
2486:          ; * 2 * STRING ADDRESS      *
2487:          ; * 3 *                      *
2488:          ; *****
2489:          ; * 4 * BRANCH TO ADDRESS *
2490:          ; * 5 *                      *
2491:          ; *****
2492:          ; THIS LIST IS TERMINATED BY A NEXT ENTRY FIELD OF ZEROS
2493:          ; A MAXIMUM OF EIGHT ENTRYS MAY BE DISPLAYED.
2494: 0C97 E5          MMENU  PUSH HL
2495: 0C98 E5          PUSH HL
2496: 0C99 CD190D       CALL MNCLR      ; CLEAR SCREEN AND THROWUP TITLE
2497: 0C9C          XYRELL DE, 16, 12
2497: 0C9C 11100C       LD  DE,[(12 SHL 8)+16]

```

Home Video Game System Library - System Subroutines

```

2497: 0C9F          ENDM
2497:
2498: 0C9F 010901      LD  BC, 109H      ; INITIALIZE ENTRY # AND COLOR
2499: 0CA2 DDE1          MMENU1 POP  IX      ; FIRST ENTRY TO IX
2500: 0CA4 78          LD  A,B          ; SELECTION NUMBER TO A
2501: 0CA5 C630          ADD  A,'0'        ; MAKE IT ASCII
2502: 0CA7          SYSTEM CHRDIS    ; AND SHOW IT
2502: 0CA7 FF          RST  56
2502: 0CA8 32          DB   CHRDIS
2502: 0000          IF   CHRDIS = INTPC
2502:          ENDIF
2502: 0CA9          ENDM
2502:
2503: 0CA9 3E2D          LD  A,'-'        ; DISPLAY DASH
2504: 0CAB          SYSTEM CHRDIS
2504: 0CAB FF          RST  56
2504: 0CAC 32          DB   CHRDIS
2504: 0000          IF   CHRDIS = INTPC
2504:          ENDIF
2504: 0CAD          ENDM
2504:
2505: 0CAD DD6603      LD  H,(IX+MNSAH) ; HL = STRING ADDRESS
2506: 0CB0 DD6E02      LD  L,(IX+MNSAL)
2507: 0CB3          SYSTEM STRDIS    ; DISPLAY SELECTION
2507: 0CB3 FF          RST  56
2507: 0CB4 34          DB   STRDIS
2507: 0000          IF   STRDIS = INTPC
2507:          ENDIF
2507: 0CB5          ENDM
2507:
2508: 0CB5 3E08          LD  A,8
2509: 0CB7 82          ADD  A,D          ; TO NEXT LINE
2510: 0CB8 57          LD  D,A
2511: 0CB9 1E10          LD  E,16
2512: 0CBB 04          INC  B            ; BUMP ENTRY #
2513: 0CBC DD6601      LD  H,(IX+MNNH)  ; HL = NEXT ENTRY ADDR
2514: 0CBF DD6E00      LD  L,(IX+MNNL)
2515: 0CC2 E5          PUSH HL
2516: 0CC3 7C          LD  A,H
2517: 0CC4 B5          OR  L
2518: 0CC5 20DB          JR  NZ,$+(MMENU1-$) ; NO - JUMP BACK
2519:          ; AT THIS POINT HL = 0 (SP) = 0
2520: 0CC7 39          ADD  HL,SP        ; HL = STACK POINTER
2521: 0CC8 C5          MMENU3 PUSH BC
2522: 0CC9 010101      LD  BC,0101H
2523: 0CCC          XYRELL DE, 16, 77 ; FEEDBACK ADDRESS
2523: 0CCC 11104D      LD  DE,[(77 SHL 8)+16]
2523: 0CCF          ENDM
2523:
2524: 0CCF          SYSTEM GETNUM    ; GET NUMBA
2524: 0CCF FF          RST  56
2524: 0CD0 4E          DB   GETNUM
2524: 0000          IF   GETNUM = INTPC
2524:          ENDIF
2524: 0CD1          ENDM
2524:
2525: 0CD1 C1          POP  BC

```

Home Video Game System Library - System Subroutines

```

2526: 0CD2 7E          LD  A,(HL)          ; HOW DOES SHE LOOK?
2527: 0CD3 A7          AND  A              ; ZERO ENTERED?
2528: 0CD4 2803       JR  Z,$+(MMENU5-$) ; JUMP IF SO
2529: 0CD6 B8          CP  B              ; IN RANGE?
2530: 0CD7 3806       JR  C,$+(MMENU6-$) ; JUMP IF SO
2531: 0CD9 3E3F       MMENU5 LD  A,'?'         ; DUD ENTRY - SHOW ?
2532: 0CDB          SYSTEM CHRDIS
2532: 0CDB FF          RST  56
2532: 0CDC 32          DB  CHRDIS
2532: 0000          IF  CHRDIS = INTPC
2532:          ENDIF
2532: 0CDD          ENDM
2532:
2533: 0CDD 18E9       JR  $+(MMENU3-$) ; GO BACK FOR NEXT TRY
2534: 0CDF E1          MMENU6 POP HL           ; THROW OUT ENTRY AREA
2535: 0CE0 D1          POP DE           ; RESTORE HEAD OF MENU LIST
2536: 0CE1 47          LD  B,A          ; NUMBER ENTERED TO B
2537: 0CE2 EB          MMENU7 EX  DE,HL   ; HL - ENTRY POINTER
2538: 0CE3 5E          LD  E,(HL)       ; DE - NEXT
2539: 0CE4 23          INC  HL
2540: 0CE5 56          LD  D,(HL)
2541: 0CE6 10FA       DJNZ $+(MMENU7-$) ; COUNT DOWN TO ENTRY
2542: 0CE8 23          INC  HL
2543: 0CE9 5E          LD  E,(HL)       ; STRING TO DE
2544: 0CEA 23          INC  HL
2545: 0CEB 56          LD  D,(HL)
2546: 0CEC 23          INC  HL
2547: 0CED 4E          LD  C,(HL)       ; GO TO ADDRESS TO BC
2548: 0CEE 23          INC  HL
2549: 0CEF 46          LD  B,(HL)
2550: 0CF0 E1          POP HL           ; HL = RETURN TO PLACE
2551: 0CF1 F1          POP AF           ; THROW OUT OLD PC
2552: 0CF2 C5          PUSH BC          ; PUT NEW PC ON STACK
2553: 0CF3 E5          PUSH HL          ; AND PUT BACK DUMMY RETURN
2554: 0CF4 FD7304     FINDL3 LD  (IY+CBE),E   ; PASS BACK TITLE ADDRESS
2555: 0CF7 FD7205     LD  (IY+CBD),D
2556: 0CFA C9          RET              ; AND GO BACK
2557:          ;
2558:          ; NAME:          GET PARAMETER
2559:          ; PURPOSE:       INPUT OF PROGRAM OPTIONS
2560:          ; INPUT:        A = NUMBER OF DIGITS
2561:          ;              BC = PROMPT STRING ADDRESS
2562:          ;              DE = FRAME TITLE ADDRESS
2563:          ;              HL = PARAMETER ADDRESS
2564:          ; DESCRIPTION:
2565:          ;          THIS ROUTINE ASKS THE USER TO ENTER A NUMBER
2566:          ;          FIRST A MENU FRAME IS CREATED, USING THE STRING
2567:          ;          POINTED AT BY DE AS A TITLE.  THE STRING 'ENTER'
2568:          ;          IS DISPLAYED, FOLLOWED BY THE PROMPT STRING.
2569:          ;          GETNUM IS THEN CALLED TO INPUT THE NUMBER.  FEEDBACK
2570:          ;          IS PROVIDED IN DOUBLE SIZED CHARACTERS.
2571:          ; NOTE: ** THIS ROUTINE USES TWO SYSTEM LEVELS AND THE AL
2572: 0CFB F5          MGETP  PUSH AF           ; SAVE NUMBER OF DIGITS
2573: 0CFC E5          PUSH HL
2574: 0CFD C5          PUSH BC
2575: 0CFE CD190D     CALL MNCLR
2576: 0D01          SYSSUK STRDIS     ; DISPLAY 'ENTER'

```

Home Video Game System Library - System Subroutines

```

2576: 0D01 FF          RST    56
2576: 0D02 35          DB     STRDIS+1
2576: 0000              IF     STRDIS = INTPC
2576:                ENDIF
2576: 0D03              ENDM
2576:
2577: 0D03 08          DB     8
2578: 0D04 20          DB     32
2579: 0D05 09          DB     1001B
2580: 0D06 B70D        DW     ENTSTG
2581: 0D08 E1          POP    HL
2582: 0D09              SYSTEM STRDIS      ; DISPLAY WHAT TO ENTER
2582: 0D09 FF          RST    56
2582: 0D0A 34          DB     STRDIS
2582: 0000              IF     STRDIS = INTPC
2582:                ENDIF
2582: 0D0B              ENDM
2582:
2583: 0D0B E1          POP    HL
2584: 0D0C F1          POP    AF
2585: 0D0D 47          LD     B,A
2586: 0D0E CBF1        SET    6,C      ; SET LARGE CHARS
2587: 0D10              XYRELL DE, 48, 48 ; LOAD FEEDBACK ADDRESS
2587: 0D10 113030      LD     DE,[(48 SHL 8)+48]
2587: 0D13              ENDM
2587:
2588: 0D13              SYSTEM GETNUM     ; GET NUMBER
2588: 0D13 FF          RST    56
2588: 0D14 4E          DB     GETNUM
2588: 0000              IF     GETNUM = INTPC
2588:                ENDIF
2588: 0D15              ENDM
2588:
2589: 0D15              SYSSUK PAWS      ; LET USER READ IT
2589: 0D15 FF          RST    56
2589: 0D16 51          DB     PAWS+1
2589: 0000              IF     PAWS = INTPC
2589:                ENDIF
2589: 0D17              ENDM
2589:
2590: 0D17 0F          DB     15
2591: 0D18 C9          RET
2592:                ; SUBROUTINE TO CLEAR SCREEN FOR MENU AND THROWUP TITLE
2593: 0D19 D5          MNCLR  PUSH DE
2594: 0D1A              SYSSUK FILL
2594: 0D1A FF          RST    56
2594: 0D1B 1B          DB     FILL+1
2594: 0000              IF     FILL = INTPC
2594:                ENDIF
2594: 0D1C              ENDM
2594:
2595: 0D1C 0040        DW     NORMEM
2596: 0D1E B801        DW     11*BYTEPL
2597: 0D20 00          DB     0
2598: 0D21              SYSSUK FILL
2598: 0D21 FF          RST    56
2598: 0D22 1B          DB     FILL+1

```

Home Video Game System Library - System Subroutines

```

2598: 0000          IF      FILL = INTPC
2598:                ENDIF
2598: 0D23          ENDM
2598:
2599: 0D23 B841          DW      NORMEM + (11*BYTEPL)
2600: 0D25 480D        DW      (NOLINE-11)*BYTEPL
2601: 0D27 55          DB      55H
2602: 0D28 E1            POP      HL
2603: 0D29          XYRELL DE, 24, 0 ; TITLE
2603: 0D29 111800      LD      DE, [(0 SHL 8)+24]
2603: 0D2C          ENDM
2603:
2604: 0D2C 0E04        LD      C, 0100B
2605: 0D2E          SYSTEM STRDIS
2605: 0D2E FF          RST      56
2605: 0D2F 34          DB      STRDIS
2605: 0000          IF      STRDIS = INTPC
2605:                ENDIF
2605: 0D30          ENDM
2605:
2606: 0D30 C9          RET
2607:                ;
2608:                ; NAME:      GET NUMBER
2609:                ; INPUT:    B = DISPLAY OPTIONS
2610:                ;          C = CHRDIS OPTIONS FOR FEEDBACK
2611:                ;          DE = COORDINATE OF FEEDBACK AREA
2612:                ;          HL = ADDRESS OF WHERE TO STASH NUMBER
2613:                ; DESCRIPTION: THIS ROUTINE CAN INPUT A NUMBER FROM
2614:                ;          EITHER THE KEYBOARD OR THE HAND CONTROL.  KEYBOARD
2615:                ;          ENTRY PROCEEDS CONVENTIONALLY.  GETNUM EXITS
2616:                ;          WHEN THE EQUALS KEY IS PRESSED OR THE REQUIRED NUM
2617:                ;          OF DIGITS IS ENTERED.
2618:                ;          PLAYER ONE HAND CONTROL MAY ALSO BE USED TO
2619:                ;          ENTER A NUMBER.  TO USE THIS OPTION, PULL THE TRIG
2620:                ;          THE ROTATE THE POT UNTIL THE NUMBER YOU WISH TO
2621:                ;          ENTER IS SHOWN IN THE FEEDBACK AREA.  PULL THE TRIG
2622:                ;          AGAIN TO REGISTER THE ENTRY.  IF DURING THE PROC
2623:                ;          THE KEYBOARD IS USED - KEYBOARD INPUT WILL OVERRIDE
2624: 0D31 D9          MGETN  EXX
2625: 0D32 CD990D      CALL  CLRNUM      ; CLEAR THE NUMBER
2626: 0D35 4F          LD      C,A        ; SET ZERO DIGITS IN - POT ENAB
2627: 0D36 FD7E07      MGETN1 LD  A, (IY+CBB) ; ENTRY COMPLETE?
2628: 0D39 A9          XOR      C
2629: 0D3A E63F        AND     3FH
2630: 0D3C C8          RET      Z        ; QUIT IF SO
2631: 0D3D 21360D      LD      HL, MGETN1
2632: 0D40 E5          PUSH   HL
2633: 0D41          SYSTEM RANGED      ; RANDOMIZE WHILE WE WAIT
2633: 0D41 FF          RST      56
2633: 0D42 76          DB      RANGED
2633: 0000          IF      RANGED = INTPC
2633:                ENDIF
2633: 0D43          ENDM
2633:
2634: 0D43          SYSSUK SENTRY
2634: 0D43 FF          RST      56
2634: 0D44 43          DB      SENTRY+1

```

Home Video Game System Library - System Subroutines

```

2634: 0000          IF      SENTRY = INTPC
2634:          ENDIF
2634: 0D45          ENDM
2634:
2635: 0D45 0B00      DW      NUMBAS
2636: 0D47          SYSSUK DOIT
2636: 0D47 FF        RST      56
2636: 0D48 45        DB      DOIT+1
2636: 0000          IF      DOIT = INTPC
2636:          ENDIF
2636: 0D49          ENDM
2636:
2637: 0D49 4C0D      DW      GNUMD0
2638: 0D4B C9        RET          ; NOTHIN - LOOP ON SENTRY
2639: 0D4C          GNUMD0 JMPd SKYD, MGETN6, 0H
2639: 0D4C 13        DB      SKYD
2639: 0D4D 7F0D      DW      MGETN6
2639: 0000          IF      0H = $C0
2639:          ENDIF
2639: 0D4F          ENDM
2639:
2640: 0D4F          JMPd ST0, MGETN2, 0H
2640: 0D4F 14        DB      ST0
2640: 0D50 550D      DW      MGETN2
2640: 0000          IF      0H = $C0
2640:          ENDIF
2640: 0D52          ENDM
2640:
2641: 0D52          JMPd SP0, MGETN3, 0H
2641: 0D52 1C        DB      SP0
2641: 0D53 610D      DW      MGETN3
2641: 0000          IF      0H = $C0
2641:          ENDIF
2641: 0D55          ENDM
2641:
2642:          ; TRIGGER ROUTINE
2643: 0D55 CB60      MGETN2 BIT 4,B          ; 0-1 TRANS?
2644: 0D57 C8        RET Z          ; NO -IGNORE
2645: 0D58 79        LD A,C
2646: 0D59 3C        INC A          ; ARE WE ALREADY IN POT MODE?
2647: 0D5A 283A      JR Z,$+(MGETN9-$) ; YEP - JUMP TO EXIT
2648: 0D5C CB79      BIT 7,C          ; POT LEGAL
2649: 0D5E C0        RET NZ          ; NO - IGNORE
2650: 0D5F 0EFF      LD C,0FFH       ; SET POT FLAG
2651:          ; POT ROUTINE
2652: 0D61 79        MGETN3 LD A,C          ; QUIT IF NOT IN POT MODE
2653: 0D62 3C        INC A
2654: 0D63 C0        RET NZ
2655:          ; HOW MANY DIGITS?
2656: 0D64 D9        EXX          ; TO NORMAL SET
2657: 0D65 78        LD A,B          ; SNATCH DIGITS
2658: 0D66 D9        EXX
2659: 0D67 FE01      CP 1          ; 1 PRAY TELL?
2660: 0D69 060A      LD B,10
2661: 0D6B 2802      JR Z,$+(MGETN4-$) ; JUMP IF GOOD GUESS
2662: 0D6D 0664      LD B,100       ; WRONG!
2663: 0D6F DB1C      MGETN4 IN A,(POT0) ; GET CURRENT POT VALUE

```

Home Video Game System Library - System Subroutines

```

2664: 0D71 57          LD    D,A          ; RANGE IT
2665: 0D72 AF          XOR   A
2666: 0D73 5F          LD    E,A
2667: 0D74 67          LD    H,A
2668: 0D75 19          MGETN5 ADD  HL,DE
2669: 0D76 CE00        ADC   A,0          ; ADD EVERY CARRY TO AC
2670: 0D78 27          DAA
2671: 0D79 10FA        DJNZ  $(MGETN5-$)
2672: 0D7B D9          EXX           ; BACK TO NORMAL SET
2673: 0D7C 77          LD    (HL),A
2674: 0D7D 1814        JR    $(MGETN8-$)
2675:                ; KEYBOARD ROUTINE
2676: 0D7F 0C          MGETN6 INC  C          ; POT MODE?
2677: 0D80 2004        JR    NZ,$+(MGETN7-$) ; JUMP IF NOT
2678: 0D82 CD990D        CALL CLRNUM
2679: 0D85 0C          INC  C          ; SET ONE DIGIT SO FAR
2680: 0D86 CBF9          MGETN7 SET  7,C      ; SET POT LOCKOUT
2681: 0D88            SYSTEM KCTASC
2681: 0D88 FF          RST   56
2681: 0D89 40          DB    KCTASC
2681: 0000            IF    KCTASC = INTPC
2681:                ENDIF
2681: 0D8A            ENDM
2681:
2682: 0D8A FE3D        CP    '='          ; EQUALS TYPED?
2683: 0D8C 2808        JR    Z,$+(MGETN9-$) ; QUIT IF EQUALS
2684: 0D8E E60F        AND  0FH
2685: 0D90 D9          EXX
2686: 0D91            SYSTEM SHIFTU      ; SHIFT DIGIT UP
2686: 0D91 FF          RST   56
2686: 0D92 60          DB    SHIFTU
2686: 0000            IF    SHIFTU = INTPC
2686:                ENDIF
2686: 0D93            ENDM
2686:
2687: 0D93 D5          MGETN8 PUSH DE
2688: 0D94            SYSTEM DISNUM
2688: 0D94 FF          RST   56
2688: 0D95 36          DB    DISNUM
2688: 0000            IF    DISNUM = INTPC
2688:                ENDIF
2688: 0D96            ENDM
2688:
2689:                ; ENTER HERE FOR EQUAL OR TRIGGER EXIT TO THROW OUT RETURN
2690: 0D96 D1          MGETN9 POP  DE
2691: 0D97 D9          EXX           ; BACK TO NORMAL
2692: 0D98 C9          RET
2693:                ;
2694:                ; SUBROUTINE TO CLEAR NUMBER
2695: 0D99 C5          CLRNUM PUSH BC
2696: 0D9A D9          EXX           ; TO NORMAL SET
2697: 0D9B E5          PUSH HL
2698: 0D9C 78          LD   A,B
2699: 0D9D 3C          INC  A
2700: 0D9E E63E        AND  3EH
2701: 0DA0 1F          RRA
2702: 0DA1 D9          EXX           ; BACK TO NORMAL SET

```

Home Video Game System Library - System Subroutines

```

2703: 0DA2 4F          LD    C,A
2704: 0DA3 AF          XOR   A
2705: 0DA4 47          LD    B,A
2706: 0DA5 D1          POP   DE
2707: 0DA6             SYSTEM FILL
2707: 0DA6 FF          RST   56
2707: 0DA7 1A          DB    FILL
2707: 0000             IF    FILL = INTPC
2707:             ENDIF
2707: 0DA8             ENDM
2707:
2708: 0DA8 C1          POP   BC
2709: 0DA9 C9          RET
2710:             ;
2711:             ; NAME:      SHIFT UP
2712:             ; INPUT:    A = DATA TO SHIFT UP
2713:             ;           B = SIZE IN DIGITS
2714:             ;           HL = AREA TO SHIFT ADDRESS
2715: 0DAA F5          MSHFTU PUSH AF
2716: 0DAB 78          LD    A,B
2717: 0DAC 3C          INC   A
2718: 0DAD E63E        AND   3EH
2719: 0DAF 47          LD    B,A
2720: 0DB0 F1          POP   AF
2721: 0DB1 ED6F        SHFTU1 RLD
2722: 0DB3 23          INC   HL
2723: 0DB4 10FB        DJNZ  $(SHFTU1-$)
2724: 0DB6 C9          RET
2725:             ;
2726: 0DB7 454E5445    ENTSTG DB  'E','N','T','E','R',' '
2726:             5220
2727: 0DBD 00          DB    0
2728: 0DBE FA01        CML   DW  CALCL
2729: 0DC0 D30D        DW  PNCM
2730: 0DC2 2813        DW  CMSTRT      ; CHECKMATE START
2731: 0DC4 0000        SCBL  DW  0
2732: 0DC6 E80D        DW  PNSCB
2733: 0DC8 190E        DW  SCBST
2734: 0DCA 47554E46    PNGF  DB  'G','U','N','F','I','G','H','T'
2734:             49474854
2735: 0DD2 00          DB    0
2736: 0DD3 43484543    PNCM  DB  'C','H','E','C','K','M','A','T','E'
2736:             4B4D4154
2736:             45
2737: 0DDC 00          DB    0
2738: 0DDD 43414C43    PNCALC DB  'C','A','L','C','U','L','A','T','O','R'
2738:             554C4154
2738:             4F52
2739: 0DE7 00          DB    0
2740: 0DE8 53435249    PNSCB DB  'S','C','R','I','B','B','L','I','N','G'
2740:             42424C49
2740:             4E47
2741: 0DF2 00          DB    0
2742: 0DF3 53454C45    GAMSTR DB  'S','E','L','E','C','T',' ','G','A','M','E'
2742:             43542047
2742:             414D45
2743: 0DFE 67          DB    67H

```

Home Video Game System Library - System Subroutines

```

2744: 0DFF 08          DB 8
2745: 0E00 58          DB 88
2746: 0E01 0D          DB 1101B
2747: 0E02 28432920      DB '(','C',')',' ','B','A','L','L','Y',' '
      42414C4C
      5920
2748: 0E0C 4D464720      DB 'M','F','G',' ','1','9','7','8'
      31393738
2749: 0E14 00          DB 0
2750:          ; FOUR FILLER BYTES BETWEEN ROM AND SCRIBBLING
2751:          ; TO MATCH 3159 ROM
2752: 0E15 FFFFFFFF      DB 0FFH, 0FFH, 0FFH, 0FFH

```

Home Video Game System Library - Scribbling

```

2753:          ;
2754:          ; *****
2755:          ; * SKETCH * Ver. 1.13 - Nov. 30, 2003
2756:          ; *****
2757:          ;
2758:          ; The official name of this
2759:          ; program is Scribbling
2760:          ;
2761:          ; zmac -i -m -o scribble.bin -x scribble.lst scribble.asm
2762:          ; Sketch equates
2763:          ; Sketch packet displacements
2764: 001E          SCPSIZ EQU 30          ; Size of sketch packet
2765: 0000          SCSAVA EQU 0          ; Save area start
2766: 001A          SCXC EQU 26          ; X coordinate
2767: 001B          SCYC EQU 27          ; Y coordinate
2768: 001C          SCSADL EQU 28          ; Save address Lo and Hi
2769: 001D          SCSADH EQU 29
2770:          ; Other goodies
2771: 0004          MOV TMR EQU 4          ; Move rate
2772: 0014          KSCTRV EQU 20          ; Color stepping time
2773: 0E19          ORG $E19          ; ** Start
2774: 0E19          BEGIN: SYSSUK GETPAR
2774: 0E19 FF          RST 56
2774: 0E1A 4D          DB GETPAR+1
2774: 0000          IF GETPAR = INTPC
2774:          ENDIF
2774: 0E1B          ENDM
2774:
2775: 0E1B 2802          DW NOPLAY
2776: 0E1D 01          DB 1
2777: 0E1E F34F          DW NUMPLY
2778: 0E20          SCCLR:
2779: 0E20 31E84E          LD SP,SCRSTK
2780: 0E23          SYSTEM INTPC
2780: 0E23 FF          RST 56
2780: 0E24 00          DB INTPC
2780: 0001          IF INTPC = INTPC
2780: 0001          INTPCC DEFL 1
2780:          ENDIF
2780: 0E25          ENDM
2780:
2781: 0E25          DO FILL
2781: 0E25 1B          DB FILL+1
2781: 0E26          ENDM
2781:
2782: 0E26 0040          DW NORMEM
2783: 0E28 600E          DW 92*BYTEPL
2784: 0E2A 00          DB 0
2785: 0E2B          DO FILL
2785: 0E2B 1B          DB FILL+1
2785: 0E2C          ENDM
2785:
2786: 0E2C F04E          DW P1SCP
2787: 0E2E 7800          DW SCPSIZ*4
2788: 0E30 00          DB 0
2789: 0E31          DO SETOUT
2789: 0E31 17          DB SETOUT+1

```

Home Video Game System Library - Scribbling

```

2789: 0E32          ENDM
2789:
2790: 0E32  B8          DB    184
2791: 0E33  28          DB    40
2792: 0E34  08          DB    8
2793: 0E35          DO    MOVE
2793: 0E35  5F          DB    MOVE+1
2793: 0E36          ENDM
2793:
2794: 0E36  E84E         DW    COLORS
2795: 0E38  0800         DW    8
2796: 0E3A  0C10         DW    INICOL
2797: 0E3C          DO    COLSET
2797: 0E3C  19          DB    COLSET+1
2797: 0E3D          ENDM
2797:
2798: 0E3D  E84E         DW    COLORS
2799: 0E3F          DO    SETW
2799: 0E3F  7D          DB    SETW+1
2799: 0E40          ENDM
2799:
2800: 0E40  46          DB    70
2801: 0E41  24          DB    36
2802: 0E42  0A4F         DW    P1SCP+SCXC
2803: 0E44          DO    SETW
2803: 0E44  7D          DB    SETW+1
2803: 0E45          ENDM
2803:
2804: 0E45  53          DB    83
2805: 0E46  24          DB    36
2806: 0E47  284F         DW    P2SCP+SCXC
2807: 0E49          DO    SETW
2807: 0E49  7D          DB    SETW+1
2807: 0E4A          ENDM
2807:
2808: 0E4A  46          DB    70
2809: 0E4B  30          DB    48
2810: 0E4C  464F         DW    P3SCP+SCXC
2811: 0E4E          DO    SETW
2811: 0E4E  7D          DB    SETW+1
2811: 0E4F          ENDM
2811:
2812: 0E4F  53          DB    83
2813: 0E50  30          DB    48
2814: 0E51  644F         DW    P4SCP+SCXC
2815: 0E53          DO    SETB
2815: 0E53  7B          DB    SETB+1
2815: 0E54          ENDM
2815:
2816: 0E54  04          DB    MOV TMR
2817: 0E55  D54F         DW    CT0
2818: 0E57          DONT  XINTC
2818: 0E57  02          DB    XINTC
2818: 0E58          ENDM
2818:
2819: 0E58  21580E      MAINLP: LD    HL,MAINLP
2820: 0E5B  E5          PUSH  HL

```

Home Video Game System Library - Scribbling

```

2821: 0E5C          SYSSUK SENTRY
2821: 0E5C FF        RST      56
2821: 0E5D 43        DB      SENTRY+1
2821: 0000          IF      SENTRY = INTPC
2821:          ENDIF
2821: 0E5E          ENDM
2821:
2822: 0E5E 650E        DW      KEYMES
2823: 0E60          SYSSUK DOIT
2823: 0E60 FF        RST      56
2823: 0E61 45        DB      DOIT+1
2823: 0000          IF      DOIT = INTPC
2823:          ENDIF
2823: 0E62          ENDM
2823:
2824: 0E62 A10E        DW      SC DOTB
2825: 0E64 C9          RET
2826: 0E65 2F        KEYMES: DB      $2F
2827: 0E66 0F        DB      $F
2828: 0E67 0F        DB      $F
2829: 0E68 0F        DB      $F
2830:          ; Keyboard handler
2831: 0E69 05        KEYB0: DEC     B
2832: 0E6A 0E03      LD      C,3
2833: 0E6C 78        LD      A,B
2834: 0E6D FE14      CP      20          ; Clear entry down?
2835: 0E6F 28AF      JR      Z,$+(SCCLR-$) ; Jump to clear if so
2836: 0E71 0F        RRCA
2837: 0E72 0F        RRCA
2838: 0E73 A1          AND     C
2839: 0E74          SYSSUK INDEXB
2839: 0E74 FF        RST      56
2839: 0E75 5D        DB      INDEXB+1
2839: 0000          IF      INDEXB = INTPC
2839:          ENDIF
2839: 0E76          ENDM
2839:
2840: 0E76 290F        DW      CDEL TB
2841: 0E78 EB          EX      DE,HL
2842: 0E79 78        LD      A,B
2843: 0E7A A1          AND     C
2844: 0E7B 67        LD      H,A
2845: 0E7C 79        LD      A,C
2846: 0E7D 94        SUB     H
2847: 0E7E          SYSSUK INDEXB          ; Point at color
2847: 0E7E FF        RST      56
2847: 0E7F 5D        DB      INDEXB+1
2847: 0000          IF      INDEXB = INTPC
2847:          ENDIF
2847: 0E80          ENDM
2847:
2848: 0E80 E84E        DW      COLORS
2849: 0E82 1A        LD      A,(DE)
2850: 0E83 86        ADD     A,(HL)          ; Add delta factor
2851: 0E84 CB58      BIT     3,B          ; Was key for intensity?
2852: 0E86 2804      JR      Z,$+(KEYB1-$)
2853: 0E88 AE          XOR     (HL)

```

Home Video Game System Library - Scribbling

```

2854: 0E89 E607          AND  7
2855: 0E8B AE             XOR  (HL)
2856: 0E8C 77          KEYB1: LD  (HL),A
2857: 0E8D 23          INC  HL           ; Change color on other side
2858: 0E8E 23          INC  HL
2859: 0E8F 23          INC  HL
2860: 0E90 23          INC  HL
2861: 0E91 77          LD  (HL),A
2862: 0E92          SYSSUK COLSET
2862: 0E92 FF          RST  56
2862: 0E93 19          DB   COLSET+1
2862: 0000          IF   COLSET = INTPC
2862:          ENDIF
2862: 0E94          ENDM
2862:
2863: 0E94 E84E          DW   COLORS
2864: 0E96 3E14        LD   A,KSCTRV     ; Set KEYSEX clear timer
2865: 0E98 32D64F      LD   (CT1),A
2866: 0E9B C9           RET
2867:          ; Routine to clear KEYSEX
2868: 0E9C AF          KLRKSX: XOR  A
2869: 0E9D 32E34F      LD   (KEYSEX),A
2870: 0EA0 C9           RET
2871: 0EA1          SCDOTB: JMPd SCT0,DOWRTS,0
2871: 0EA1 01          DB   SCT0
2871: 0EA2 D30F          DW   DOWRTS
2871: 0000          IF   0 = $C0
2871:          ENDIF
2871: 0EA4          ENDM
2871:
2872: 0EA4          JMPd SCT1,KLRKSX,0
2872: 0EA4 02          DB   SCT1
2872: 0EA5 9C0E          DW   KLRKSX
2872: 0000          IF   0 = $C0
2872:          ENDIF
2872: 0EA7          ENDM
2872:
2873: 0EA7          JMPd SKYD,KEYB0,0
2873: 0EA7 13          DB   SKYD
2873: 0EA8 690E          DW   KEYB0
2873: 0000          IF   0 = $C0
2873:          ENDIF
2873: 0EAA          ENDM
2873:
2874:          ; Iterate through active players subroutine
2875: 0EAA DD21F04E      ITER4: LD  IX,P1SCP
2876: 0EAE 3AF34F      LD  A,(NUMPLY)
2877: 0EB1 47          LD  B,A
2878: 0EB2 4F          LD  C,A
2879: 0EB3 C5          ITER41: PUSH BC
2880: 0EB4 E5          PUSH HL
2881: 0EB5 11BA0E      LD  DE,ITRET
2882: 0EB8 D5          PUSH DE
2883: 0EB9 E9          JP  (HL)
2884: 0EBA 111E00      ITRET: LD  DE,SCPSIZ
2885: 0EBD DD19          ADD  IX,DE
2886: 0EBF E1          POP  HL

```

Home Video Game System Library - Scribbling

```

2887: 0EC0 C1          POP BC
2888: 0EC1 10F0         DJNZ $+(ITER41-$)
2889: 0EC3 C9          RET
2890:                ; Update Coordinates Routine
2891: 0EC4 79          SCRUPD: LD A,C
2892: 0EC5 90          SUB B
2893: 0EC6             SYSSUK INDEXB
2893: 0EC6 FF          RST 56
2893: 0EC7 5D          DB INDEXB+1
2893: 0000             IF INDEXB = INTPC
2893:                ENDIF
2893: 0EC8             ENDM
2893:
2894: 0EC8 E44F         DW OSW0
2895: 0ECA E60F         AND $F
2896: 0ECC CD0110       CALL GETDLT ; Get deltas
2897: 0ECF DD7E1A       LD A,(IX+SCXC) ; Update X
2898: 0ED2 82          ADD A,D
2899: 0ED3 FE98         CP 152 ; Out of bounds?
2900: 0ED5 3003         JR NC,$+(SCRUP1-$)
2901: 0ED7 DD771A       LD (IX+SCXC),A
2902: 0EDA DD7E1B       SCRUP1: LD A,(IX+SCYC) ; Same for Y
2903: 0EDD 84          ADD A,H
2904: 0EDE FE55         CP 85
2905: 0EE0 D0          RET NC
2906: 0EE1 DD771B       LD (IX+SCYC),A
2907: 0EE4 C9          RET
2908:                ; Restore
2909: 0EE5 DDE5         SCREST: PUSH IX
2910: 0EE7 D1          POP DE
2911: 0EE8 1A          LD A,(DE)
2912: 0EE9 A7          AND A
2913: 0EEA C8          RET Z
2914: 0EEB DD661D       LD H,(IX+SCSADH)
2915: 0EEE DD6E1C       LD L,(IX+SCSADL)
2916: 0EF1             SYSTEM RESTOR
2916: 0EF1 FF          RST 56
2916: 0EF2 2E          DB RESTOR
2916: 0000             IF RESTOR = INTPC
2916:                ENDIF
2916: 0EF3             ENDM
2916:
2917: 0EF3 AF          XOR A
2918: 0EF4 12          LD (DE),A
2919: 0EF5 C9          RET
2920:                ; Write Routine
2921: 0EF6 79          SCRVRT: LD A,C
2922: 0EF7 90          SUB B
2923: 0EF8             SYSSUK INDEXB
2923: 0EF8 FF          RST 56
2923: 0EF9 5D          DB INDEXB+1
2923: 0000             IF INDEXB = INTPC
2923:                ENDIF
2923: 0EFA             ENDM
2923:
2924: 0EFA E44F         DW OSW0
2925: 0EFC E610         AND $10

```

Home Video Game System Library - Scribbling

```

2926: 0EFE C8          RET Z
2927: 0EFF 2B          SCRWR1: DEC HL          ; Backup to pot
2928: 0F00 2B          DEC HL
2929: 0F01 2B          DEC HL
2930: 0F02 2B          DEC HL
2931: 0F03 2B          DEC HL
2932: 0F04 7E          LD A,(HL)
2933: 0F05 07          RLCA
2934: 0F06 07          RLCA
2935: 0F07 4F          LD C,A
2936: 0F08 E603        AND 3
2937: 0F0A          SYSSUK INDEXB          ; Set sizes
2937: 0F0A FF          RST 56
2937: 0F0B 5D          DB INDEXB+1
2937: 0000          IF INDEXB = INTPC
2937:          ENDIF
2937: 0F0C          ENDM
2937:
2938: 0F0C 260F        DW SIZTBL
2939: 0F0E DD561B      LD D,(IX+SCYC)
2940: 0F11 DD5E1A      LD E,(IX+SCXC)
2941: 0F14 47          LD B,A
2942: 0F15 79          LD A,C
2943: 0F16 07          RLCA
2944: 0F17 07          RLCA
2945: 0F18 E603        AND 3
2946: 0F1A          SCRWR2: SYSSUK INDEXB
2946: 0F1A FF          RST 56
2946: 0F1B 5D          DB INDEXB+1
2946: 0000          IF INDEXB = INTPC
2946:          ENDIF
2946: 0F1C          ENDM
2946:
2947: 0F1C 220F        DW COLMSK
2948: 0F1E 48          LD C,B
2949: 0F1F          SYSTEM RECTAN
2949: 0F1F FF          RST 56
2949: 0F20 1C          DB RECTAN
2949: 0000          IF RECTAN = INTPC
2949:          ENDIF
2949: 0F21          ENDM
2949:
2950: 0F21 C9          RET
2951: 0F22 00          COLMSK: DB 0
2952: 0F23 55          DB 01010101B
2953: 0F24 AA          DB 10101010B
2954: 0F25 FF          DB 11111111B
2955: 0F26 01          SIZTBL: DB 1
2956: 0F27 02          DB 2
2957: 0F28 04          DB 4
2958: 0F29 08          CDELTB: DB 8
2959: 0F2A F8          DB -8
2960: 0F2B 01          DB 1
2961: 0F2C FF          DB -1
2962:          ; Save Routine
2963: 0F2D 78          SCRSAV: LD A,B
2964: 0F2E          SYSSUK INDEXB

```

Home Video Game System Library - Scribbling

```

2964: 0F2E FF RST 56
2964: 0F2F 5D DB INDEXB+1
2964: 0000 IF INDEXB = INTPC
2964: ENDIF
2964: 0F30 ENDM
2964:
2965: 0F30 E34F DW OSW0-1
2966: 0F32 E610 AND $10
2967: 0F34 C0 RET NZ
2968: 0F35 E5 PUSH HL
2969: 0F36 DD561B LD D,(IX+SCYC)
2970: 0F39 DD5E1A LD E,(IX+SCXC)
2971: 0F3C SYSTEM RELAB1
2971: 0F3C FF RST 56
2971: 0F3D 3A DB RELAB1
2971: 0000 IF RELAB1 = INTPC
2971: ENDIF
2971: 0F3E ENDM
2971:
2972: 0F3E DD721D LD (IX+SCSADH),D
2973: 0F41 DD731C LD (IX+SCSADL),E
2974: 0F44 EB EX DE,HL
2975: 0F45 DDE5 PUSH IX
2976: 0F47 D1 POP DE
2977: 0F48 010308 LD BC,$0803 ; Save worst case
2978: 0F4B SYSTEM SAVE
2978: 0F4B FF RST 56
2978: 0F4C 2C DB SAVE
2978: 0000 IF SAVE = INTPC
2978: ENDIF
2978: 0F4D ENDM
2978:
2979: 0F4D E1 POP HL
2980: 0F4E 18AF JR $+(SCRWR1-$)
2981: ; Zero Player Game Write Handler
2982: 0F50 21F04E ZEROPL: LD HL,ZPSTMR ; LOAD PTR TO SIZE TIMER
2983: 0F53 11F34E LD DE,ZPSIZ ; AND SIZE TRACKER
2984: 0F56 35 DEC (HL) ; DECREMENT SIZE TIMER
2985: 0F57 F2690F JP P,ZPA ; JUMP IF NO COUNTDOWN
2986: 0F5A SYSSUK RANGED
2986: 0F5A FF RST 56
2986: 0F5B 77 DB RANGED+1
2986: 0000 IF RANGED = INTPC
2986: ENDIF
2986: 0F5C ENDM
2986:
2987: 0F5C 30 DB 48
2988: 0F5D FE08 CP 8 ; 8-479 ???
2989: 0F5F 3802 JR C,$+(ZP0-$) ; NO - ZP0
2990: 0F61 E603 AND 3 ; YES - HAVE MORE 1-4S
2991: 0F63 3C ZP0: INC A
2992: 0F64 12 LD (DE),A ; GET NEW SIZE
2993: 0F65 SYSSUK RANGED ; GET NEW SIZE TIMER
2993: 0F65 FF RST 56
2993: 0F66 77 DB RANGED+1
2993: 0000 IF RANGED = INTPC
2993: ENDIF

```

Home Video Game System Library - Scribbling

```

2993: 0F67          ENDM
2993:
2994: 0F67  78          DB      120
2995: 0F68  77          LD      (HL),A
2996: 0F69  23          ZPA:   INC      HL          ; ADVANCE TO COLOR STUFF
2997: 0F6A  13          INC      DE
2998: 0F6B  35          DEC      (HL)          ; AND DEC COLOR TIMER
2999: 0F6C  F2770F       JP      P,ZPB
3000: 0F6F          SYSSUK RANGED          ; GET NEW COLOR
3000: 0F6F  FF          RST      56
3000: 0F70  77          DB      RANGED+1
3000: 0000          IF      RANGED = INTPC
3000:          ENDIF
3000: 0F71          ENDM
3000:
3001: 0F71  04          DB      4
3002: 0F72  12          LD      (DE),A
3003: 0F73          SYSSUK RANGED          ; GET NEW COLOR TIMER
3003: 0F73  FF          RST      56
3003: 0F74  77          DB      RANGED+1
3003: 0000          IF      RANGED = INTPC
3003:          ENDIF
3003: 0F75          ENDM
3003:
3004: 0F75  78          DB      120
3005: 0F76  77          LD      (HL),A
3006: 0F77  23          ZPB:   INC      HL          ; TO DIRECTION STUFF
3007: 0F78  13          INC      DE
3008: 0F79  35          DEC      (HL)          ; DECREMENT DIRECTION TIMER
3009: 0F7A  F2930F       JP      P,ZPD
3010: 0F7D  11F54E       ZPC:   LD      DE,DIRVAL ; DE = DIRECTION TRACKER
3011: 0F80          SYSSUK RANGED          ; DRAW NEW DIRECTION
3011: 0F80  FF          RST      56
3011: 0F81  77          DB      RANGED+1
3011: 0000          IF      RANGED = INTPC
3011:          ENDIF
3011: 0F82          ENDM
3011:
3012: 0F82  0A          DB      10
3013: 0F83  3C          INC      A
3014: 0F84  FE03         CP      3          ; REJECT ILLEGAL MOVES
3015: 0F86  28F5         JR      Z,$+(ZPC-$)
3016: 0F88  FE07         CP      7
3017: 0F8A  28F1         JR      Z,$+(ZPC-$)
3018: 0F8C  12          LD      (DE),A
3019: 0F8D          SYSSUK RANGED
3019: 0F8D  FF          RST      56
3019: 0F8E  77          DB      RANGED+1
3019: 0000          IF      RANGED = INTPC
3019:          ENDIF
3019: 0F8F          ENDM
3019:
3020: 0F8F  28          DB      40
3021: 0F90  32F24E       LD      (DIRTMR),A
3022: 0F93  1A          ZPD:   LD      A,(DE)          ; GET DIRECTION VALUE
3023: 0F94  CD0110       CALL   GETDLT          ; GET DELTAS
3024: 0F97  010A4F       LD      BC,P1SCP+SCXC ; POINT AT COORDINATE

```

Home Video Game System Library - Scribbling

```

3025: 0F9A 0A          LD  A,(BC)
3026: 0F9B 82          ADD A,D
3027: 0F9C FE50        CP   80
3028: 0F9E 30DD        JR   NC,$+(ZPC-$) ; GET NEW DIRECTION IF AT LMT
3029: 0FA0 02          LD  (BC),A
3030: 0FA1 5F          LD  E,A           ; SAVE X COORDINATE
3031: 0FA2 03          INC BC
3032: 0FA3 0A          LD  A,(BC)
3033: 0FA4 84          ADD A,H
3034: 0FA5 FE2E        CP   46
3035: 0FA7 30D4        JR   NC,$+(ZPC-$)
3036: 0FA9 02          LD  (BC),A
3037: 0FAA 57          LD  D,A           ; SET Y COORDINATE
3038: 0FAB 21F34E       LD  HL,ZPSIZ     ; POINT AT SIZES AGAIN
3039: 0FAE 46          LD  B,(HL)
3040: 0FAF 23          INC HL
3041: 0FB0 7E          LD  A,(HL)       ; GET COLOR TOO
3042: 0FB1 CD1A0F       CALL SCRWR2      ; DO FIRST WRITE
3043: 0FB4 67          LD  H,A           ; SAVE COLOR
3044: 0FB5 D5          PUSH DE          ; AND X,Y
3045: 0FB6 3E5C        LD  A,92         ; REFLECT Y
3046: 0FB8 90          SUB B
3047: 0FB9 92          SUB D
3048: 0FBA 57          LD  D,A
3049: 0FBB 7C          LD  A,H
3050: 0FBC          SYSTEM RECTAN
3050: 0FBC FF          RST  56
3050: 0FBD 1C          DB   RECTAN
3050: 0000          IF   RECTAN = INTPC
3050:          ENDIF
3050: 0FBE          ENDM
3050:
3051: 0FBE 3EA0        LD  A,160        ; REFLECT X
3052: 0FC0 91          SUB C
3053: 0FC1 93          SUB E
3054: 0FC2 5F          LD  E,A
3055: 0FC3 7C          LD  A,H
3056: 0FC4          SYSTEM RECTAN
3056: 0FC4 FF          RST  56
3056: 0FC5 1C          DB   RECTAN
3056: 0000          IF   RECTAN = INTPC
3056:          ENDIF
3056: 0FC6          ENDM
3056:
3057: 0FC6 E1          POP HL           ; RESTART X,Y
3058: 0FC7 54          LD  D,H           ; RESTORE Y
3059: 0FC8          SYSTEM RECTAN
3059: 0FC8 FF          RST  56
3059: 0FC9 1C          DB   RECTAN
3059: 0000          IF   RECTAN = INTPC
3059:          ENDIF
3059: 0FCA          ENDM
3059:
3060: 0FCA 3EFF        LD  A,$FF        ; RESET TIMEOUT
3061: 0FCC 32EC4F       LD  (TIMOUT),A
3062: 0FCF 3E01        ZERO1: LD  A,1           ; RESET COUNTER-TIMER
3063: 0FD1 182A        JR   $+(ZERO2-$)

```

Home Video Game System Library - Scribbling

```

3064: 0FD3 3AF34F      DOWRTS: LD  A,(NUMPLY)
3065: 0FD6 3D          DEC  A
3066: 0FD7 FE04          CP   4
3067: 0FD9 D2500F          JP   NC,ZEROPL
3068: 0FDC 21C40E          LD   HL,SCRUPD
3069: 0FDF CDAA0E          CALL ITER4
3070: 0FE2 21E50E          LD   HL,SCREST
3071: 0FE5 CDAA0E          CALL ITER4
3072: 0FE8 21F60E          LD   HL,SCRWRT
3073: 0FEB CDAA0E          CALL ITER4
3074:                ; NOW GOING BACKWARDS SAVE AND WRITE EVERYBODY WITH TRIGG
3075: 0FEE 41            LD   B,C
3076: 0FEF 11E2FF          SCR3:  LD  DE,-SCPSIZ
3077: 0FF2 DD19            ADD  IX,DE
3078: 0FF4 C5              PUSH BC
3079: 0FF5 CD2D0F          CALL SCRSAV
3080: 0FF8 C1              POP  BC
3081: 0FF9 10F4            DJNZ $+(SCR3-$)
3082: 0FFB 3E04            LD   A,MOVTMR
3083: 0FFD 32D54F          ZERO2: LD  (CT0),A
3084: 1000 C9              RET                                ; DONE
3085:                ; SUBROUTINE TO SCARE UP DELTAS
3086: 1001 C5              GETDLT: PUSH BC
3087: 1002 47              LD   B,A
3088: 1003                SYSSUK MSKTD
3088: 1003 FF              RST   56
3088: 1004 7F              DB   MSKTD+1
3088: 0000                IF   MSKTD = INTPC
3088:                ENDIF
3088: 1005                ENDM
3088:
3089: 1005 0001            DW   $100
3090: 1007 00              DB   0
3091: 1008 0001            DW   $100
3092: 100A C1              POP  BC
3093: 100B C9              RET
3094:                ; INITIAL COLORS
3095: 100C 08              INICOL: DB  $8
3096: 100D 62              DB  $62                ; Value changed from $5B to
match
3097: 100E A5              DB  $A5                ; exactly with "3159" ROM
3098: 100F 07              DB  $7
3099: 1010 08              DB  $8
3100: 1011 62              DB  $62                ; Value changed from $5B to
match
3101: 1012 A5              DB  $A5                ; exactly with "3159" ROM
3102: 1013 07              DB  $7
3103:                ;          ORG $4000+3720 - Note - DEFS statements removed
3104:                ; SWITCH RAM:          Original Source Syntax
3105:                ;          DEFS 96          -----
3106: 4EE8                SCRSTK EQU  $4EE8          ; SCRSTK:
3107: 4EE8                COLORS EQU  $4EE8          ; COLORS: DEFS 8
3108: 4EF0                ZPSTMR EQU  $4EF0          ; ZPSTMR EQU  $
3109: 4EF2                DIRTMR EQU  ZPSTMR+2        ; DIRTMR EQU  ZPSTMR+2
3110: 4EF3                ZPSIZ EQU  DIRTMR+1        ; ZPSIZ EQU  DIRTMR+1
3111: 4EF5                DIRVAL EQU  ZPSIZ+2        ; DIRVAL EQU  ZPSIZ+2
3112: 4EF0                P1SCP EQU  $4EF0          ; P1SCP:  DEFS SCPSIZ

```

Home Video Game System Library - Scribbling

```
3113: 4F0E          P2SCP EQU $4F0E          ; P2SCP: DEFS SCPSIZ
3114: 4F2C          P3SCP EQU $4F2C          ; P3SCP: DEFS SCPSIZ
3115: 4F4A          P4SCP EQU $4F4A          ; P4SCP: DEFS SCPSIZ
3116:                ;
3117:                ;          END OF SCRIBBLING PROGRAM
3118:                ;
3119:                ; TWELVE FILLER BYTES BETWEEN SCRIBBLING AND
3120:                ; CALCULATOR TO MATCH 3159 ROM
3121: 1014 FFFFFFFF          DB  0FFH, 0FFH, 0FFH, 0FFH, 0FFH
          FF
3122: 1019 FFFFFFFF          DB  0FFH, 0FFH, 0FFH, 0FFH, 0FFH
          FF
3123: 101E FFFF          DB  0FFH, 0FFH
```

This page intentionally left blank for double-sided printing purposes

Home Video Game System Library - Calculator

```

3124:          ;
3125:          ; *****
3126:          ; * CALCULATOR * (Dec 4, 2003)
3127:          ; *****
3128:          ;
3129:          ; CALCULATOR EQUATES
3130:          ; SCREEN PARAMETERS
3131: 005C          CALCLN EQU 92          ; LAST DISPLAY LINE USED
3132:          ; SCREEN MEMORY STACK PARAMETERS
3133: 005C          MAXSTK EQU CALCLN          ; MAX NUMBER OF STACK ENTRYS
3134: 4E58          STKBOT EQU MAXSTK*BYTEPL+4000H-8 ; ADDR OF STK BOTTOM
3135:          ; CALCULATOR MAJOR STATES
3136: 0000          ARG1MD EQU 0          ; ARG1 STATE
3137: 0001          ARG2MD EQU 1          ; ARG2 STATE
3138: 0002          SHOWMD EQU 2          ; SHOW STATE
3139:          ;
3140: 1020          ORG 1020H          ; ***
3141: 1020 31C04E    CLEAR LD SP,CALSTK          ; BUCHWACK THE STACK
3142: 1023          SYSTEM INTPC          ; INTERPRET
3142: 1023 FF          RST 56
3142: 1024 00          DB INTPC
3142: 0001          IF INTPC = INTPC
3142: 0001          INTPCC DEFL 1
3142:          ENDIF
3142: 1025          ENDM
3142:          DO FILL
3143: 1025 1B          DB FILL+1
3143: 1026          ENDM
3143:
3144: 1026 0040          DW 4000H
3145: 1028 600E          DW CALCLN*BYTEPL
3146: 102A 00          DB 0
3147: 102B          DO RECTAN
3147: 102B 1D          DB RECTAN+1
3147: 102C          ENDM
3147:
3148: 102C 08          DB 8
3149: 102D 52          DB 82
3150: 102E 70          DB 112
3151: 102F 09          DB 9
3152: 1030 FF          DB 11111111B
3153: 1031          DO SETOUT
3153: 1031 17          DB SETOUT+1
3153: 1032          ENDM
3153:
3154: 1032 B8          DB CALCLN*2
3155: 1033 1F          DB 31
3156: 1034 08          DB 8
3157: 1035          DO COLSET
3157: 1035 19          DB COLSET+1
3157: 1036          ENDM
3157:
3158: 1036 1C13          DW CALCLR          ; CALCULATOR COLORS
3159: 1038          DO FILL
3159: 1038 1B          DB FILL+1
3159: 1039          ENDM

```

Home Video Game System Library - Calculator

```

3159:
3160: 1039 C04E          DW  CALRAM
3161: 103B 9400          DW  ENDCR-CALRAM
3162: 103D 00            DB  0
3163: 103E          DO  ACTINT
3163: 103E 0F            DB  ACTINT+1
3163: 103F          ENDM
3163:
3164: 103F          DO  XINTC          ; QUIT INTERPRETING
3164: 103F 03            DB  XINTC+1
3164: 1040          ENDM
3164:
3165: 1040 CD8611          CALL DOCLRE          ; SHOW A ZERO WHEN WE COME UP
3166:          ; GET NEXT KEY DEPRESSION
3167: 1043 214310        KEYGET LD  HL,KEYGET
3168: 1046 E5            PUSH HL
3169: 1047          SYSSUK SENTRY
3169: 1047 FF            RST  56
3169: 1048 43            DB  SENTRY+1
3169: 0000          IF  SENTRY = INTPC
3169:          ENDIF
3169: 1049          ENDM
3169:
3170: 1049 1402          DW  ALKEYS
3171: 104B FE13          CP  SKYD          ; KEY DOWN?
3172: 104D C0            RET  NZ          ; NO - WAIT
3173: 104E 78            LD  A,B
3174: 104F          SYSSUK INDEXB
3174: 104F FF            RST  56
3174: 1050 5D            DB  INDEXB+1
3174: 0000          IF  INDEXB = INTPC
3174:          ENDIF
3174: 1051          ENDM
3174:
3175: 1051 9F10          DW  TOKETB-1
3176: 1053 E60F          AND  0FH          ; ISOLATE 'TRANSLATE TO' FIELD
3177: 1055 4F            LD  C,A
3178: 1056 AE            XOR  (HL)          ; BRANCH ON TOKEN TYPE
3179: 1057 0F            RRCA
3180: 1058 0F            RRCA
3181: 1059 0F            RRCA
3182: 105A 0F            RRCA
3183: 105B 5F            LD  E,A
3184:          ; ARE WE WAITING FOR USER TO TYPE IN A MEMORY NUMBER?
3185: 105C 21C24E        LD  HL,MEMSTA      ; POINT AT MEMORY STATE
3186: 105F 7E            LD  A,(HL)
3187: 1060 A7            AND  A
3188: 1061 2832          JR  Z,$+(KEYGT1-$) ; JUMP IF NOT IN THAT STATE
3189: 1063 7B            LD  A,E          ; WAS DIGIT TYPED?
3190: 1064 A7            AND  A
3191: 1065 C0            RET  NZ          ; NO - GO TO NEXT ENTRY
3192: 1066 57            LD  D,A          ; ZERO OUT D
3193: 1067 79            LD  A,C
3194: 1068 FE0E          CP  0EH          ; DID OPERATOR HIT A DECIMAL PO
3195: 106A C8            RET  Z          ; YES - QUIT
3196:          ; SET MEMORY DIGIT FOR DISPLAY
3197: 106B 23            INC  HL

```

Home Video Game System Library - Calculator

```

3198: 106C 71          LD   (HL),C
3199: 106D 34          INC  (HL)          ; ZERO DESIGNATES NOTHIN
3200: 106E 5F          LD   E,A
3201: 106F 2B          DEC  HL
3202: 1070 7E          LD   A,(HL)       ; S-R STATUS TO A AGAIN
3203: 1071 0F          RRCA              ; TEST STATUS
3204: 1072 3807        JR   C,$+(GMEM1-$) ; JUMP IF RECALL
3205:                ; STORE PROCESSING
3206: 1074 CDCA10      CALL XMEM         ; DO POINTING AND SETUP
3207: 1077 EDB0        LDIR              ; PERFORM STORE
3208: 1079 1809        JR   $+(SHONUM-$)
3209:                ; RECALL PROCESSING
3210: 107B CD0A12      GMEM1 CALL SHOWCK   ; CHECK FOR SHOW MODE
3211: 107E CDCA10      CALL XMEM
3212: 1081 EB          EX   DE,HL
3213: 1082 EDB0        LDIR              ; COPY MEM TO ARG2
3214: 1084 210000      SHONUM LD   HL,0
3215: 1087 22DA4E      LD   (DIGITS),HL ; CLEAR DIGIT FLAGS
3216: 108A 21DD4E      LD   HL,SHOWBF
3217: 108D E5          PUSH HL
3218: 108E CDC712      CALL CONVRT
3219: 1091 E1          POP  HL
3220: 1092 C3DE11      JP   ADIG7        ; JOIN DISPLAY SEQUENCE
3221: 1095                KEYGT1
3222: 1095 7B          LD   A,E
3223: 1096                SYSSUK INDEXW
3223: 1096 FF          RST  56
3223: 1097 5B          DB   INDEXW+1
3223: 0000                IF   INDEXW = INTPC
3223:                ENDIF
3223: 1098                ENDM
3223:
3224: 1098 B810        DW   TOKEJT
3225: 109A D5          PUSH DE
3226: 109B 21C14E      LD   HL,STATEV
3227: 109E 7E          LD   A,(HL)
3228: 109F C9          RET              ; CALL VIA RETURN
3229:                ; TOKEN TYPE-TRANSLATE TABLE
3230:                ; HO NIBBLE IS TYPE, LO NIBBLE IS WHAT TO TRANSLATE TO
3231: 10A0 80          TOKETB DB   80H   ; 1 C
3232: 10A1 60          DB   60H   ; 2 VA
3233: 10A2 61          DB   61H   ; 3 DA
3234: 10A3 31          DB   31H   ; 4 %
3235: 10A4 51          DB   51H   ; 5 MR
3236: 10A5 52          DB   52H   ; 6 MS
3237: 10A6 40          DB   40H   ; 7 +-
3238: 10A7 14          DB   14H   ; 8 DIVIDE
3239: 10A8 07          DB   07H   ; 9 7
3240: 10A9 08          DB   08H   ; 10 8
3241: 10AA 09          DB   09H   ; 11 9
3242: 10AB 13          DB   13H   ; 12 X
3243: 10AC 04          DB   04H   ; 13 4
3244: 10AD 05          DB   05H   ; 14 5
3245: 10AE 06          DB   06H   ; 15 6
3246: 10AF 12          DB   12H   ; 16 -
3247: 10B0 01          DB   01H   ; 17 1
3248: 10B1 02          DB   02H   ; 18 2

```

Home Video Game System Library - Calculator

```

3249: 10B2 03          DB 03H          ; 19 3
3250: 10B3 11          DB 11H          ; 20 +
3251: 10B4 70          DB 70H          ; 21 CE
3252: 10B5 00          DB 00H          ; 22 0
3253: 10B6 0E          DB 0EH          ; 23 .
3254: 10B7 20          DB 20H          ; 24 =
3255:                ; TOKEN TYPE JUMP TABLE
3256: 10B8 8A11        TOKEJT DW ACTNUM        ; 0 NUMBER
3257: 10BA E810        DW OPROUT       ; 1 OPERATOR
3258: 10BC 1111        DW EQROUT       ; 2 EQUAL SIGN
3259: 10BE 2311        DW PERCEN       ; 3 PERCENT
3260: 10C0 DE10        DW CHGSGN       ; 4 CHANGE SIGN
3261: 10C2 D810        DW MEMFUN       ; 5 MEMORY
3262: 10C4 3812        DW SCRLKY       ; 6 SCROLL KEYS
3263: 10C6 AF11        DW CLRENT       ; 7 CLEAR ENTRY
3264: 10C8 2010        DW CLEAR        ; 8 CLEAR CALC
3265:                ; SUBROUTINE TO POINT DE AT A MEMORY AND LOAD HL
3266:                ; INPUT D=0,E=DIGIT (0 TO 9 MEMORY #)
3267:                ; OUTPUT DE = POINTER TO MEM HL=ARG2 BC=12
3268: 10CA 21E64E        XMEM LD HL,CMEM0
3269: 10CD 060B          LD B,11
3270: 10CF 48            LD C,B
3271: 10D0 19            XMEM1 ADD HL,DE
3272: 10D1 10FD          DJNZ $+(XMEM1-$)
3273: 10D3 EB            EX DE,HL
3274: 10D4 21CF4E        LD HL,ARG2
3275: 10D7 C9            RET
3276:                ; MEMORY KEY ROUTINE
3277: 10D8 21C24E        MEMFUN LD HL,MEMSTA ; SET MEMORY STATE
3278: 10DB 71            LD (HL),C ; SET MEMORY STATE
3279: 10DC 1842          JR $+(EQRT1-$)
3280:                ; CHANGE SIGN ROUTINE
3281: 10DE CD0A12        CHGSGN CALL SHOWCK
3282: 10E1                SYSSUK BCDCHS
3282: 10E1 FF            RST 56
3282: 10E2 6B            DB BCDCHS+1
3282: 0000                IF BCDCHS = INTPC
3282:                ENDIF
3282: 10E3                ENDM
3282:
3283: 10E3 0B            DB 11
3284: 10E4 CF4E          DW ARG2
3285: 10E6 1838          JR $+(EQRT1-$)
3286:                ; OPERATOR ROUTINE
3287: 10E8 FE02          OPROUT CP SHOWMD ; ARE WE IN SHOW STATE?
3288: 10EA 2004          JR NZ,$+(OP1-$) ; JUMP IF NOT
3289: 10EC 3601          LD (HL),ARG2MD ; GO TO ARG2 STATE
3290: 10EE 181B          JR $+(OP3-$)
3291: 10F0 FE00          OP1 CP ARG1MD ; ARG1 MODE?
3292: 10F2 08            EX AF,AF'
3293: 10F3 79            LD A,C
3294: 10F4 08            EX AF,AF'
3295: 10F5 200F          JR NZ,$+(OP2-$) ; JUMP IF NOT
3296: 10F7 3601          LD (HL),ARG2MD ; YUP, GO TO ARG2 STATE
3297: 10F9 CD1612        CALL PUSHEN ; PUSH ARG1 DOWN
3298: 10FC                SYSSUK MOVE
3298: 10FC FF            RST 56

```

Home Video Game System Library - Calculator

```

3298: 10FD 5F          DB      MOVE+1
3298: 0000              IF      MOVE = INTPC
3298:                  ENDIF
3298: 10FE              ENDM
3298:
3299: 10FE C44E          DW      ARG1
3300: 1100 0B00          DW      11
3301: 1102 CF4E          DW      ARG2
3302: 1104 1803          JR      $+(OP2A-$)
3303: 1106 CD4611        OP2     CALL DOOP          ; DO IT
3304: 1109 08           OP2A    EX      AF,AF'
3305: 110A 4F           LD      C,A
3306: 110B 79           OP3     LD      A,C
3307: 110C 32C04E        LD      (OPCOD),A
3308: 110F 180F          JR      $+(EQRT1-$)          ; JOIN EQUAL DISPLAY CALL
3309:                  ; EQUALS ROUTINE
3310: 1111 3D           EQROUT  DEC     A          ; ARE WE IN ARG2MD?
3311: 1112 C0           RET     NZ
3312: 1113 3602          LD      (HL),SHOWMD        ; ENTER SHOW STATE
3313: 1115 CD4611        CALL DOOP
3314: 1118 3E05          LD      A,5
3315: 111A 32C04E        LD      (OPCOD),A
3316: 111D CD1612        CALL PUSHEN
3317: 1120 C38410        EQRT1  JP      SHONUM
3318: 1123 A9           PERCEN  XOR     C          ; ARG2 MODE?
3319: 1124 C0           RET     NZ
3320: 1125              SYSSUK  MOVE
3320: 1125 FF           RST     56
3320: 1126 5F           DB      MOVE+1
3320: 0000              IF      MOVE = INTPC
3320:                  ENDIF
3320: 1127              ENDM
3320:
3321: 1127 CF4E          DW      ARG2
3322: 1129 0900          DW      9
3323: 112B D04E          DW      ARG2+1
3324: 112D 32D84E        LD      (ARG2+9),A
3325: 1130 21C44E        LD      HL,ARG1
3326: 1133 060B          LD      B,11
3327: 1135 3ACE4E        LD      A,(ARG1+10)        ; SAVE SIGN OF ARG1
3328: 1138              SYSTEM  BCDMUL
3328: 1138 FF           RST     56
3328: 1139 66           DB      BCDMUL
3328: 0000              IF      BCDMUL = INTPC
3328:                  ENDIF
3328: 113A              ENDM
3328:
3329: 113A 32CE4E        LD      (ARG1+10),A        ; RESTORE SIGN OF ARG1
3330: 113D EB           EX      DE,HL
3331: 113E 010B00        LD      BC,11
3332: 1141 CD7511        CALL OVRCHK          ; CHECK FOR OVERFLOW
3333: 1144 18DA          JR      EQRT1
3334:                  ; SUBROUTINE TO DO OPERATION
3335:                  ; NOTE THIS ROUTINE SETS BOTH ARG1 AND ARG2 EQUAL TO RESU
3336: 1146 CD1612        DOOP    CALL PUSHEN
3337: 1149 3AC04E        LD      A,(OPCOD)
3338: 114C              SYSSUK  INDEXW

```

Home Video Game System Library - Calculator

```

3338: 114C FF          RST    56
3338: 114D 5B          DB     INDEXW+1
3338: 0000              IF     INDEXW = INTPC
3338:                ENDIF
3338: 114E              ENDM
3338:
3339: 114E 5811          DW     DOOPTB-2
3340: 1150 D5            PUSH  DE
3341: 1151 21CF4E        LD     HL,ARG2
3342: 1154 11C44E        LD     DE,ARG1
3343: 1157 060B          LD     B,11
3344: 1159 C9            RET                                ; JUMP USING RETURN
3345: 115A 6211          DOOPTB DW     DOADD
3346: 115C 6511          DW     DOSUB
3347: 115E 6811          DW     DOMUL
3348: 1160 6B11          DW     DODIV
3349: 1162          DOADD SYSTEM BCDADD
3349: 1162 FF          RST    56
3349: 1163 62          DB     BCDADD
3349: 0000          IF     BCDADD = INTPC
3349:                ENDIF
3349: 1164          ENDM
3349:
3350: 1164 01          DB     01
3351: 1165          DOSUB SYSTEM BCDSUB
3351: 1165 FF          RST    56
3351: 1166 64          DB     BCDSUB
3351: 0000          IF     BCDSUB = INTPC
3351:                ENDIF
3351: 1167          ENDM
3351:
3352: 1167 01          DB     01
3353: 1168          DOMUL SYSTEM BCDMUL
3353: 1168 FF          RST    56
3353: 1169 66          DB     BCDMUL
3353: 0000          IF     BCDMUL = INTPC
3353:                ENDIF
3353: 116A          ENDM
3353:
3354: 116A 01          DB     01
3355: 116B          DODIV SYSTEM BCDDIV
3355: 116B FF          RST    56
3355: 116C 68          DB     BCDDIV
3355: 0000          IF     BCDDIV = INTPC
3355:                ENDIF
3355: 116D          ENDM
3355:
3356: 116D          NUMCHK SYSSUK MOVE      ; ARG2 = ARG1
3356: 116D FF          RST    56
3356: 116E 5F          DB     MOVE+1
3356: 0000          IF     MOVE = INTPC
3356:                ENDIF
3356: 116F          ENDM
3356:
3357: 116F CF4E          DW     ARG2
3358: 1171 0B00          DW     11
3359: 1173 C44E          DW     ARG1

```

Home Video Game System Library - Calculator

```

3360: 1175 09          OVRCHK  ADD  HL,BC
3361: 1176 2B          DEC  HL
3362: 1177 7E          LD   A,(HL)
3363: 1178 E60F        AND  0FH          ; OVERFLOW SET?
3364: 117A C8          RET  Z           ; QUIT IF NOT
3365: 117B 3E01       LD   A,1
3366: 117D 77          LD   (HL),A
3367: 117E 3C          INC  A           ; FOR SHOW MODE
3368: 117F 32C14E     LD   (STATEV),A
3369: 1182 CD1612     CALL PUSHEN
3370: 1185 E1          POP  HL          ; THROW OUT RETURN ADDRESS
3371: 1186 0E00       DOCLRE LD  C,0      ; FORCE CLEAR ENTRY
3372: 1188 1825       JR   $+(CLRENT-$)
3373:                ; SUBROUTINE TO ACCEPT A DIGIT
3374:                ; ONE DIGIT
3375: 118A CD0A12       ACTNUM CALL SHOWCK
3376: 118D 21DA4E     LD   HL,DIGITS
3377: 1190 11CF4E     LD   DE,ARG2
3378: 1193 7E          LD   A,(HL)      ; HOW MANY DIGITS SO FAR?
3379: 1194 FE0A       CP   10
3380: 1196 C8          RET  Z           ; QUIT IF LIMIT ATTAINED
3381: 1197 23          INC  HL          ; DECIMAL POINT ENTERED YET?
3382: 1198 7E          LD   A,(HL)
3383: 1199 A7          AND  A
3384: 119A 280F       JR   Z,$+(ADIG1-$) ; JUMP IF NO
3385: 119C B9          CP   C           ; IS INPUT A DECIMAL POINT?
3386:                ; (FLAG = 0E IF SET!!)
3387: 119D C8          RET  Z           ; QUIT IF EXTRA
3388: 119E 79          LD   A,C
3389: 119F 23          INC  HL
3390: 11A0 4E          LD   C,(HL)      ; C = DECIMAL POINT
3391: 11A1 35          DEC  (HL)
3392: 11A2 EB          EX   DE,HL
3393: 11A3          SYSTEM STOREN  ; STORE THE NIBBLE
3393: 11A3 FF          RST  56
3393: 11A4 58          DB   STOREN
3393: 0000          IF   STOREN = INTPC
3393:                ENDIF
3393: 11A5          ENDM
3393:
3394: 11A5 4F          LD   C,A         ; GET DIGIT BACK IN C
3395: 11A6 EB          EX   DE,HL
3396: 11A7 2B          DEC  HL
3397: 11A8 2B          DEC  HL
3398: 11A9 182A       JR   $+(ADIG4-$) ; GO ADD TO SHOW BUFFER
3399:                ; PRE DECIMAL POINT
3400: 11AB 2B          ADIG1 DEC HL      ; BACKUP TO DIGITS
3401: 11AC B6          OR   (HL)        ; HOW MANY ENTRYRS?
3402: 11AD 2012       JR   NZ,$+(ADIG2-$) ; ADIG2 IF NOT ZERO
3403: 11AF CD0A12     CLRENT CALL SHOWCK ; PUSH DOWN IF RESULT IS THERE
3404: 11B2 C5          PUSH BC          ; CLEAR ARG2, SHOWBUF AND VARS
3405: 11B3          SYSSUK FILL
3405: 11B3 FF          RST  56
3405: 11B4 1B          DB   FILL+1
3405: 0000          IF   FILL = INTPC
3405:                ENDIF
3405: 11B5          ENDM

```

Home Video Game System Library - Calculator

```

3405:
3406: 11B5 CF4E          DW ARG2
3407: 11B7 1500         DW 21
3408: 11B9 00          DB 0
3409: 11BA 21DA4E       LD HL,DIGITS
3410: 11BD C1          POP BC
3411: 11BE B1          OR C           ; WAS DIGIT ZERO?
3412: 11BF 281C        JR Z,$+(ADIG6-$) ; JUMP IF SO
3413:                ; CODE FOR NONZERO PREVIOUS ENTRY OR NONZERO FIRST ENTRY
3414: 11C1 79          ADIG2 LD A,C
3415: 11C2 FE0E        CP 0EH        ; WAS CHAR A DECIMAL POINT?
3416: 11C4 2007        JR NZ,$+(ADIG3-$) ; NO ADIG3
3417: 11C6 23          INC HL        ; TO DECIMAL FLAG
3418: 11C7 77          LD (HL),A    ; SET IT
3419: 11C8 23          INC HL
3420: 11C9 3609        LD (HL),9    ; SET DECIMAL POINT POINTER
3421: 11CB 1809        JR $+(ADIG5-$) ; GO STUFF INTO SHOW BUFFER
3422: 11CD 13          ADIG3 INC DE    ; LEADING NUMERIC - STUFF IT
3423: 11CE 13          INC DE
3424: 11CF 13          INC DE
3425: 11D0 13          INC DE
3426: 11D1 13          INC DE
3427: 11D2 CD0F13     CALL ADDTB
3428: 11D5 34          ADIG4 INC (HL)  ; BUMP DIGIT COUNTER
3429: 11D6 79          ADIG5 LD A,C    ; GET DIGIT BACK
3430: 11D7 11DD4E      LD DE,SHOWBF ;ADD DIGIT TO SHOW BUFFER
3431: 11DA CD0F13     CALL ADDTB
3432: 11DD EB          ADIG6 EX DE,HL
3433: 11DE                ADIG7 XYRELL DE,8,83
3433: 11DE 110853       LD DE,[(83 SHL 8)+8]
3433: 11E1                ENDM
3433:
3434: 11E1 0E07         LD C,0111B   ; DISPLAY ALWAYS POSITIVE
3435:                ; DISPLAY RECALL OR STORE?
3436: 11E3 E5          PUSH HL
3437: 11E4 21C24E       LD HL,MEMSTA
3438: 11E7 7E          LD A,(HL)
3439: 11E8 A7          AND A
3440: 11E9 2802         JR Z,$+(ADIG8-$) ; ADIG8 IF NOT WANTED
3441: 11EB C631         ADD A,'R'-21H
3442: 11ED C620         ADIG8 ADD A,20H
3443: 11EF                SYSTEM CHRDIS
3443: 11EF FF          RST 56
3443: 11F0 32          DB CHRDIS
3443: 0000             IF CHRDIS = INTPC
3443:                ENDIF
3443: 11F1                ENDM
3443:
3444:                ; SAME THING FOR STORE-RECALL DIGIT
3445: 11F1 23          INC HL
3446: 11F2 7E          LD A,(HL)
3447: 11F3 A7          AND A
3448: 11F4 2807         JR Z,$+(ADIG9-$)
3449: 11F6 C60F        ADD A,'0'-21H
3450: 11F8 3600        LD (HL),0    ; CLEAR DIGIT AND RECALL MODE
3451: 11FA 2B          DEC HL
3452: 11FB 3600        LD (HL),0

```

Home Video Game System Library - Calculator

```

3453: 11FD C620      ADIG9  ADD  A,20H
3454: 11FF          SYSTEM CHRDIS
3454: 11FF FF        RST   56
3454: 1200 32        DB    CHRDIS
3454: 0000          IF    CHRDIS = INTPC
3454:          ENDIF
3454: 1201          ENDM
3454:
3455: 1201 E1          POP  HL
3456: 1202 3AC04E      LD   A,(OPCOD)
3457: 1205 0E03        LD   C,0011B
3458: 1207 C38A12      JP   NUMBA
3459:          ; SUBROUTINE TO CHECK TO SEE IF WE ARE IN SHOW MODE
3460:          ; IF SO - PUSH ARG2 AND GO INTO ARG1 MODE
3461: 120A 21C14E      SHOWCK LD   HL,STATEV
3462: 120D 7E          LD   A,(HL)
3463: 120E FE02        CP   SHOWMD
3464: 1210 C0          RET  NZ
3465: 1211 AF          XOR  A
3466: 1212 77          LD   (HL),A
3467: 1213 2B          DEC  HL
3468: 1214 77          LD   (HL),A      ; CLEAR OPERATOR
3469: 1215 C9          RET
3470:          ; SUBROUTINE TO ADD AN ENTRY TO THE STACK
3471: 1216          PUSHEN SYSSUK SCROLL      ; PUSH STACK UP
3471: 1216 FF        RST   56
3471: 1217 31        DB    SCROLL+1
3471: 0000          IF    SCROLL = INTPC
3471:          ENDIF
3471: 1218          ENDM
3471:
3472: 1218 2800        DW   BYTEPL
3473: 121A 08          DB   8
3474: 121B 5C          DB   MAXSTK
3475: 121C 2040        DW   4020H      ; (FIRST LINE)
3476: 121E 21584E      LD   HL,STKBOT
3477: 1221 CDC712      CALL CONVRT      ; CONVERT AND ADD TO STACK
3478: 1224 21E44E      LD   HL,NLINES  ; HOW MANY ENTRIES?
3479: 1227 7E          LD   A,(HL)      ; ON THE STACK?
3480: 1228 FE5C        CP   MAXSTK      ; AT MAX?
3481: 122A 2801        JR   Z,$+(PUSHE1-$) ; JUMP IF SO
3482: 122C 34          INC  (HL)        ; BUMP ENTRY COUNT
3483:          ; ARE WE AT TOP OR BOTTOM OF STACK?
3484: 122D 23          PUSHE1 INC HL      ; POINT AR SCHOLL POINTER
3485: 122E 7E          LD   A,(HL)
3486: 122F A7          AND  A          ; JUMP IF AT BOTTOM
3487: 1230 2819        JR   Z,$+(SCRLU1-$)
3488: 1232 FE52        CP   MAXSTK-10  ; OR IF AT TOP
3489: 1234 2815        JR   Z,$+(SCRLU1-$) ; FIX SCROLL POINTER
3490: 1236 34          INC  (HL)        ; FIX SCROLL POINTER
3491: 1237 C9          RET             ; AND QUIT
3492:          ; SCROLL KEY ENTRY
3493: 1238 AF          SCRLKY XOR  A          ; CLEAR KEYSEX FOR SCROLL MOTOR
3494: 1239 32E34F      LD   (KEYSEX),A
3495: 123C 21E44E      LD   HL,NLINES
3496: 123F 7E          LD   A,(HL)      ; A = NUMBER OF LINES ON STK
3497: 1240 CB41        BIT  0,C          ; UP OR DOWN

```

Home Video Game System Library - Calculator

```

3498: 1242 201A          JR  NZ,$+(SCRLDN-$)  ; JUMP IF DOWN
3499: 1244 A7           AND  A                ; QUIT IF ZERO LINES ON STACK
3500: 1245 C8           RET  Z
3501: 1246 23           INC  HL                ; WHERE'S THE SCROLL POINTER?
3502: 1247 7E           LD   A,(HL)
3503: 1248 A7           AND  A                ; QUIT IF AT BOTTOM
3504: 1249 C8           RET  Z
3505: 124A 35           DEC  (HL)             ; ELSE DECREMENTETH
3506:                ; PUSHEN JOINS HERE
3507: 124B 3E08        SCRLU1 LD  A,8            ; DO IT 8 TIMES
3508: 124D                SCRLU2 SYSSUK SCROLL
3508: 124D FF           RST  56
3508: 124E 31           DB   SCROLL+1
3508: 0000                IF   SCROLL = INTPC
3508:                ENDF
3508: 124F                ENDM
3508:
3509: 124F 2800        DW   BYTEPL
3510: 1251 20           DB   32
3511: 1252 4F           DB   79
3512: 1253 2840        DW   4028H
3513: 1255 3D           DEC  A
3514: 1256 20F5        JR  NZ,$+(SCRLU2-$)
3515: 1258 3C           INC  A
3516: 1259                XYRELL DE,24,73
3516: 1259 111849       LD   DE,[(73 SHL 8)+24]
3516: 125C                ENDM
3516:
3517: 125C 1818        JR  $+(SCRLD1-$)     ; JOIN SCRLDN DISPLAY CALL
3518:                ; SCROLL DOWN ROUTINE
3519: 125E 23           SCRLDN INC HL            ; ADVANCE THE SCROLL POINTER
3520: 125F 96           SUB  (HL)             ; SUBTRACT FROM LINES ON STACK
3521: 1260 FE0B        CP   11              ; FAR ENOUGH APART?
3522: 1262 D8           RET  C                ; QUIT IF NOT
3523: 1263 34           INC  (HL)             ; YEAH - ADVANCE SCROLL POINTER
3524: 1264 3E08        LD   A,8
3525: 1266                SCRLD0 SYSSUK SCROLL
3525: 1266 FF           RST  56
3525: 1267 31           DB   SCROLL+1
3525: 0000                IF   SCROLL = INTPC
3525:                ENDF
3525: 1268                ENDM
3525:
3526: 1268 D8FF        DW   -BYTEPL
3527: 126A 20           DB   32
3528: 126B 50           DB   80
3529: 126C 804C        DW   4000H+3200
3530: 126E 3D           DEC  A
3531: 126F 20F5        JR  NZ,$+(SCRLD0-$)
3532: 1271 3E0A        LD   A,10
3533: 1273                XYRELL DE,24,1
3533: 1273 111801       LD   DE,[(1 SHL 8)+24]
3533: 1276                ENDM
3533:
3534: 1276 21E54E       SCRLD1 LD  HL,SCRPTR
3535: 1279 86           ADD  A,(HL)          ; ADD SCROLL PTR
3536: 127A 21804E       LD   HL,STKBOT+BYTEPL

```

Home Video Game System Library - Calculator

```

3537: 127D 01D8FF          LD    BC,-BYTEPL
3538: 1280 09          SCRLD2 ADD  HL,BC
3539: 1281 3D          DEC  A
3540: 1282 20FC          JR   NZ, $(SCRLD2-$)
3541: 1284 0E0D          LD   C,13          ; GET OPERATION
3542: 1286          SYSTEM INDEXN
3542: 1286 FF          RST  56
3542: 1287 56          DB   INDEXN
3542: 0000          IF   INDEXN = INTPC
3542:          ENDIF
3542: 1288          ENDM
3542:
3543: 1288 0E00          LD   C,0          ; WRITE WITH BLACK BACKGROUND
3544: 128A F5          NUMBA PUSH AF
3545: 128B 41          LD   B,C
3546: 128C 0E0C          LD   C,12
3547: 128E          SYSTEM INDEXN
3547: 128E FF          RST  56
3547: 128F 56          DB   INDEXN
3547: 0000          IF   INDEXN = INTPC
3547:          ENDIF
3547: 1290          ENDM
3547:
3548: 1290 CB4F          BIT  1,A
3549: 1292 2016          JR   NZ, $(OVFERR-$)
3550: 1294 CB47          BIT  0,A
3551: 1296 3E04          LD   A,0100B
3552: 1298 2802          JR   Z, $(NUMBA1-$)
3553: 129A 3E08          LD   A,1000B
3554: 129C B0          NUMBA1 OR   B
3555: 129D 4F          LD   C,A
3556: 129E 068B          LD   B,8BH
3557: 12A0          SYSTEM DISNUM
3557: 12A0 FF          RST  56
3557: 12A1 36          DB   DISNUM
3557: 0000          IF   DISNUM = INTPC
3557:          ENDIF
3557: 12A2          ENDM
3557:
3558: 12A2 F1          POP  AF
3559: 12A3          SYSSUK INDEXB
3559: 12A3 FF          RST  56
3559: 12A4 5D          DB   INDEXB+1
3559: 0000          IF   INDEXB = INTPC
3559:          ENDIF
3559: 12A5          ENDM
3559:
3560: 12A5 C112          DW   OPTBL
3561: 12A7          SYSTEM CHRDIS
3561: 12A7 FF          RST  56
3561: 12A8 32          DB   CHRDIS
3561: 0000          IF   CHRDIS = INTPC
3561:          ENDIF
3561: 12A9          ENDM
3561:
3562: 12A9 C9          NULKEY RET
3563: 12AA          OVFERR

```

Home Video Game System Library - Calculator

```

3564: 12AA 48          LD    C,B
3565: 12AB CBD9        SET   3,C
3566: 12AD 21B412     LD    HL,OVFMSG
3567: 12B0          SYSTEM STRDIS
3567: 12B0 FF          RST   56
3567: 12B1 34          DB    STRDIS
3567: 0000          IF    STRDIS = INTPC
3567:                ENDIF
3567: 12B2          ENDM
3567:
3568: 12B2 F1          POP   AF
3569: 12B3 C9          RET
3570: 12B4 202A4F56    OVFMSG DB  ' ','*', 'O', 'V', 'E', 'R', 'F', 'L', 'O', 'W', '*', '
'
          4552464C
          4F572A20
3571: 12C0 00          DB    0
3572: 12C1 20          OPTBL DB  ' '          ; NULL OP
3573: 12C2 2B          DB  '+'
3574: 12C3 2D          DB  '-'
3575: 12C4 62          DB  62H
3576: 12C5 63          DB  63H
3577: 12C6 3D          DB  '='
3578: 12C7 11CF4E     CONVRT LD  DE,ARG2
3579: 12CA 3AC04E     LD   A,(OPCOD)
3580: 12CD 0E0D     LD   C,13
3581: 12CF          SYSTEM STOREN      ; STORE THAT NIBBLE
3581: 12CF FF          RST   56
3581: 12D0 58          DB    STOREN
3581: 0000          IF    STOREN = INTPC
3581:                ENDIF
3581: 12D1          ENDM
3581:
3582:                ; TRANSFER SIGN
3583: 12D1 3AD94E     LD   A,(ARG2+10)  ; GET SIGN BYTE
3584: 12D4 07          RLCA          ; PUT SIGN IN LOW BIT
3585: 12D5 0D          DEC   C
3586: 12D6          SYSTEM STOREN
3586: 12D6 FF          RST   56
3586: 12D7 58          DB    STOREN
3586: 0000          IF    STOREN = INTPC
3586:                ENDIF
3586: 12D8          ENDM
3586:
3587: 12D8 EB          PACKER EX  DE,HL      ; DE = SAVE HL = ARG
3588: 12D9 010600     LD   BC,6        ; CLEAR SAVE
3589: 12DC AF          XOR   A
3590: 12DD          SYSTEM FILL
3590: 12DD FF          RST   56
3590: 12DE 1A          DB    FILL
3590: 0000          IF    FILL = INTPC
3590:                ENDIF
3590: 12DF          ENDM
3590:
3591: 12DF 0E13          LD   C,19
3592: 12E1          CCVT1 SYSTEM INDEXN    ; GET NIBBLE
3592: 12E1 FF          RST   56

```

Home Video Game System Library - Calculator

```

3592: 12E2 56          DB      INDEXN
3592: 0000          IF      INDEXN = INTPC
3592:                ENDIF
3592: 12E3          ENDM
3592:
3593: 12E3 CD0F13      CALL  ADDTB          ; ADD INTO 'SAVE'
3594: 12E6 200D      JR    NZ,$+(CCVT1A-$) ; JUMP IF DONE
3595: 12E8 0D        DEC   C              ; DECREMENT ARGPTR
3596: 12E9 79        LD   A,C
3597: 12EA FE09      CP    9              ; IS IT NOW 9
3598: 12EC 20F3      JR    NZ,$+(CCVT1-$) ; NO - KEEP GOING
3599: 12EE 3E0E      LD   A,0EH          ; YEAH - PUT IN DECIMAL POINT
3600: 12F0 CD0F13      CALL  ADDTB
3601: 12F3 28EC      JR    Z,$+(CCVT1-$) ; JUMP BACK IF OKAY
3602:                ; NOW REMOVE TRAILING ZEROS AND DECIMAL POINT
3603: 12F5 EB        CCVT1A EX  DE,HL      ; HL = SAVE
3604: 12F6 7E        CCVT2  LD   A,(HL)    ; GET LO NIBBLE
3605: 12F7 E60F      AND   0FH           ; IS IT NONZERO?
3606: 12F9 2005      JR    NZ,$+(CCVT3-$) ; JUMP IF NOT ZERO
3607: 12FB CD0313      CALL  KILTRZ        ; REMOVE ONE TRAILING ZERO
3608: 12FE 18F6      JR    $+(CCVT2-$)
3609: 1300 FE0E      CCVT3  CP   0EH      ; DECIMAL POINT
3610: 1302 C0        RET   NZ            ; QUIT IF NOT
3611: 1303 010600     KILTRZ LD   BC,6
3612: 1306 09        ADD  HL,BC
3613: 1307 A7        AND  A
3614: 1308 41        LD   B,C
3615: 1309 2B        KILTR1 DEC  HL
3616: 130A ED67      RRD
3617: 130C 10FB      DJNZ  $+(KILTR1-$)
3618: 130E C9        RET
3619:                ; SUBROUTINE TO ADD A DIGIT TO NUMBER POINTED AT BY DE
3620: 130F 0605      ADDTB LD   B,5
3621: 1311 EB        ADDTD  EX  DE,HL
3622: 1312 E5        PUSH  HL
3623: 1313 ED6F      ADDTB1 RLD
3624: 1315 23        INC  HL
3625: 1316 10FB      DJNZ  $+(ADDTB1-$)
3626: 1318 77        LD   (HL),A        ; STUFF LAST
3627: 1319 E1        POP  HL
3628: 131A EB        EX   DE,HL
3629: 131B C9        RET
3630:                ; COLORS FOR CALCULATOR
3631: 131C 76        CALCLR DB  76H      ; GREY
3632: 131D 62        DB   62H          ; RED
3633: 131E 00        DB   00H          ; BLACK
3634: 131F 07        DB   07H          ; WHITE
3635: 1320 0707      DW   0707H
3636: 1322 0707      DW   0707H
3637:                ; *****
3638:                ; * CALCULATOR RAM *
3639:                ; *****
3640:                ; ORG 4000H
3641:                ; DEFS CALCLN*BYTEPL
3642:                ; DEFS 96
3643: 4EC0          CALSTK EQU $4EC0
3644: 4EC0          CALRAM EQU $4EC0

```

Home Video Game System Library - Calculator

```

3645: 4EC0          OPCODE EQU $4EC0 ; OPERATION TO DO
3646: 4EC1          STATEV EQU $4EC1 ; STATE VARIABLE
3647: 4EC2          MEMSTA EQU $4EC2 ; MEMORY STATE VARIABLE
3648: 4EC3          MEMDIG EQU $4EC3 ; MEMORY DIGIT VARIABLE
3649: 4EC4          ARG1 EQU $4EC4 ; ARGUMNET 1 - 11 BYTES
3650:                ; *** NOTE *** CODE EXPECTS ARG2 THRU SHOWBF TO BE IN ONE
3651: 4ECF          ARG2 EQU $4ECF ; ARGUMENT 2 - 11 BYTES
3652: 4EDA          DIGITS EQU $4EDA ; DIGITS COUNTER
3653: 4EDB          DPOINT EQU $4EDB ; DECIMAL POINT ENTERED FLAG
3654: 4EDC          POINTP EQU $4EDC ; DECIMAL POINT POINTER
3655: 4EDD          SHOWBF EQU $4EDD ; ENTRY FEEDBACK BUFFER - 7
BYTES
3656: 4EE4          NLines EQU $4EE4 ; NUMBER OF LINES ON SCROLL STA
3657: 4EE5          SCRPTR EQU $4EE5 ; CURRENT POS OF SCROLL SYSTEM
3658: 4EE6          CMEM0 EQU $4EE6 ; CALCULATOR MEMORYS - 110 BYTES
3659: 4F54          ENDCR EQU $4F54
3660:                ;
3661:                ;          END OF CALCULATOR PROGRAM
3662:                ;
3663:                ; FOUR FILLER BYTES BETWEEN SCRIBBLING AND
3664:                ; CALCULATOR TO MATCH 3159 ROM
3665: 1324 FFFFFFFF          DB 0FFH, 0FFH, 0FFH, 0FFH

```

Home Video Game System Library - Checkmate

```

3666:          ;
3667:          ;*****
3668:          ;* H V G   C H E C K M A T E *   VER. 2.0 4-2-2003
3669:          ;*****
3670:          ;
3671:          ;
3672:          ; M A C R O S
3673:          ;
3674:          DEF4X  MACRO A4X, B4X, C4X, D4X
3675:                   DB  A4X
3676:                   DB  B4X
3677:                   DB  C4X
3678:                   DB  D4X
3679:                   ENDM
3680:          WRECK  MACRO
3681:                   DW  (9 SHL 8)+32
3682:                   DB  0000B
3683:                   ENDM
3684:          ;
3685:          ;
3686:          ; E Q U A T E S
3687:          ;
3688: 0000          OLDWAY EQU 1-NWHDWR          ;1=DO OLD WAY 0=DO NEW WAY
3689: 0001          NEWWAY EQU 1-OLDWAY          ;OPPOSITE OF OLDWAY
3690:          ; VARIOUS 'EQU'S
3691: 000C          RLMOVE EQU 1100B            ;RIGHT AND LEFT MOVES
3692: 0003          UDMOVE EQU 0011B            ;UP AND DOWN MOVES
3693: 0002          NOBIT  EQU 2                ;# OF GAME BITS
3694: 0003          NPBIT  EQU 3                ;# PLAYERS BIT
3695: 0003          ANIMAX EQU 3                ;MAX # TICKS PER ANIMATION FRAM
3696: 0098          XMAX   EQU (BYTEPL-2)*4     ;MAX X COORD - CHANGED TO MATCH
ROOM
3697: 0015          YLINES EQU 21                ;# VERT BLOCKS
3698: 000B          LOWY   EQU 11                ;LOWEST Y COORD
3699: 005B          YMAX   EQU ((YLINES-1)*4)+LOWY ;MAX Y COORD
3700: 0000          LOWX   EQU 0                ;LOWEST X COORD
3701: 0008          AMOVE  EQU 8H                ;AN ARBITRARY MOVE
3702: 0009          MUSVOL EQU 09H                ;MUSIC VOLUME
3703: 0024          TDOPT  EQU 100100B          ;TIME DISPLAY OPTIONS
3704: 0044          CDOPT  EQU 01000100B        ;COUNT DOWN OPT
3705: 0010          WRITOR EQU 010000B          ;WRITE WITH MAGIC OR
3706:          ; PLAYER PACKER OFFSETS
3707: 0000          LASTSW EQU 0                ;LAST SWITCH SETTING
3708: 0001          LASTMV EQU 1                ;LAST ACTUAL MOVE
3709: 0002          CURSW  EQU 2                ;CURRENT SWITCH SETTING
3710: 0003          AROT   EQU 3                ;ARROW ROTATION SETTING
3711: 0004          ARRX   EQU 4                ;ARROW X COORD
3712: 0005          ARRY   EQU 5                ;ARROW Y COORD
3713: 0006          PSTAT  EQU 6                ;PLAYER STATUS
3714:          ; PLAYER STATUS MASKS
3715: 0080          ACTIVE EQU $80
3716: 0040          HUMAN  EQU $40
3717: 0007          ACTBIT EQU 7                ;1=ACTIVE 0=DEAD
3718: 0006          HUMBIT EQU 6                ;1=HUMAN 0=COMPUTER
3719:          ;SCREEN TABS
3720: 0028          XTAB1  EQU ((BYTEPL/4)*4)
3721: 0050          XTAB2  EQU XTAB1*2

```

Home Video Game System Library - Checkmate

```

3722: 0078          XTAB3   EQU   XTAB1*3
3723: 0014          YTAB    EQU   (((Y LINES-1)/4)*4)
3724: 001F          YTAB1   EQU   YTAB+LOWY
3725: 0033          YTAB2   EQU   (2*YTAB)+LOWY
3726: 0047          YTAB3   EQU   (3*YTAB)+LOWY
3727:                ;OFFSETS FOR EACH PLAYERS ROM DATA
3728:                ;NOTE0   EQU   0                ;EACH DIRECTIONS NOTES
3729:                ;NOTE1   EQU   1 ; THESE FOUR LINES COMMENTED OUT BECAUSE
3730:                ;NOTE2   EQU   2 ; THEY INTERFERE WITH BALLYEQU.H AND ARE
3731:                ;NOTE3   EQU   3 ; NOT USED ANYWAY- ADAM, 2-16-2002
3732: 0004          PPATL   EQU   4                ;PLAYER PAT ADDR LOW
3733: 0005          PPATH   EQU   5                ;PLAYER PAT ADDR HIGH
3734: 0006          PCDOP   EQU   6                ;PLAYER CHAR DISP OPT
3735: 0007          PSPOSX  EQU   7                ;X COORD OF PLAYER SCORE
3736: 0008          PSPOSY  EQU   8                ;Y COORD OF PLAYER SCORE
3737: 0009          PSDOP   EQU   9                ;PLAYER SCORE DISP OPT
3738:                ;MORE EQU'S
3739: 00F6          FORCEM   EQU   $F6                ;VAL TO FORCE RANDOM MOVE
3740: 0004          WIDTH   EQU   $4                ;# PIXELS WIDE OF PLAYER PAT
3741: 0004          HEIGHT  EQU   $4                ;# PIXELS HIGH OF PLAYER PAT
3742: 0D20          ALLBYT   EQU   (Y LINES*4)*BYTEPL ;ALL BYTES ON SCREEN
3743: 41B8          STARTS  EQU   (LOWY*BYTEPL)+NORMEM ; LOWEST ADDR OF PLAY FI
3744: 0001          PATXSZ  EQU   1                ;#BYTES WIDE OF PLAYER PATTERN
3745: 0004          PATYSZ  EQU   4                ;#BYTES HIGH OF PLAYER PATTERN
3746: 0104          PATDIM  EQU   (PATXSZ SHL 8) OR PATYSZ ;PATTERNS DIMENSIONS
3747: 000F          JUSJOY  EQU   $F                ;ONLY JOY STICK BITS
3748: 0008          CBLN    EQU   8                ;COLOR BLOCK LENGTH
3749: 0008          SBLN    EQU   8                ;SOUND BLOCK LENGTH
3750: 0000          WPONOF  EQU   0
3751: 0001          WPOPT   EQU   1
3752: 0002          WPPAL   EQU   2
3753: 0003          WPPAH   EQU   3
3754: 0005          WPXSIZ  EQU   5
3755: 0004          WPYSIZ  EQU   4
3756:                ;
3757:                ;
3758:                ;
3759:                ;   ORG   NORMEM+0F96H ;SHOULD BE EQUAL TO RSTART
3760:                ;UNCLEARED RAM - THESE ARE NOW EQUS, NOT DEFS
3761: 4F96          UNCRAM  EQU   $4F96
3762: 4F96          CURSCR  EQU   $4F96                ;ALL CURRENT SCORES
3763:                ;CLEARED
3764: 4FA2          CNOPL   EQU   $4FA2                ;CURRENT # PLAYERS
3765: 4FA3          PPIX    EQU   $4FA3                ;WHO IS CURRENT PLAYER
3766: 4FA4          CNOHUM  EQU   $4FA4                ;CURRENT # HUMANS
3767: 4FA5          TARRX   EQU   $4FA5                ;TEMP ARROW X COORD
3768: 4FA6          TARRY   EQU   $4FA6                ;TEMP ARROW Y COORD
3769: 4FA7          RMASK   EQU   $4FA7                ;ROTATE MASK
3770: 1328          PPACKS:                ;START OF PLAYER PACKETS
3771: 4FA8          PLAY0   EQU   $4FA8
3772: 4FAF          PLAY1   EQU   $4FAF
3773: 4FB6          PLAY2   EQU   $4FB6
3774: 4FBD          PLAY3   EQU   $4FBD
3775: 4FC4          ENDRAM  EQU   $4FC4
3776: 4FA1          RSTART  EQU   BEGRAM-(ENDRAM-UNCRAM)+1 ; SHOULD BE RAM STA
3777: 1328          ORG     1328H
3778: 1328          ONETIM:

```

Home Video Game System Library - Checkmate

```

3779:                                ;ONE TIME ONLY HOUSEKEEPING
3780: 1328 31964F                    LD  SP,UNCRAM
3781: 132B                                SYSSUK GETPAR
3781: 132B FF                          RST  56
3781: 132C 4D                          DB   GETPAR+1
3781: 0000                                IF   GETPAR = INTPC
3781:                                ENDIF
3781: 132D                                ENDM
3781:
3782: 132D 3502                        DW   NOGAME
3783: 132F 82                          DB   $82
3784: 1330 DC4F                        DW   CT7
3785: 1332                                SYSSUK GETPAR
3785: 1332 FF                          RST  56
3785: 1333 4D                          DB   GETPAR+1
3785: 0000                                IF   GETPAR = INTPC
3785:                                ENDIF
3785: 1334                                ENDM
3785:
3786: 1334 2802                        DW   NOPLAY
3787: 1336 01                          DB   1
3788: 1337 F34F                        DW   NUMPLY
3789: 1339                                SYSSUK FILL
3789: 1339 FF                          RST  56
3789: 133A 1B                          DB   FILL+1
3789: 0000                                IF   FILL = INTPC
3789:                                ENDIF
3789: 133B                                ENDM
3789:
3790: 133B 964F                        DW   CURSCR
3791: 133D 0C00                        DW   12
3792: 133F 00                          DB   0
3793: 1340                                FIREIT:
3794:                                ;REENTRY POINT FROM END OF GAME
3795: 1340 F3                          DI
3796: 1341 31964F                    LD  SP,UNCRAM
3797: 1344                                SYSTEM INTPC
3797: 1344 FF                          RST  56
3797: 1345 00                          DB   INTPC
3797: 0001                                IF   INTPC = INTPC
3797: 0001 INTPCC DEFL  1
3797:                                ENDIF
3797: 1346                                ENDM
3797:
3798:                                ;OUPUT COLOR BLOCK
3799: 1346                                DO   COLSET
3799: 1346 19                          DB   COLSET+1
3799: 1347                                ENDM
3799:
3800: 1347 A917                        DW   CBLOCK
3801: 1349                                DO   EMUSIC
3801: 1349 15                          DB   EMUSIC+1
3801: 134A                                ENDM
3801:
3802:                                ; CLEAR JOYSTICKS
3803: 134A                                DO   FILL
3803: 134A 1B                          DB   FILL+1

```

Home Video Game System Library - Checkmate

```

3803: 134B          ENDM
3803:
3804: 134B E44F          DW   OSW0
3805: 134D 0400          DW   4
3806: 134F 00           DB   0
3807:                ;CLEAR ALL RAM DATA
3808: 1350          DO   FILL
3808: 1350 1B          DB   FILL+1
3808: 1351          ENDM
3808:
3809: 1351 A24F          DW   CNOPL
3810: 1353 2200          DW   (PLAY3+PSTAT)-CNOPL+1
3811: 1355 00           DB   0
3812: 1356          DO   SETOUT
3812: 1356 17          DB   SETOUT+1
3812: 1357          ENDM
3812:
3813: 1357 BE           DB   ((Y LINES*4)+LOWY)*2 ; VER BLK
3814: 1358 67           DB   67H+0 ; HOR COL BND
3815: 1359 08           DB   08H ; INTER MODE
3816:                ; CLEAR SCORE BLOCKS
3817: 135A          DO   RECTAN
3817: 135A 1D          DB   RECTAN+1
3817: 135B          ENDM
3817:
3818: 135B 0000          DW   0
3819: 135D A00B          DW   (11 SHL 8)+160
3820: 135F 55           DB   01010101B
3821: 1360          DO   RECTAN
3821: 1360 1D          DB   RECTAN+1
3821: 1361          ENDM
3821:
3822: 1361 7A00          DW   $007A
3823:                ; WRECK - CHANGED TO MATCH ROM
3824: 1363 1E09          DW   $091E
3825: 1365 00           DB   0000B
3826: 1366          DO   RECTAN
3826: 1366 1D          DB   RECTAN+1
3826: 1367          ENDM
3826:
3827: 1367 5700          DW   $0057
3828:                ; WRECK - CHANGED TO MATCH ROM
3829: 1369 1D09          DW   $091D
3830: 136B 00           DB   0000B
3831: 136C          DO   RECTAN
3831: 136C 1D          DB   RECTAN+1
3831: 136D          ENDM
3831:
3832: 136D 2B00          DW   $002B
3833:                ; WRECK - CHANGED TO MATCH ROM
3834: 136F 1D09          DW   $091D
3835: 1371 00           DB   0000B
3836: 1372          DO   RECTAN
3836: 1372 1D          DB   RECTAN+1
3836: 1373          ENDM
3836:
3837: 1373 0400          DW   $0004

```

Home Video Game System Library - Checkmate

```

3838:                                ; WRECK - CHANGED TO MATCH ROM
3839: 1375 2109                        DW  $0921
3840: 1377 00                          DB  0000B
3841: 1378                                DO  ACTINT
3841: 1378 0F                          DB  ACTINT+1
3841: 1379                                ENDM
3841:
3842: 1379                                EXIT
3842: 1379 02                          DB  XINTC
3842: 0000                                INTPCC DEF 0
3842: 137A                                ENDM
3842:
3843:                                ; INITIALIZE STARTING ADDRESS OF ARROWS
3844: 137A 212833                        LD  HL,(YTAB2 SHL 8)+XTAB1
3845: 137D 22AC4F                        LD  (PLAY0+ARRX),HL
3846: 1380 217833                        LD  HL,(YTAB2 SHL 8)+XTAB3
3847: 1383 22B34F                        LD  (PLAY1+ARRX),HL
3848: 1386 21501F                        LD  HL,(YTAB1 SHL 8)+XTAB2
3849: 1389 22BA4F                        LD  (PLAY2+ARRX),HL
3850: 138C 215047                        LD  HL,(YTAB3 SHL 8)+XTAB2
3851: 138F 22C14F                        LD  (PLAY3+ARRX),HL
3852:                                ; CLEAR FIELD
3853: 1392 CDB314                        CALL CLEARF
3854:                                ; DISPLAY # GAMES
3855: 1395 DD210D02                      LD  IX,FNTSML
3856: 1399                                SYSSUK DISNUM
3856: 1399 FF                            RST  56
3856: 139A 37                            DB  DISNUM+1
3856: 0000                                IF  DISNUM = INTPC
3856:                                ENDIF
3856: 139B                                ENDM
3856:
3857: 139B 4C                            DB  76
3858: 139C 02                            DB  2
3859: 139D 24                            DB  TDOPT
3860: 139E 42                            DB  42H
3861: 139F DC4F                          DW  CT7
3862: 13A1                                DONTD:
3863:                                ; GET # HUMANS
3864: 13A1 3AF34F                        LD  A,(NUMPLY)
3865: 13A4 FE05                          CP  5
3866: 13A6 3802                          JR  C,$+(GOTNPL-$)
3867: 13A8 3E04                          LD  A,4
3868: 13AA                                GOTNPL:
3869: 13AA 32A44F                        LD  (CNOHUM),A
3870:                                ; GET # PLAYERS
3871:                                ; IF HUMANS=1 OR 0 OR > 4 THEN PLAYERS=4 ELSE PLA
3872: 13AD FE02                          CP  2
3873: 13AF 3804                          JR  C,$+(FPLAY-$)
3874: 13B1 FE05                          CP  5
3875: 13B3 3802                          JR  C,$+(ALLHUM-$)
3876: 13B5 3E04                          FPLAY: LD  A,4
3877: 13B7 32A24F                        ALLHUM: LD (CNOPL),A
3878:                                ; INITIALIZE THE PLAYER PACKETS
3879:                                ; B=CURR # HUMANS
3880:                                ; C=CURR # PLAYERS
3881:                                ; D=THIS PLAYER #

```

Home Video Game System Library - Checkmate

```

3882: 13BA 3AA44F      INTIPP: LD    A,(CNOHUM)
3883: 13BD 47          LD    B,A
3884: 13BE 3AA24F      LD    A,(CNOPL)
3885: 13C1 4F          LD    C,A
3886: 13C2 1600        LD    D,0
3887: 13C4 7A          GTPLIX: LD    A,D
3888: 13C5 CD5B16        CALL LDPLIX
3889: 13C8 C5          PUSH BC
3890: 13C9 D5          PUSH DE
3891: 13CA 7A          LD    A,D
3892: 13CB C631        ADD  A,$31          ;SETUP ASCII LITERAL
3893: 13CD DD5E04        LD    E,(IX+ARRX)
3894: 13D0 DD5605        LD    D,(IX+ARRY)  ;??? IS THIS RIGHT?
3895: 13D3 1D          DEC  E
3896: 13D4 1D          DEC  E
3897: 13D5 FD4E06        LD    C,(IY+PCDOP)
3898: 13D8          SYSTEM CHRDIS      ;DISPLAY PATTERN # ON FIELD
3898: 13D8 FF          RST   56
3898: 13D9 32          DB    CHRDIS
3898: 0000          IF    CHRDIS = INTPC
3898:          ENDIF
3898: 13DA          ENDM
3898:
3899: 13DA FD5E07        LD    E,(IY+PSPOX)
3900: 13DD FD5608        LD    D,(IY+PSPOX)
3901: 13E0 D5          PUSH DE
3902: 13E1          SYSTEM CHRDIS      ;DISPLAY# ON SCORE BLOCK
3902: 13E1 FF          RST   56
3902: 13E2 32          DB    CHRDIS
3902: 0000          IF    CHRDIS = INTPC
3902:          ENDIF
3902: 13E3          ENDM
3902:
3903: 13E3 D1          POP  DE
3904: 13E4 7B          LD    A,E
3905: 13E5 C606        ADD  A,6
3906: 13E7 5F          LD    E,A
3907: 13E8 14          INC  D
3908: 13E9 14          INC  D
3909: 13EA 010104        LD    BC,(PATYSZ SHL 8) + PATXSZ
3910: 13ED FD6605        LD    H,(IY+PPATH)
3911: 13F0 FD6E04        LD    L,(IY+PPATL)
3912: 13F3 3E10        LD    A,00010000B
3913: 13F5          SYSTEM WRIT ;WRIT PLAYER PAT IN SCORE BLOCK
3913: 13F5 FF          RST   56
3913: 13F6 24          DB    WRIT
3913: 0000          IF    WRIT = INTPC
3913:          ENDIF
3913: 13F7          ENDM
3913:
3914: 13F7 D1          POP  DE
3915: 13F8 D5          PUSH DE
3916: 13F9 DDE5        PUSH IX
3917: 13FB 7A          LD    A,D
3918: 13FC 0600        LD    B,0
3919: 13FE 4A          LD    C,D
3920: 13FF 21964F        LD    HL,CURSCR

```

Home Video Game System Library - Checkmate

```

3921: 1402 09          ADD HL,BC
3922: 1403 09          ADD HL,BC
3923: 1404 09          ADD HL,BC
3924: 1405 CDE215      CALL DISPSC          ; DISPLAY SCORES
3925: 1408 DDE1        POP IX
3926: 140A D1         POP DE
3927: 140B C1         POP BC
3928: 140C AF         XOR A
3929: 140D B0         OR B
3930: 140E 2809      JR Z,$+(NOTHUM-$)
3931: 1410 3EC0      LD A,ACTIVE+HUMAN
3932: 1412 DD7706    LD (IX+PSTAT),A
3933: 1415 05         DEC B
3934: 1416 1806      JR $+(CKNOPL-$)
3935: 1418 00         CKSUM3: DB 0
3936: 1419 3E80      NOTHUM: LD A,ACTIVE
3937: 141B DD7706    LD (IX+PSTAT),A
3938: 141E 14         CKNOPL: INC D
3939: 141F 0D         DEC C
3940: 1420 AF         XOR A
3941: 1421 B1         OR C
3942: 1422 20A0      JR NZ,$+(GTPLIX-$)
3943: 1424 3E03      LD A,3
3944: 1426          CDOWNL:
3945: 1426 F5          PUSH AF
3946: 1427          SYSSUK PAWS
3946: 1427 FF          RST 56
3946: 1428 51          DB PAWS+1
3946: 0000          IF PAWS = INTPC
3946:          ENDIF
3946: 1429          ENDM
3946:
3947: 1429 05          DB 5
3948: 142A 32A34F      LD (PLIX),A
3949: 142D CD9014      CALL UPMUZK          ; MAKE SOUND FOR COUNTDOWN
3950: 1430 F1          POP AF
3951: 1431 F5          PUSH AF
3952: 1432 C630        ADD A,30H
3953: 1434          XYRELL DE,(XTAB2-4),(YTAB2-4)
3953: 1434 114C2F      LD DE,[(YTAB2-4) SHL 8)+(XTAB2-4)]
3953: 1437          ENDM
3953:
3954: 1437 0E44        LD C,CDOPT
3955: 1439          SYSTEM CHRDIS          ; DISPLAY COUNTDOWN #
3955: 1439 FF          RST 56
3955: 143A 32          DB CHRDIS
3955: 0000          IF CHRDIS = INTPC
3955:          ENDIF
3955: 143B          ENDM
3955:
3956: 143B          SYSSUK PAWS
3956: 143B FF          RST 56
3956: 143C 51          DB PAWS+1
3956: 0000          IF PAWS = INTPC
3956:          ENDIF
3956: 143D          ENDM
3956:

```

Home Video Game System Library - Checkmate

```

3957: 143D 28          DB 40
3958: 143E          SYSTEM EMUSIC
3958: 143E FF          RST 56
3958: 143F 14          DB EMUSIC
3958: 0000          IF EMUSIC = INTPC
3958:                ENDIF
3958: 1440          ENDM
3958:
3959: 1440 F1          POP AF
3960: 1441 3D          DEC A
3961: 1442 20E2        JR NZ,$+(CDOWNL-$)
3962: 1444 CDB314        CALL CLEARF
3963:                ; INT TICK COUNT
3964: 1447 CD4916        CALL TICKIT
3965: 144A AF          XOR A
3966: 144B 32DD4F        LD (CNT),A
3967: 144E          LOOPY:
3968: 144E          SYSSUK SENTRY
3968: 144E FF          RST 56
3968: 144F 43          DB SENTRY+1
3968: 0000          IF SENTRY = INTPC
3968:                ENDIF
3968: 1450          ENDM
3968:
3969: 1450 1402          DW ALKEYS
3970: 1452          SYSSUK DOIT
3970: 1452 FF          RST 56
3970: 1453 45          DB DOIT+1
3970: 0000          IF DOIT = INTPC
3970:                ENDIF
3970: 1454          ENDM
3970:
3971: 1454 5814          DW THETBL
3972: 1456 18F6          JR $+(LOOPY-$)
3973: 1458          THETBL: RC SCT0,ACTION,0
3973: 1458 41          DB SCT0+$40
3973: 1459 6B14          DW ACTION
3973: 0000          IF 0 = $C0
3973:                ENDIF
3973: 145B          ENDM
3973:
3974: 145B          RC SJ0,MOVJOY,0
3974: 145B 55          DB SJ0+$40
3974: 145C 8314          DW MOVJOY
3974: 0000          IF 0 = $C0
3974:                ENDIF
3974: 145E          ENDM
3974:
3975: 145E          RC SJ1,MOVJOY,0
3975: 145E 57          DB SJ1+$40
3975: 145F 8314          DW MOVJOY
3975: 0000          IF 0 = $C0
3975:                ENDIF
3975: 1461          ENDM
3975:
3976: 1461          RC SJ2,MOVJOY,0
3976: 1461 59          DB SJ2+$40

```

Home Video Game System Library - Checkmate

```

3976: 1462 8314          DW    MOVJOY
3976: 0000              IF    0 = $C0
3976:                  ENDIF
3976: 1464              ENDM
3976:
3977: 1464              RC    SJ3,MOVJOY,0
3977: 1464 5B           DB    SJ3+$40
3977: 1465 8314         DW    MOVJOY
3977: 0000              IF    0 = $C0
3977:                  ENDIF
3977: 1467              ENDM
3977:
3978: 1467              RC    SKYD,CALPIZ,ENDX
3978: 1467 53           DB    SKYD+$40
3978: 1468 8A14         DW    CALPIZ
3978: 0001              IF    ENDX = $C0
3978: 146A C0           DB    ENDX
3978:                  ENDIF
3978: 146B              ENDM
3978:
3979:
3980: 146B              ACTION:
3981: 146B CD4916         CALL TICKIT
3982:                  ; INCREMENT THE CURRENT PLAYER INDEX BY 1 UNTIL
3983:                  ; AN ACTIVE PLAYER IS FOUND THEN UPDATE HIM
3984: 146E 3AA34F       INCIX: LD    A,(PLIX)
3985: 1471 3C           INC    A
3986: 1472 E603         AND    03H
3987: 1474 32A34F       LD    (PLIX),A      ; CURR PLAYER IX<-CURR PL IX+1 M
3988: 1477 CD5B16       CALL LDPLIX
3989: 147A DDCB067E     BIT    ACTBIT,(IX+PSTAT) ; TEST FOR ACTIVE PLAYER
3990: 147E 28EE         JR    Z,$+(INCIX-$)
3991: 1480 C3BB14       JP    MOVEIT        ; THE MAJOR EVENT
3992: 1483              MOVJOY:
3993: 1483 D615         SUB    SJ0          ; TAKE OFF WHATEVER
3994: 1485 CB3F         SRL    A            ; DIVIDE BY 2
3995: 1487 C31D16       JP    STALL
3996: 148A              CALPIZ:
3997: 148A CD4916         CALL TICKIT
3998: 148D              SYSTEM PIZBRK
3998: 148D FF           RST    56
3998: 148E 48           DB    PIZBRK
3998: 0000              IF    PIZBRK = INTPC
3998:                  ENDIF
3998: 148F              ENDM
3998:
3999: 148F C9           RET
4000: 1490 3AA34F       UPMUZK: LD    A,(PLIX)
4001: 1493 CD5B16       CALL LDPLIY
4002: 1496 DD7E03       LD    A,(IX+AROT)
4003: 1499 0603         LD    B,3
4004: 149B              TSTBIT:
4005: 149B 0F           RRCA
4006: 149C 3802         JR    C,$+(GOTBIT-$)
4007: 149E 10FB         DJNZ  $+(TSTBIT-$)
4008: 14A0              GOTBIT:
4009: 14A0 48           LD    C,B

```

Home Video Game System Library - Checkmate

```

4010: 14A1 0600          LD    B,0
4011: 14A3 FD09          ADD   IY,BC
4012: 14A5 FD7E00       LD    A,(IY+0)
4013: 14A8 D313          OUT   (TONEC),A
4014: 14AA 3E09          LD    A,MUSVOL
4015: 14AC D315          OUT   (VOLC),A
4016: 14AE 3E11          LD    A,OA4
4017: 14B0 D310          OUT   (TONMO),A
4018: 14B2 C9           RET
4019: 14B3          CLEARF:
4020:          ; CLEAR FIELD
4021: 14B3          SYSSUK FILL
4021: 14B3 FF          RST   56
4021: 14B4 1B          DB    FILL+1
4021: 0000          IF    FILL = INTPC
4021:          ENDIF
4021: 14B5          ENDM
4021:
4022: 14B5 B841          DW   STARTS
4023: 14B7 200D         DW   ALLBYT
4024: 14B9 00          DB    0
4025: 14BA C9           RET
4026: 14BB          MOVEIT:
4027:          ; THIS ROUTINE UPDATES A PLAYER'S POSITION
4028:          ; INPUT PARAS ARE: IX=POINTER TO PLAYERS PACKET
4029:          ; DURING ROUTINE B=CURRENT SWITCH C=LAST SWITCH
4030: 14BB DD4E00       LD    C,(IX+LASTSW)
4031: 14BE DD4602       LD    B,(IX+CURSW)
4032: 14C1 DDCB0676     BIT   HUMBIT,(IX+PSTAT)
4033: 14C5 2003         JR    NZ,$+(NOCUR-$) ; IF NOT HUMAN
4034: 14C7 AF          ZSW:  XOR   A          ; CLEAR A
4035: 14C8 47          LD    B,A          ; CLEAR CURRENT SWITCH
4036: 14C9 4F          LD    C,A          ; CLEAR LAST SWITCH ENDIF
4037: 14CA 78          NOCUR: LD    A,B          ; IF CURR SW = 0
4038: 14CB B7          OR    A
4039: 14CC 2001         JR    NZ,$+(RANTST-$)
4040: 14CE 41          LD    B,C          ; THEN CURR SW<-LAST SW ENDIF
4041: 14CF DD7000       RANTST: LD   (IX+LASTSW),B ; SAVE LAST SW
4042: 14D2 78          LD    A,B          ; IF CURR SW=0
4043: 14D3 B7          OR    A
4044: 14D4 2005         JR    NZ,$+(GOTSW-$)
4045: 14D6 0E00        LD    C,0          ; LAST SW<-0
4046: 14D8 CD7E16      CALL  RANMOV       ; GET RANDOM MOVE ENDIF
4047: 14DB          GOTSW:
4048: 14DB DD7E01       LD    A,(IX+LASTMV) ; GET LAST MOVE
4049: 14DE CDAB16      CALL  MOVST
4050: 14E1 2813         JR    Z,$+(GOTMOV-$)
4051:          ; ANY MOVE AND CURR SW
4052: 14E3 CDA916      CALL  MOVANY
4053: 14E6 280E         JR    Z,$+(GOTMOV-$)
4054: 14E8 41          LD    B,C          ; TRY LAST SW
4055:          ; ANY MOVE
4056: 14E9 CDA916      CALL  MOVANY
4057: 14EC 2808         JR    Z,$+(GOTMOV-$)
4058: 14EE DD4601       LD    B,(IX+LASTMV) ; TRY LAST MOVE
4059:          ; ANY MOVE
4060: 14F1 CDA916      CALL  MOVANY

```

Home Video Game System Library - Checkmate

```

4061: 14F4 203C          JR    NZ,$+(CRASH-$)
4062: 14F6          GOTMOV:
4063:          ; A LEGIT MOVE HAS BEEN FOUND SO UPDATE THE GUY
4064: 14F6 DD7701          LD    (IX+LASTMV),A ; SAVE ACTUAL MOVE FOR LATER
4065: 14F9 DD7703          LD    (IX+AROT),A ;ARROW ROTATION AMOUNT<- THE MOV
4066: 14FC DD5605          LD    D,(IX+ARRY)
4067: 14FF DD5E04          LD    E,(IX+ARRX)
4068: 1502 CD2415          CALL ERASE
4069: 1505 FD6605          LD    H,(IY+PPATH)
4070: 1508 FD6E04          LD    L,(IY+PPATL)
4071: 150B 010104         LD    BC,(PATYSZ SHL 8) + PATXSZ
4072: 150E 3E10          LD    A,WRITOR
4073: 1510          SYSTEM WRIT      ; WRITE PLAYER PATTERN OVER ARRO
4073: 1510 FF          RST    56
4073: 1511 24          DB     WRIT
4073: 0000          IF     WRIT = INTPC
4073:          ENDIF
4073: 1512          ENDM
4073:
4074: 1512 3AA54F         LD    A,(TARRX)
4075: 1515 DD7704          LD    (IX+ARRX),A ;SAVE NEW ARROW X
4076: 1518 3AA64F         LD    A,(TARRY)
4077: 151B DD7705          LD    (IX+ARRY),A ;SAVE NEW ARROW Y
4078: 151E CDFF15         CALL ANIARR      ;ANIMATE THE ARROW
4079: 1521 C39014         JP    UPMUZK
4080: 1524          ERASE:
4081: 1524 D5          PUSH DE
4082: 1525          SYSSUK RELAB1
4082: 1525 FF          RST    56
4082: 1526 3B          DB     RELAB1+1
4082: 0000          IF     RELAB1 = INTPC
4082:          ENDIF
4082: 1527          ENDM
4082:
4083: 1527 00          DB    0
4084: 1528 EB          EX    DE,HL
4085: 1529 0600         LD    B,0
4086: 152B 110104         LD    DE,[(PATYSZ SHL 8) + PATXSZ]
4087: 152E          SYSTEM BLANK
4087: 152E FF          RST    56
4087: 152F 2A          DB     BLANK
4087: 0000          IF     BLANK = INTPC
4087:          ENDIF
4087: 1530          ENDM
4087:
4088: 1530 D1          POP   DE
4089: 1531 C9          RET
4090: 1532          CRASH:
4091:          ; A PLAYER HAS CRASHED. DESTROY HIS ARROW AND
4092:          ; ELIMINATE HIM FROM THE GAME.
4093: 1532 016C17         LD    BC,EXPATS
4094: 1535 118017         LD    DE,EXCOLS ;DE<-EXPLODE COLOR TABLE ADDR
4095: 1538 3E05          LD    A,5
4096: 153A 21B117         LD    HL,EXPSND
4097: 153D F5          EXLOOP: PUSH AF      ;PUSH LOOP COUNT
4098: 153E C5          PUSH BC      ;PUSH EXT PAT ADDRESS
4099: 153F D5          PUSH DE      ;PUSH EXPLODE COLOR TABLE ADDR

```

Home Video Game System Library - Checkmate

```

4100: 1540 E5          PUSH HL          ;PUSH EXPLODE SOUND ADDRESS
4101: 1541 1A          LD  A,(DE)      ;A<-EXPLODE COLOR
4102: 1542 D304        OUT  (COL0L),A   ; CHANGED TO MATCH ROM
4103: 1544 C5          PUSH BC
4104: 1545 DD5605       LD  D,(IX+ARRY)
4105: 1548 DD5E04       LD  E,(IX+ARRX)
4106: 154B CD2415       CALL ERASE
4107: 154E E1          POP  HL          ;PATTERN ADDRESS
4108: 154F 3E10        LD  A,WRITOR
4109: 1551 010104      LD  BC,(PATYSZ SHL 8) + PATXSZ
4110: 1554          SYSTEM WRIT      ;WRIT EXPLOSION
4110: 1554 FF          RST  56
4110: 1555 24          DB   WRIT
4110: 0000          IF   WRIT = INTPC
4110:          ENDIF
4110: 1556          ENDM
4110:
4111: 1556          SYSSUK PAWS
4111: 1556 FF          RST  56
4111: 1557 51          DB   PAWS+1
4111: 0000          IF   PAWS = INTPC
4111:          ENDIF
4111: 1558          ENDM
4111:
4112: 1558 07          DB   7
4113: 1559 E1          POP  HL          ;GET EXPLODE SOUND ADDR
4114: 155A 011808      LD  BC,(SBLLEN SHL 8) + SNDBX
4115: 155D EDB3        OTIR
4116: 155F D1          POP  DE
4117: 1560 C1          POP  BC
4118: 1561 F1          POP  AF
4119: 1562 3D          DEC  A
4120: 1563 2807        JR  Z,$+(EXPFIN-$) ;LOOP COUNT EXPIRED
4121: 1565 13          INC  DE          ;INC TO NEXT COLOR
4122: 1566 03          INC  BC          ;BUMP UP TO NEXT PAT ADDR
4123: 1567 03          INC  BC
4124: 1568 03          INC  BC
4125: 1569 03          INC  BC
4126: 156A 18D1        JR  $+(EXLOOP-$)
4127: 156C          EXPFIN:
4128: 156C FD5605       LD  D,(IY+PPATH)
4129: 156F FD5E04       LD  E,(IY+PPATL) ;DE<-PLAYER PAT ADDR
4130: 1572 FD210000    LD  IY,0
4131: 1576 FD19        ADD  IY,DE       ;IY<-PLAYER PAT ADDR
4132: 1578 110004      LD  DE,4 SHL 8 + 0 ;D<-LOOP COUNT
4133: 157B FD7E00      STOMP: LD  A,(IY+0) ;A<-BYTE OF PLAYER PATTERN
4134: 157E 21B841      LD  HL,STARTS
4135: 1581 01200D      LD  BC,ALLBYT
4136: 1584 EDB1        STLOOP: CPIR
4137: 1586 2005        JR  NZ,$+(RESTOM-$)
4138: 1588 03          INC  BC
4139: 1589 2B          DEC  HL
4140: 158A 73          LD  (HL),E
4141: 158B 18F7        JR  $+(STLOOP-$)
4142: 158D FD23        RESTOM: INC  IY
4143: 158F 15          DEC  D
4144: 1590 20E9        JR  NZ,$+(STOMP-$)

```

Home Video Game System Library - Checkmate

```

4145: 1592 DDCB0676          BIT  HUMBIT,(IX+PSTAT) ;
4146: 1596 2804             JR   Z,$+(KILLST-$)   ;IF HUMAN
4147: 1598 21A44F           LD   HL,CNOHUM
4148: 159B 35                DEC  (HL)              ;DEC CURRENT # HUMANS END IF
4149: 159C DDCB06BE          KILLST: RES  ACTBIT,(IX+PSTAT) ;KILL STATUS
4150:                          ; INC ALL ACTIVE PLAYERS SCORES
4151: 15A0 0E04             LD   C,4
4152: 15A2                          BUMPEM:
4153: 15A2 0D                DEC  C
4154: 15A3 79                LD   A,C
4155: 15A4 CD5B16           CALL LDPLIX
4156: 15A7 DDCB067E          BIT  ACTBIT,(IX+PSTAT)
4157: 15AB 2818             JR   Z,$+(BUMPCK-$)
4158: 15AD 0600             LD   B,0
4159: 15AF C5                PUSH BC
4160: 15B0 79                LD   A,C
4161: 15B1 21964F           LD   HL,CURSCR
4162: 15B4 09                ADD  HL,BC
4163: 15B5 09                ADD  HL,BC
4164: 15B6 09                ADD  HL,BC
4165: 15B7 37                SCF
4166: 15B8 CDE215           CALL DISPSC
4167: 15BB C1                POP  BC
4168: 15BC                          SYSTEM INCSCR
4168: 15BC FF                RST  56
4168: 15BD 54                DB   INCSCR
4168: 0000                  IF   INCSCR = INTPC
4168:                          ENDIF
4168: 15BE                          ENDM
4168:
4169: 15BE 79                LD   A,C
4170: 15BF C5                PUSH BC
4171: 15C0 B7                OR   A
4172: 15C1 CDE215           CALL DISPSC
4173: 15C4 C1                POP  BC
4174: 15C5                          BUMPCK:
4175: 15C5                  SYSSUK PAWS
4175: 15C5 FF                RST  56
4175: 15C6 51                DB   PAWS+1
4175: 0000                  IF   PAWS = INTPC
4175:                          ENDIF
4175: 15C7                          ENDM
4175:
4176: 15C7 1E                DB   30
4177: 15C8 79                LD   A,C
4178: 15C9 B7                OR   A
4179: 15CA 20D6             JR   NZ,$+(BUMPEM-$)
4180:                          ;DEC CURR # PLAYERS
4181:                          ;IF CURR # PLAYERS LEQ GO TO END OF GAME
4182: 15CC 21A24F           LD   HL,CNOPL
4183: 15CF 35                DEC  (HL)
4184: 15D0 35                DEC  (HL)
4185: 15D1 2802             JR   Z,$+(ENDCHK-$)
4186: 15D3 34                INC  (HL)
4187: 15D4 C9                RET
4188: 15D5                          ENDCHK:
4189: 15D5 3ADC4F           LD   A,(CT7)

```

Home Video Game System Library - Checkmate

```

4190: 15D8 3D          DEC  A
4191: 15D9 27          DAA
4192: 15DA 32DC4F      LD   (CT7),A
4193: 15DD C24013      JP   NZ,FIREIT
4194: 15E0              SYSTEM QUIT
4194: 15E0 FF          RST  56
4194: 15E1 78          DB   QUIT
4194: 0000              IF   QUIT = INTPC
4194:                ENDIF
4194: 15E2              ENDM
4194:
4195: 15E2          DISPSC:
4196:                ;DISPLAY SCORE
4197:                ;A=PLAYER #
4198:                ;HL->LAST BYTE OF SCORE
4199: 15E2 FD4E09      LD   C,(IY+PSDOP)
4200: 15E5 3004      JR   NC,$+(NOTXOR-$)
4201: 15E7 CBA1      RES  MROR,C
4202: 15E9 CBE9      SET  MRXOR,C
4203: 15EB          NOTXOR:
4204: 15EB FD5E07      LD   E,(IY+PSPOX)
4205: 15EE FD5608      LD   D,(IY+PSPOSY)
4206: 15F1 3E0C      LD   A,12
4207: 15F3 83        ADD  A,E
4208: 15F4 5F        LD   E,A
4209: 15F5 14        INC  D
4210: 15F6 0643      LD   B,$43
4211: 15F8 DD210D02    LD   IX,FNTSML
4212: 15FC          SYSTEM DISNUM
4212: 15FC FF          RST  56
4212: 15FD 36        DB   DISNUM
4212: 0000          IF   DISNUM = INTPC
4212:                ENDIF
4212: 15FE          ENDM
4212:
4213: 15FE C9          RET
4214: 15FF          ANIARR:
4215:                ;ANIMATE THE ARROW
4216:                ;INPUT AND OUTPUT IS IX WHO STAYS THE SAME
4217:                ;DESTROYS ALL OTHER REGISTERS
4218: 15FF DDCB067E    BIT  ACTBIT,(IX+PSTAT)
4219: 1603 C8          RET  Z           ;EXIT IF NOT ACTIVE
4220: 1604 DD7E03      LD   A,(IX+AROT)
4221: 1607 CD2E17      CALL GETROT      ;HL<-ARROW PAT ADDR
4222: 160A DD5605      LD   D,(IX+ARRY)
4223: 160D DD5E04      LD   E,(IX+ARRX)
4224: 1610 E5          PUSH HL
4225: 1611 CD2415      CALL ERASE
4226: 1614 E1          POP  HL
4227: 1615 010104     LD   BC,(PATYSZ SHL 8) + PATXSZ
4228: 1618 3E10      LD   A,WRITOR
4229: 161A          SYSTEM WRIT
4229: 161A FF          RST  56
4229: 161B 24        DB   WRIT
4229: 0000          IF   WRIT = INTPC
4229:                ENDIF
4229: 161C          ENDM

```

Home Video Game System Library - Checkmate

```

4229:
4230: 161C C9          RET
4231: 161D          STALL:
4232:                ;THIS ROUTINE TAKES CARE OF ARROW ANIMATION
4233:                ;AND SHOWING A PLAYER HIS CURRENT JOYSTICK POSIT
4234:                ;A=WHICH PLAYER
4235:                ;B=JOYSTICK BITS
4236: 161D CD5B16      CALL LDPLIX          ;IX<-ADDR OF PLAYER PACKET
4237: 1620 AF          XOR A
4238: 1621 B0         OR B
4239: 1622 2003       JR NZ,$+(STORIT-$)
4240: 1624 DD7E02     LD A,(IX+CURSW)
4241: 1627 DD7702     STORIT: LD (IX+CURSW),A
4242: 162A DDAE01     XOR (IX+LASTMV)    ;A<-DIFFERENCE FROM LAST MOVE
4243: 162D 2812       JR Z,$+(GETLM-$)  ;IF DIFFERENCE=0 USE LAST MOVE
4244: 162F EE0C       XOR RLMOVE
4245: 1631 280E       JR Z,$+(GETLM-$)
4246: 1633 EE0C       XOR RLMOVE
4247: 1635 EE03       XOR UDMOVE
4248: 1637 2808       JR Z,$+(GETLM-$)
4249: 1639 EE03       XOR UDMOVE
4250: 163B DDCB0676   HUMCHK: BIT HUMBIT,(IX+PSTAT)
4251: 163F 2003       JR NZ,$+(GOTIT-$) ;IF HUMAN WE'VE GOT IT
4252: 1641 DD7E01     GETLM LD A,(IX+LASTMV) ;GET LAST MOVE
4253: 1644 DD7703     GOTIT: LD (IX+AROT),A ;STORE ARROW ROTATION
4254: 1647 18B6       JR $+(ANIARR-$)
4255: 1649          TICKIT:
4256:                ;TICK COUNT<-(8-CURR # PLAYERS)
4257: 1649 3AA44F     LD A,(CNOHUM)
4258: 164C B7         OR A
4259: 164D 3E02       LD A,2
4260: 164F 2806       JR Z,$+(STICK-$)
4261: 1651 21A24F     LD HL,CNOPL
4262: 1654 3E08       LD A,8
4263: 1656 96         SUB (HL)
4264: 1657 32D54F     STICK: LD (CT0),A
4265: 165A C9         RET
4266: 165B          LDPLIX:
4267: 165B          LDPLIY:
4268:                ;LOAD IY WITH POINTER TO CURR PLAYER ROM DATA
4269:                ;LOAD IX WITH A POINTER TO CURRENT PLAYER PACKET
4270:                ;A=PLAYER# MUST BE GEQ 0 $ LEQ 3
4271: 165B D5         PUSH DE
4272: 165C E5         PUSH HL
4273: 165D          SYSSUK INDEXW
4273: 165D FF        RST 56
4273: 165E 5B        DB INDEXW+1
4273: 0000          IF INDEXW = INTPC
4273:          ENDIF
4273: 165F          ENDM
4273:
4274: 165F 6E16      DW ROMTBL
4275: 1661 D5         PUSH DE
4276: 1662 FDE1      POP IY
4277: 1664          SYSSUK INDEXW
4277: 1664 FF        RST 56
4277: 1665 5B        DB INDEXW+1

```

Home Video Game System Library - Checkmate

```

4277: 0000          IF      INDEXW = INTPC
4277:                ENDIF
4277: 1666          ENDM
4277:
4278: 1666  7616          DW      RAMTBL
4279: 1668  D5            PUSH  DE
4280: 1669  DDE1          POP   IX
4281: 166B  E1            POP   HL
4282: 166C  D1            POP   DE
4283: 166D  C9            RET
4284: 166E  4417          ROMTBL: DW      PLROM0
4285: 1670  4E17          DW      PLROM1
4286: 1672  5817          DW      PLROM2
4287: 1674  6217          DW      PLROM3
4288: 1676  A84F          RAMTBL: DW      PLAY0
4289: 1678  AF4F          DW      PLAY1
4290: 167A  B64F          DW      PLAY2
4291: 167C  BD4F          DW      PLAY3
4292: 167E          RANMOV:
4293:                ;GENERATE A RANDOM MOVE FOR THE PLAYER PACKET POI
4294:                ;INPUT AND OUTPUT:
4295:                ;B=CURRENT SWITCH C=LAST SWITCH
4296: 167E          SYSSUK(RANGED)
4296: 167E  FF          RST   56
4296: 167F  77          DB    (RANGED)+1
4296: 0000          IF      (RANGED) = INTPC
4296:                ENDIF
4296: 1680          ENDM
4296:
4297: 1680  20            DB    32
4298: 1681  B7            OR    A                ;TIME TO CHANGE DIRECTION?
4299: 1682  2808          JR    Z,$+(NEWMOV-$)
4300: 1684  DD4601        LD    B,(IX+LASTMV) ;USE LAST MOVE
4301: 1687  78            LD    A,B
4302: 1688  CDAB16        CALL MOVST
4303: 168B  C8            RET   Z                ;LAST MOVE IS GOOD ENOUGH
4304: 168C          NEWMOV: SYSSUK RANGED
4304: 168C  FF          RST   56
4304: 168D  77          DB    RANGED+1
4304: 0000          IF      RANGED = INTPC
4304:                ENDIF
4304: 168E          ENDM
4304:
4305: 168E  04            DB    4
4306: 168F  47            LD    B,A
4307: 1690  04            INC   B
4308: 1691  3E80          LD    A,$80
4309: 1693  07            SHFTIT: RLCA
4310: 1694  10FD          DJNZ $+(SHFTIT-$)
4311: 1696  47            LD    B,A
4312: 1697  3E08          RANFIN: LD    A,$8
4313: 1699  CDAB16        CALL MOVST
4314: 169C  2002          JR    NZ,$+(ANYMOV-$)
4315: 169E  47            LD    B,A
4316: 169F  C9            RET
4317: 16A0  060F          ANYMOV: LD    B,$F        ;TRY ALL MOVES
4318: 16A2  3E08          LD    A,$8

```

Home Video Game System Library - Checkmate

```

4319: 16A4 CDAB16          CALL MOVST
4320: 16A7 47              LD B,A
4321: 16A8 C9              RET
4322: 16A9                MOVANY:
4323: 16A9 3E08          LD A,AMOVE
4324: 16AB                MOVST:
4325:                    ;TEST THE NEW MOVE FOR VALIDITY
4326:                    ;THE INPUTS AND OUTPUTS:
4327:                    ;B=A SET ??? MOVES TO BE TESTED (IS UNCHANGED)
4328:                    ;C=UNCHANGED
4329:                    ;A=FIRST MOVE TO TEST,VALUE OF GOOD MOVE ON OUTPU
4330:                    ;DE,HL=RETURNED UNCHANGED
4331:                    ;D=# ROTATES
4332:                    ;Z FLAG=Z IF GOOD MOVE FOUND(A CONTAINS FIRST GOO
4333:                    ;Z FLAG=??? IF NO GOOD MOVES FOUND(IN B)
4334: 16AB D5              PUSH DE
4335: 16AC 1608          LD D,8          ;INIT # ROTATES
4336: 16AE 0F          ROTMSK: RRCA    ;ROTATE TO NEXT MOVE
4337: 16AF 5F          LD E,A
4338: 16B0 A0          AND B
4339: 16B1 CDBF16      CALL CHKMOV    ;CHECK MOVE
4340: 16B4 7B          LD A,E
4341: 16B5 2806          JR Z,$+(MOVEXT-$) ;FOUND ONE
4342: 16B7 15          DEC D          ;DEC # ROTATES
4343: 16B8 20F4          JR NZ,$+(ROTMSK-$)
4344: 16BA 37          SCF          ;NO GOOD MOVES
4345: 16BB CB12          RL D          ;SET Z FLAG=NZ
4346: 16BD D1          MOVEXT: POP DE
4347: 16BE C9          RET
4348: 16BF                CHKMOV:
4349:                    ;CHECK THE MOVE IN A FOR BEING UNOCCUPIED
4350:                    ;INPUT AND OUTPUT:
4351:                    ;A=UP, DOWN, RIGHT OR LEFT BIT(RETURN UNCHANGED)
4352:                    ;Z FLAG=Z IF MOVE IN A IS TO AN EMPTY POSITION
4353:                    ;Z FLAG=NZ IF MOVE IN A IS BAD
4354:                    ;BC,DE,HL RETURNED UNTOUCHED
4355:                    ;IX=POINTER TO CURRENT PLAYER PACKET
4356:                    ;LOCAL TO THIS ROUTINE:
4357:                    ; D=TEMP X COORD OF ARROW
4358:                    ; E=TEMP Y COORD OF ARROW
4359: 16BF C5          PUSH BC
4360: 16C0 D5          PUSH DE
4361: 16C1 E5          PUSH HL
4362: 16C2 F5          PUSH AF
4363: 16C3 DD5604      LD D,(IX+ARRX) ;GET X COORD OF ARROW
4364: 16C6 DD5E05      LD E,(IX+ARRY) ;GET Y COORD OF ARROW
4365: 16C9 CB57          TLEFT: BIT CHLEFT,A
4366: 16CB 280A          JR Z,$+(TRIGHT-$)
4367: 16CD 7A          LD A,D          ;GET A LEFT MOVE
4368: 16CE FE00          CP LOWX
4369: 16D0 282F          JR Z,$+(BADMOV-$) ;ALREADY AT LOWEST X
4370: 16D2 D604          SUB WIDTH      ;DEC TEMP X BY 1 POSITION
4371: 16D4 57          LD D,A
4372: 16D5 1830          JR $+(LOOKSQ-$)
4373: 16D7 CB5F          TRIGHT: BIT CHRIGH,A
4374: 16D9 280A          JR Z,$+(TUP-$)
4375: 16DB 7A          LD A,D          ;GOT A RIGHT MOVE

```

Home Video Game System Library - Checkmate

```

4376: 16DC FE98          CP    XMAX
4377: 16DE 3021          JR    NC,$+(BADMOV-$) ;ALREADY GEQ MAX X
4378: 16E0 C604          ADD   A,WIDTH
4379: 16E2 57             LD    D,A
4380: 16E3 1822          JR    $+(LOOKSQ-$)
4381: 16E5 CB47          TUP:  BIT   CHUP,A
4382: 16E7 280A          JR    Z,$+(TDOWN-$)
4383: 16E9 7B             LD    A,E ;GOT AN UP MOVE
4384: 16EA FE0B          CP    LOWY
4385: 16EC 2813          JR    Z,$+(BADMOV-$) ;ALREADY AT LOWEST Y
4386: 16EE D604          SUB   HEIGHT ;DEC TEMP Y BY 1 POSITION
4387: 16F0 5F             LD    E,A
4388: 16F1 1814          JR    $+(LOOKSQ-$)
4389: 16F3 CB4F          TDOWN: BIT  CHDOWN,A
4390: 16F5 280A          JR    Z,$+(BADMOV-$)
4391: 16F7 7B             LD    A,E ;GOT A DOWN MOVE
4392: 16F8 FE5B          CP    YMAX
4393: 16FA 2805          JR    Z,$+(BADMOV-$) ;ALREADY AT HIGHEST Y
4394: 16FC C604          ADD   A,HEIGHT ;INC TEMP Y BY 1 POSITION
4395: 16FE 5F             LD    E,A
4396: 16FF 1806          JR    $+(LOOKSQ-$)
4397: 1701 F1             BADMOV: POP  AF
4398: 1702 37             SCF
4399: 1703 CB12          RL    D ;SET ??? FLAG = NZ
4400: 1705 1823          JR    $+(MOVEND-$)
4401: 1707
4402: ;SEE IF THE NEW SQUARE IS OCCUPIED
4403: 1707 D5             PUSH  DE
4404: 1708 D5             PUSH  DE
4405: 1709 C1             POP   BC
4406: 170A 51             LD    D,C ;REVERSE X,Y FOR SYSTEM
4407: 170B 58             LD    E,B
4408: 170C SYSSUK RELAB1
4408: 170C FF             RST   56
4408: 170D 3B             DB    RELAB1+1
4408: 0000 IF    RELAB1 = INTPC
4408: ENDIF
4408: 170E ENDM
4408:
4409: 170E 00             DB    0
4410: 170F E1             POP   HL
4411: 1710 EB             EX    DE,HL
4412: 1711 7E             LD    A,(HL)
4413: 1712 B7             OR    A ;TEST SQUARE
4414: 1713 20EC          JR    NZ,$+(BADMOV-$)
4415: 1715 012800        LD    BC,BYTEPL
4416: 1718 09             ADD   HL,BC
4417: 1719 7E             LD    A,(HL)
4418: 171A B7             OR    A
4419: 171B 20E4          JR    NZ,$+(BADMOV-$)
4420: 171D 7A             LD    A,D
4421: 171E 32A54F        LD    (TARRX),A ;STORE TEMP ARROW X COORD
4422: 1721 7B             LD    A,E
4423: 1722 32A64F        LD    (TARRY),A ;STORE TEMP ARROW Y COORD
4424: 1725 F1             POP   AF
4425: 1726 1600          LD    D,0
4426: 1728 CB3A          SRL   D ;SET Z FLAG=Z

```

Home Video Game System Library - Checkmate

```

4427: 172A E1      MOVEND: POP  HL
4428: 172B D1              POP  DE
4429: 172C C1              POP  BC
4430: 172D C9              RET
4431: 172E          GETROT:
4432:                ;HL<-BASE ADDR OF ROTATED PATTERN
4433:                ;A<-DIRECTION OF ROTATION
4434:                ;IF A HAS MORE THAN 1 BIT SET THEN ONLY ONE IS US
4435: 172E 218917      LD    HL,AUP
4436: 1731 CB47        BIT   CHUP,A
4437: 1733 C0          RET   NZ
4438: 1734 219117      LD    HL,ADOWN
4439: 1737 CB4F        BIT   CHDOWN,A
4440: 1739 C0          RET   NZ
4441: 173A 218D17      LD    HL,ARIGHT
4442: 173D CB5F        BIT   CHRIGH,A
4443: 173F C0          RET   NZ
4444: 1740 219517      LD    HL,ALEFT
4445: 1743 C9          RET
4446:                ;START OF ROM DATA FOR EACH PLAYER.
4447:                ;CONTAINS 4 PLAYER NOTES, PLAYER PATTERN ADDR
4448:                ; , PLAYER CHAR DISP OPT
4449:                ;PLAYER SCORE DISP OPT
4450:                ;AND PLAYER SCORE POSITION
4451: 1744          PLROM0:
4452: 1744 FDEEE1D4      PNOTE0: DB   G0,GS0,A0,AS0
4453: 1748 9917        PPADR0: DW   PPAT0
4454: 174A 18         PCDOP0: DB   011000B
4455: 174B 0C         PSPOS0: DB   12           ; CHANGED TO MATCH ROM
4456: 174C 01         PSPOS0: DB   1
4457: 174D 18         PSDOP0: DB   011000B
4458: 174E          PLROM1:
4459: 174E C8BDB2A8      PNOTE1: DB   B0,C1,CS1,D1
4460: 1752 9D17        PPADR1: DW   PPAT1
4461: 1754 1C         PCDOP1: DB   011100B
4462: 1755 7D         PSPOS1: DB   125         ; CHANGED TO MATCH ROM
4463: 1756 01         PSPOS1: DB   1
4464: 1757 1C         PSDOP1: DB   011100B
4465: 1758          PLROM2:
4466: 1758 9F968D85      PNOTE2: DB   DS1,E1,F1,FS1
4467: 175C A117        PPADR2: DW   PPAT2
4468: 175E 1C         PCDOP2: DB   011100B
4469: 175F 2D01        PSPOS2: DW   45 + (1 SHL 8)
4470: 1761 1C         PSDOP2: DB   011100B
4471: 1762          PLROM3:
4472: 1762 7E77706A      PNOTE3: DB   G1,GS1,A1,AS1
4473: 1766 A517        PPADR3: DW   PPAT3
4474: 1768 18         PCDOP3: DB   011000B
4475: 1769 5901        PSPOS3: DW   $0159       ; CHANGED TO MATCH ROM
4476: 176B 18         PSDOP3: DB   011000B
4477:                ;EXPLOSION PATTERNS
4478: 176C          EXPATS:
4479: 176C 00141400      EXPAT1: DB   0,00010100B,00010100B,0
4480: 1770 00455100      EXPAT2: DB   0,01000101B,01010001B,0
4481: 1774 05400150      EXPAT3: DB   00000101B,01000000B,00000001B,01010000B
4482: 1778 11400144      EXPAT4: DB   00010001B,01000000B,00000001B,01000100B
4483: 177C 00000000      EXPAT5: DB   0,0,0,0

```

Home Video Game System Library - Checkmate

```

4484:                ;EXPLOSION COLORS
4485: 1780           EXCOLS:
4486: 1780 07             DB    7
4487: 1781 03             DB    3
4488: 1782 07             DB    7
4489: 1783 03             DB    3
4490: 1784 87             DB    $87           ; CHANGED TO MATCH ROM
4491:                ;COUNT DOWN DISPLAY PACKET
4492: 1785 0400          CDCOLR: DW    0100B + (0 SHL 8)
4493:                ;TIMER DISPLAY PACKET
4494: 1787 0180          TDPACK: DW    0001B + (10000000B SHL 8)
4495:                ;ARROW ANIMATION PATTERNS FOR EACH ROTATION
4496: 1789           AUP:
4497: 1789 14554100        DB    00010100B,01010101B,01000001B,0
4498: 178D           ARIGHT:
4499: 178D 14050514        DB    00010100B,00000101B,00000101B,00010100B
4500: 1791           ADOWN:
4501: 1791 00415514        DB    0,01000001B,01010101B,00010100B
4502: 1795           ALEFT:
4503: 1795 14505014        DB    00010100B,01010000B,01010000B,00010100B
4504:                ;PLAYER PATTERNS
4505: 1799 08A82A20        PPAT0: DB    00001000B,10101000B,00101010B,00100000B
4506: 179D FFC3C3FF        PPAT1: DB    11111111B,11000011B,11000011B,11111111B
4507: 17A1 0CFC3F30        PPAT2: DB    00001100B,11111100B,00111111B,00110000B
4508: 17A5 AA8282AA        PPAT3: DB    10101010B,10000010B,10000010B,10101010B
4509:                ;COLOR BOCK
4510: 17A9           CBLOCK:
4511: 17A9 B3              DB    $B3           ; CHANGED TO MATCH ROM
4512: 17AA 62              DB    $62           ; CHANGED TO MATCH ROM
4513: 17AB F8              DB    $F8
4514: 17AC 87              DB    $87           ; CHANGED TO MATCH ROM
4515: 17AD F8              DB    $F8           ; CHANGED TO MATCH ROM
4516: 17AE F8              DB    $F8           ; CHANGED TO MATCH ROM
4517: 17AF F8              DB    $F8
4518: 17B0 F8              DB    $F8           ; CHANGED TO MATCH ROM
4519:                ;EXPLOSION SOUNDS
4520: 17B1 EFFF3F00          EXPSND: DB $EF,$FF,$3F,$00,$FF,$FD,$F5,$F5
      FFFDF5F5
4521: 17B9 8FEE3E00          DB $8F,$EE,$3E,$00,$FF,$FD,$F5,$F5
      FFFDF5F5
4522: 17C1 4E883800          DB $4E,$88,$38,$00,$FF,$FD,$F5,$F5
      FFFDF5F5
4523: 17C9 48443400          DB $48,$44,$34,$00,$FF,$FD,$F5,$F5
      FFFDF5F5
4524: 17D1 00000000          DB 0,0,0,0,0,0,0,0
      00000000
4525:
4526:                ;
4527:                ;           END OF CHECKMATE PROGRAM
4528:                ;
4529:                ; FIVE FILLER BYTES BETWEEN CHECKMATE AND
4530:                ; GUNFIGHT TO MATCH 3159 ROM
4531: 17D9 FFFFFFFF          DB    0FFH, 0FFH, 0FFH, 0FFH, 0FFH
      FF

```

Home Video Game System Library - Gunfight

```

4532:          ;
4533:          ;
4534:          ; *****
4535:          ; * H V G   G U N F I G H T * ; VER. 1.0 - DEC. 15, 2003
4536:          ; *****
4537:          ;
4538:          ;
4539:          ; Program Name - Gun Fight
4540:          ;                   for the Bally Astrocade
4541:          ; Retyped by Adam Trionfo
4542:
4543: 17DE  C3E819          JP      INIT
4544:
4545:          ; *****
4546:          ; * GUN FIGHT EQUATES *
4547:          ; *****
4548:          ; GUNFIGHT BACKGROUND JOB
4549:          ; CONSISTING OF INITIALIZATION, PRE-ROUND DISPLAY,
4550:          ; MONITORING OF CONTROLS AND VECTOR DELTA CHANGING
4551:          ; DEATH, POST ROUND STUFF AND END GAME
4552:
4553:          ;EQUATES
4554: 0008          LNX      EQU  8          ; LEFT NUMBER X
4555: 0002          BSY      EQU  2          ; BANNER STRINGS Y
4556: 0088          RNX      EQU 136         ; RIGHT NUMBER X
4557: 0020          LBULX    EQU  32         ; LEFT BULLETS X
4558: 0068          RBULX    EQU 104         ; RIGHT BULLETS X
4559: 004C          STMRX    EQU  76         ; SUB TIMER X
4560: 002C          GRX      EQU  44         ; GET READY X
4561: 0001          GRY      EQU  1          ; GET READY Y
4562: 0040          DRX      EQU  64         ; DRAW X
4563: 0014          TCACY    EQU  20         ; TOP CACTUS Y
4564: 000F          TTREEY   EQU TCACY-5
4565: 002A          MCACY    EQU  42         ; MID CACTUS Y
4566: 0046          BCACY    EQU  70         ; BOTTOM CACTUS Y
4567: 0041          BTREEY   EQU BCACY-5
4568: 0040          LCACX    EQU  64         ; LEFT CACTUS X
4569: 0058          RCACX    EQU  88         ; RIGHT CACTUS X
4570: 004C          CCACX    EQU  76         ; CENTER CACTUS X
4571: 0048          WAGX     EQU  72         ; WAGON X
4572: 0060          COWX     EQU RCACX+8     ; OTHER COWBOYS WINDOW X
4573:          ;
4574: 000A          TLINE    EQU  10         ; TOP LINE OF GUNSPACE
4575: 0009          ALINE    EQU TLINE-1
4576: 005C          BLINE    EQU  92         ; BOTTOM LINE OF GUNSPACE
4577:          ;
4578: 0012          BULVSZ   EQU  18         ; BULLET VECTOR SIZE
4579: 0017          GFVSIZ   EQU  23
4580: 0012          WAGVSZ   EQU  18         ; WAGON VECTOR SIZE
4581:          ;
4582: 0032          WINBND   EQU  50         ; TOP BOTTOM WINDOW BOUNDARY
4583: 006A          TOPLIN   EQU 53*2        ; TOP WINDOWS LINE
4584: 0000          BOTLIN   EQU  0          ; BOTTOM WINDOW LINE
4585: 00C8          LFRLIN   EQU 100*2       ; LOW PRIORITY FOREGROUND LINE
4586:          ;
4587: FFFF          NEXT     EQU -1          ; NEXT LINE FOR QUEUES
4588: 000F          VBARM    EQU VBOAH+1     ; ARM STATE

```

Home Video Game System Library - Gunfight

```

4589: 0010          VBOARM EQU VBOARM+1      ; LAST ARM PATTERN WRITTEN
4590: 0011          VBLEGT EQU VBOARM+1      ; LEG TIMER
4591: 0012          VBLEG EQU VBLEGT+1        ; LEG LINK
4592: 0013          VBCOMP EQU VBLEG+1        ; TIMER FOR COMPUTER CONTROL
4593:                ; BITS
4594: 0000          VBSWAG EQU 0              ; WAGON BIT
4595: 0003          VBSCHG EQU 3              ; CHANGE STATUS BIT
4596: 0004          VBSNOM EQU 4              ; NOT MOVING STATUS
4597: 0005          VBSINT EQU 5              ; INTERCEPTED/DEAD STATUS
4598:
4599:                ; *****
4600:                ; * SUBROUTINES *
4601:                ; *****
4602:                ; DISPLAY CLOCK AND UPDATE CT4
4603: 17E1 F3          DCLOCK DI
4604: 17E2                SYSSUK DECCTS
4604: 17E2 FF          RST 56
4604: 17E3 11          DB DECCTS+1
4604: 0000                IF DECCTS = INTPC
4604:                ENDIF
4604: 17E4                ENDM
4604:
4605: 17E4 80                DB 10000000B
4606: 17E5 DD210D02        LD IX, FNTSML
4607: 17E9 3ADC4F          LD A, (CT7)
4608: 17EC B7              OR A
4609: 17ED 2808            JR Z, $(DCOUT-$)
4610: 17EF                SYSSUK DISNUM
4610: 17EF FF          RST 56
4610: 17F0 37              DB DISNUM+1
4610: 0000                IF DISNUM = INTPC
4610:                ENDIF
4610: 17F1                ENDM
4610:
4611: 17F1 4C              DB STMRX
4612: 17F2 02              DB BSY
4613: 17F3 0B              DB TIME
4614: 17F4 42              DB 42H
4615: 17F5 DC4F            DW CT7
4616: 17F7 AF              DCOUT XOR A
4617: 17F8 D30C            OUT (MAGIC), A
4618: 17FA 32FF0F          LD (WASTE), A
4619: 17FD FB              EI
4620: 17FE C9              RET
4621:                ; FIRE BULLETS
4622:                ; LEFT COWBOY
4623: 17FF                FIRE0 SYSSUK SUCK
4623: 17FF FF          RST 56
4623: 1800 0D              DB SUCK+1
4623: 0000                IF SUCK = INTPC
4623:                ENDIF
4623: 1801                ENDM
4623:
4624: 1801 DC              DB 11011100B
4625: 1802 614F            DW LCOWB
4626: 1804 DA4F            DW LBULS
4627: 1806 194F            DW BULV1+1

```

Home Video Game System Library - Gunfight

```

4628: 1808 1809          JR    $(ZORE-$)
4629: 180A          FIRE1 SYSSUK SUCK
4629: 180A FF          RST    56
4629: 180B 0D          DB     SUCK+1
4629: 0000          IF     SUCK = INTPC
4629:          ENDIF
4629: 180C          ENDM
4629:
4630: 180C DC          DEFB  11011100B
4631: 180D 784F        DW    RCOWB
4632: 180F DB4F        DW    RBULS
4633: 1811 3D4F        DW    BULV3+1
4634: 1813 FD7E07      ZORE  LD    A,(IY+CBB)
4635: 1816 B7          OR    A
4636: 1817 C8          RET   Z
4637: 1818 0A          LD    A,(BC)          ; GET BULLET COUNT
4638: 1819 B7          OR    A
4639: 181A C8          RET   Z
4640: 181B 7E          LD    A,(HL)          ; CHECK IF BULLET IS AVAILABLE
4641: 181C B7          OR    A
4642: 181D 2809        JR    Z,$+(ZOK-$)
4643: 181F 111200      LD    DE,BULVSZ      ; DELTA TO NEXT BULLET
4644: 1822 19          ADD   HL,DE
4645: 1823 7E          LD    A,(HL)
4646: 1824 B7          OR    A
4647: 1825 2801        JR    Z,$+(ZOK-$)
4648: 1827 C9          RET
4649:          ; N?W HL->BULLET
4650:          ; IX->COWBOY
4651:          ; SUB 1 FROM BULLET COUNT
4652: 1828 0A          ZOK  LD    A,(BC)
4653: 1829 3D          DEC   A
4654: 182A 02          LD    (BC),A
4655:          ; SET SUB TIMER IF OUT OF BULLETS
4656: 182B 200D        JR    NZ,$+(BERASE-$)
4657: 182D 3ADC4F      LD    A,(CT7)
4658: 1830 B7          OR    A
4659: 1831 3E10        LD    A,10H
4660: 1833 2802        JR    Z,$+(STSEC-$)
4661: 1835 3E02        LD    A,2
4662: 1837 32DC4F      STSEC LD    (CT7),A
4663: 183A E5          BERASE PUSH HL
4664: 183B DDE5        PUSH  IX
4665: 183D 0A          LD    A,(BC)
4666: 183E 6F          LD    L,A
4667: 183F 2600        LD    H,0
4668: 1841 29          ADD   HL,HL
4669: 1842 29          ADD   HL,HL          ; *4
4670: 1843 116802      LD    DE,BSY*256+RBULX
4671: 1846 DDCB0076     BIT   MRFLOP,(IX+VBMR)
4672: 184A 3E40        LD    A,40H          ; FLOPPED MR
4673: 184C 2801        JR    Z,$+(RITB-$)
4674: 184E AF          XOR   A          ; NORMAL MR
4675:          ; NOW POSITION AND ERASE
4676: 184F 19          RITB ADD   HL,DE
4677: 1850 EB          EX    DE,HL
4678: 1851          SYSTEM RELAB1

```

Home Video Game System Library - Gunfight

```

4678: 1851 FF          RST      56
4678: 1852 3A          DB       RELAB1
4678: 0000              IF       RELAB1 = INTPC
4678:                  ENDIF
4678: 1853              ENDM
4678:
4679: 1853 EB          EX      DE,HL
4680: 1854 0605         LD       B,5
4681: 1856 112800        LD       DE,40          ; INC TO NEXT LINE
4682: 1859 36FF          BELP     LD       (HL),0FFH      ; ERASE LINE
4683: 185B 19          ADD      HL,DE          ; GO DOWN A LINE
4684: 185C 10FB         DJNZ    $+(BELP-$)
4685: 185E 1600         LD       D,0
4686: 1860 DD5E0F        LD       E,(IX+VBARM)  ; GET CURRENT ARM POS
4687: 1863 62          LD       H,D
4688: 1864 6B          LD       L,E
4689: 1865 29          ADD      HL,HL          ; *2
4690: 1866 19          ADD      HL,DE          ; *3
4691: 1867 118F1D       LD       DE,BULTAB
4692: 186A 19          ADD      HL,DE          ; -> BULTAB(ARM)
4693: 186B EB          EX      DE,HL
4694: 186C C1          POP      BC            ; BC<=IX
4695: 186D E1          POP      HL            ; BUL [STAT]
4696: 186E E5          PUSH     HL            ; SAVE FOR ACTIVATE
4697: 186F 23          INC      HL            ; BUL [DEL TIME]
4698: 1870 3601        LD       (HL),1        ; MAKE BULIT JUMP OUT
4699: 1872 23          INC      HL            ; BUL [DEL XLOW]
4700: 1873 03          INC      BC            ; COW [STAT]
4701: 1874 03          INC      BC            ; COW [DEL TIME]
4702: 1875 03          INC      BC            ; COW [DX LO]
4703: 1876 CDD319       CALL    PUTVEC
4704: 1879 03          INC      BC            ; COW [XCHK]
4705: 187A 03          INC      BC            ; COW [DY LO]
4706: 187B 23          INC      HL            ; BUL [XCHK]
4707: 187C 3601        LD       (HL),1        ; LIMIT CHECK
4708: 187E 23          INC      HL            ; BUL [DY LO]
4709: 187F CDD319       CALL    PUTVEC
4710: 1882 E1          POP      HL            ; BUL [STAT]
4711: 1883 3680        LD       (HL),80H      ; ACTIVE
4712: 1885              SYSSUK BMUSIC
4712: 1885 FF          RST      56
4712: 1886 13          DB       BMUSIC+1
4712: 0000              IF       BMUSIC = INTPC
4712:                  ENDIF
4712: 1887              ENDM
4712:
4713: 1887 124F         DW       MSTACK
4714: 1889 01          DB       00000001B     ; JUST NOISE
4715: 188A D71F         DW       GUNSHOT
4716: 188C C9          RET
4717:                  ; TAKE A COFFEE BREAK
4718: 188D          NBRK     DONT    PIZBRK      ; SEE IF I CARE
4718: 188D 48          DB       PIZBRK
4718: 188E              ENDM
4718:
4719: 188E          DO      MRET
4719: 188E 09          DB       MRET+1

```

Home Video Game System Library - Gunfight

```

4719: 188F          ENDM
4719:
4720:          ; CONVERT JOYSTICKS
4721: 188F DD21614F    JOY0    LD      IX,LCOWB
4722: 1893 1804          JR      $+(PJOY-$)
4723: 1895 DD21784F    JOY1    LD      IX,RCOWB
4724:          ; CONVERT JOYSTICKS
4725: 1899 DD4E00      PJOY    LD      C,(IX+VBMR)
4726: 189C 118000      LD      DE,128
4727: 189F 218000      LD      HL,128
4728: 18A2          SYSTEM MSKTD          ; COMPUTE DELTAS
4728: 18A2 FF          RST     56
4728: 18A3 7E          DB      MSKTD
4728: 0000          IF      MSKTD = INTPC
4728:          ENDIF
4728: 18A4          ENDM
4728:
4729: 18A4 DD7409          LD      (IX+VBDYH),H
4730: 18A7 DD7508          LD      (IX+VBDYL),L
4731: 18AA DD7204          LD      (IX+VBDXH),D
4732: 18AD DD7303          LD      (IX+VBDXL),E
4733: 18B0 C9          RET
4734: 18B1 DD21784F    PPOT1: LD      IX,RCOWB
4735: 18B5 78          LD      A,B          ; POT MUST BE FLOPPED BECAUSE
4736: 18B6 2F          CPL          ; ARM IS FLOPPED
4737: 18B7 1805      JR      $+(PPOT-$)
4738: 18B9 DD21614F    PPOT0 LD      IX,LCOWB
4739: 18BD 78          LD      A,B
4740:          ; CONVERT POT AND STORE
4741: 18BE E6E0      PPOT AND 0E0H
4742: 18C0 0F          RRCA
4743: 18C1 0F          RRCA
4744: 18C2 0F          RRCA
4745: 18C3 0F          RRCA
4746: 18C4 FE0E      CP      0EH
4747: 18C6 2002      JR      NZ,$+(KART-$)
4748: 18C8 3E0C      LD      A,0CH          ; IF KNOB=7 THEN SET TO 6
4749: 18CA DD770F    KART LD      (IX+VBARM),A          ; SET ARM POSITION
4750: 18CD C9          RET
4751:          ; CHECK IF BULLET HIT ANYTHING
4752: 18CE DD7E01    HITCHK LD      A,(IX+VBSTAT)
4753: 18D1 E660          AND     060H
4754: 18D3 FE20      CP      20H          ; CHECK ONLY IF BLANKED
4755: 18D5 280F      JR      Z,$+(HIT-$)
4756: 18D7 D0          RET     NC          ; RETURN IF NOT BLANKED
YET
4757: 18D8 DDCB075E    BIT     VBCLAT,(IX+VBXCHK)
4758: 18DC C8          RET     Z
4759: 18DD DD360100    LD      (IX+VBSTAT),0          ; BULLET HIT WALL
4760: 18E1 DD360701    LD      (IX+VBXCHK),1          ; SET LIMIT CHECK
4761: 18E5 C9          RET
4762: 18E6 DD7E06    HIT LD      A,(IX+VBXH)          ; CHECK WHAT PART OF SCR ITS
ON
4763: 18E9 FE48      CP      WAGX
4764: 18EB 300E      JR      NC,$+(HIT1-$)
4765: 18ED DD360202    LD      (IX+VBTIMB),2          ;MAKE IT JUMP OUT
4766: 18F1 DD360180    LD      (IX+VBSTAT),80H          ; RE ACTIVATE

```

Home Video Game System Library - Gunfight

```

4767: 18F5 218B1D          LD    HL,BULLMT
4768: 18F8                SYSTEM VECT
4768: 18F8 FF                RST   56
4768: 18F9 3E                DB    VECT
4768: 0000                IF    VECT = INTPC
4768:                    ENDIF
4768: 18FA                ENDM
4768:
4769: 18FA C9                RET
4770: 18FB DD360100          HIT1  LD    (IX+VBSTAT),0 ; BULLET DIES FROM WAGON ON
4771: 18FF FE58                CP    RCACX
4772: 1901 301D                JR    NC,$+(HIT2-$)
4773: 1903 3A904F          LD    A,(WAGON)
4774: 1906 B7                OR    A                ; IS IT A CACTII?
4775: 1907 C0                RET   NZ                ; NOPE ITS A WAGON
4776: 1908 1E4C                LD    E,CCACX          ; LOAD X
4777:                    ; ERASE OBJECT BULLET HITS
4778: 190A DD560B          ERASE_ LD    D,(IX+VBYH) ; LOAD Y
4779: 190D 15                DEC   D
4780: 190E                SYSSUK RELAB1
4780: 190E FF                RST   56
4780: 190F 3B                DB    RELAB1+1
4780: 0000                IF    RELAB1 = INTPC
4780:                    ENDIF
4780: 1910                ENDM
4780:
4781: 1910 00                DB    0
4782: 1911 EB                EX    DE,HL
4783: 1912 11D7FF          LD    DE,-41
4784: 1915 0600                LD    B,0
4785: 1917 7E                ELOP  LD    A,(HL)
4786: 1918 70                LD    (HL),B          ; ZERO THE SCREEN BYTE
4787: 1919 23                INC   HL
4788: 191A B6                OR    (HL)
4789: 191B 70                LD    (HL),B
4790: 191C 19                ADD   HL,DE
4791: 191D 20F8                JR    NZ,$+(ELOP-$)
4792: 191F C9                RET
4793: 1920 FE60          HIT2  CP    RCACX+8          ; GUNFTR SPACE
4794: 1922 300C                JR    NC,$+(DIE-$)
4795: 1924 1E40                LD    E,LCACX
4796: 1926 DDCB0076          BIT   MRFLOP,(IX+VBMR)
4797: 192A 20DE                JR    NZ,$+(ERASE_-$)
4798: 192C 1E58                LD    E,RCACX
4799: 192E 18DA                JR    $+(ERASE_-$)
4800: 1930 DDCB0076          DIE   BIT   MRFLOP,(IX+VBMR) ; WHO DIED?
4801: 1934 280C                JR    Z,$+(DLEFT-$)
4802: 1936                SYSSUK SUCK
4802: 1936 FF                RST   56
4802: 1937 0D                DB    SUCK+1
4802: 0000                IF    SUCK = INTPC
4802:                    ENDIF
4802: 1938                ENDM
4802:
4803: 1938 DD                DB    11011101B
4804: 1939 614F          DW    LCOWB
4805: 193B 08                DB    8

```

Home Video Game System Library - Gunfight

```

4806: 193C AD1F          DW    TAPS
4807: 193E A64F          DW    RSCORE
4808: 1940 180A          JR    $+(DIE1-$)
4809: 1942          DLEFT SYSSUK SUCK
4809: 1942 FF            RST    56
4809: 1943 0D            DB     SUCK+1
4809: 0000          IF     SUCK = INTPC
4809:          ENDIF
4809: 1944          ENDM
4809:
4810: 1944 DD            DB    11011101B
4811: 1945 784F          DW    RCOWB
4812: 1947 64           DB    100
4813: 1948 BD1F          DW    FUNERL
4814: 194A A24F          DW    LSCORE
4815: 194C DD361106      DIE1  LD    (IX+VBLEGT),6      ; SET FIRST CELL TIME
4816: 1950 DD361280      LD    (IX+VBLEG),KILL AND OFFH ; ??
4817: 1954 DD360168      LD    (IX+VBSTAT),068H      ; KILL HIM
4818: 1958 DD7E0B      LD    A,(IX+VBYH)          ; WHERE TO WRITE GOT ME
4819: 195B D608          SUB    8
4820: 195D FE13          CP    TLINE+9
4821: 195F 3002          JR    NC, $+(DIE4-$)
4822: 1961 C620          ADD    A,32
4823: 1963 57           DIE4  LD    D,A
4824: 1964          SYSTEM INCSCR
4824: 1964 FF            RST    56
4824: 1965 54           DB     INCSCR
4824: 0000          IF     INCSCR = INTPC
4824:          ENDIF
4824: 1966          ENDM
4824:
4825: 1966 2B           DEC    HL
4826: 1967 7E           LD    A,(HL)              ; FIELD
4827: 1968 FE05          CP    5
4828: 196A CE00          ADC    A,0
4829: 196C 77           LD    (HL),A
4830:          ; PLAY DEATH SONG
4831: 196D 60           LD    H,B
4832: 196E 69           LD    L,C
4833: 196F DD21124F      LD    IX,MSTACK
4834: 1973 3EC0          LD    A,11000000B
4835: 1975          SYSTEM BMUSIC
4835: 1975 FF            RST    56
4835: 1976 12           DB     BMUSIC
4835: 0000          IF     BMUSIC = INTPC
4835:          ENDIF
4835: 1977          ENDM
4835:
4836: 1977 0E0C          LD    C,LARG2
4837: 1979 21021F      LD    HL,GOTME
4838: 197C F3           DI
4839: 197D          SYSTEM STRDIS
4839: 197D FF            RST    56
4839: 197E 34           DB     STRDIS
4839: 0000          IF     STRDIS = INTPC
4839:          ENDIF
4839: 197F          ENDM

```

Home Video Game System Library - Gunfight

```

4839:
4840: 197F          SYSSUK PAWS
4840: 197F FF          RST      56
4840: 1980 51          DB       PAWS+1
4840: 0000            IF       PAWS = INTPC
4840:                ENDIF
4840: 1981            ENDM
4840:
4841: 1981 FA          DB       250
4842: 1982 3E01        LD       A,1
4843: 1984 32DE4F      LD       (SEMI4S),A ; SET FLAG0
4844: 1987 C9          RET
4845:                ; FIELD PUTS UP THE SACTII APPROP TO SCORE
4846:                ; A=SCORE OF OPP PLAYER UPTO 6
4847:                ; BC -> ARRAY OF Y POSITIONS
4848: 1988 21F41E      FIELD   LD     HL,CACTUS ; -> CACTUS PATTERN
4849: 198B F5          PUSH    AF
4850: 198C 3E08        LD     A,1000B
4851: 198E D319        OUT    (XPAND),A
4852: 1990 F1          POP     AF
4853: 1991 FE01        CP     1
4854: 1993 D8          RET     C
4855: 1994 FE04        CP     4
4856: 1996 3003        JR     NC, $(TCAC-$)
4857: 1998 CDC819      CALL   CACW
4858: 199B 03          TCAC   INC     BC
4859: 199C FE02        CP     2
4860: 199E D8          RET     C
4861: 199F FE05        CP     5
4862: 19A1 3003        JR     NC, $(MCAC-$)
4863: 19A3 CDC819      CALL   CACW
4864: 19A6 FE03        MCAC   CP     3
4865: 19A8 D8          RET     C
4866: 19A9 03          INC     BC
4867: 19AA 08          EX     AF,AF'
4868: 19AB 3E81        LD     A,81H ; ACIVATE WAGON
4869: 19AD 32904F      LD     (WAGON),A
4870: 19B0 08          EX     AF,AF'
4871: 19B1 CDC819      CALL   CACW
4872: 19B4 FE04        CP     4
4873: 19B6 D8          RET     C
4874: 19B7 03          INC     BC
4875: 19B8 21E51D      LD     HL,TREE
4876: 19BB F5          PUSH    AF
4877: 19BC 3E0C        LD     A,1100B
4878: 19BE D319        OUT    (XPAND),A
4879: 19C0 F1          POP     AF
4880: 19C1 CDC819      CALL   CACW
4881: 19C4 FE05        CP     5
4882: 19C6 D8          RET     C
4883: 19C7 03          INC     BC
4884: 19C8 F5          CACW   PUSH    AF
4885: 19C9 D5          PUSH    DE
4886: 19CA 0A          LD     A,(BC)
4887: 19CB 57          LD     D,A
4888: 19CC 3E08        LD     A,8 ; EXPAND
4889: 19CE            SYSTEM WRITP

```

Home Video Game System Library - Gunfight

```

4889: 19CE FF          RST    56
4889: 19CF 22          DB     WRITP
4889: 0000          IF     WRITP = INTPC
4889:                ENDIF
4889: 19D0          ENDM
4889:
4890: 19D0 D1          POP    DE
4891: 19D1 F1          POP    AF
4892: 19D2 C9          RET
4893:                ; PUT DEL X,Y INTO BULLET VECTORS
4894: 19D3 1A          PUTVEC LD    A,(DE)      ; TABLE [D LO]
4895: 19D4 77          LD     (HL),A      ; BUL [D LO]
4896: 19D5 13          INC    DE          ; TAB [D HI]
4897: 19D6 03          INC    BC          ; COW [D HI]
4898: 19D7 23          INC    HL          ; BUL [D HI]
4899: 19D8 1A          LD     A,(DE)
4900: 19D9 77          LD     (HL),A
4901: 19DA 23          INC    HL          ; BUL [LO]
4902: 19DB 13          INC    DE          ; TAB [HI]
4903: 19DC 03          INC    BC          ; COW [LO]
4904: 19DD 3600        LD     (HL),0
4905: 19DF 03          INC    BC          ; COW [HI]
4906: 19E0 23          INC    HL          ; BUL [HI]
4907: 19E1 0A          LD     A,(BC)
4908: 19E2 EB          EX     DE,HL
4909: 19E3 86          ADD    A,(HL)
4910: 19E4 EB          EX     DE,HL
4911: 19E5 77          LD     (HL),A      ; BUL [HI]=COW [HI]+TAB [HI]
4912: 19E6 13          INC    DE          ; TAB [D HI]
4913: 19E7 C9          RET
4914:                ; GUNFIGHT START UP ROUTINE (ONCE PER GAME)
4915: 19E8          INIT    SYSSUK GETPAR
4915: 19E8 FF          RST    56
4915: 19E9 4D          DB     GETPAR+1
4915: 0000          IF     GETPAR = INTPC
4915:                ENDIF
4915: 19EA          ENDM
4915:
4916: 19EA 1E02        DW     MXSCR
4917: 19EC 84          DB     84H
4918: 19ED F44F        DW     ENDSCR
4919: 19EF 31064F      LD     SP,STACK
4920: 19F2          SYSTEM INTPC
4920: 19F2 FF          RST    56
4920: 19F3 00          DB     INTPC
4920: 0001          IF     INTPC = INTPC
4920: 0001          INTPCC DEFL    1
4920:                ENDIF
4920: 19F4          ENDM
4920:
4921: 19F4          DO     FILL
4921: 19F4 1B          DB     FILL+1
4921: 19F5          ENDM
4921:
4922: 19F5 064F        DW     STACK
4923: 19F7 D600        DW     CT7-STACK
4924: 19F9 00          DB     0

```

Home Video Game System Library - Gunfight

```

4925: 19FA          DO      SETB
4925: 19FA 7B       DB      SETB+1
4925: 19FB          ENDM
4925:
4926: 19FB 02       DB      2*GSBSCR
4927: 19FC F84F     DW      GAMSTB
4928: 19FE          DO      SETOUT      ; SET UP GAME PORTS
4928: 19FE 17       DB      SETOUT+1
4928: 19FF          ENDM
4928:
4929: 19FF B8         DB      BLINE*2      ; BOTTOM LINE - VRT BLK
4930: 1A00 D6         DB      214D        ; HORZ BOUNDS
4931: 1A01 08       DB      8           ; INMOD
4932: 1A02          DO      COLSET
4932: 1A02 19       DB      COLSET+1
4932: 1A03          ENDM
4932:
4933: 1A03 C31D     DW      GFCOLS
4934: 1A05          DO      BMUSIC      ; PLAYS STREETS OF LOREDO
4934: 1A05 13       DB      BMUSIC+1
4934: 1A06          ENDM
4934:
4935: 1A06 124F     DW      MSTACK
4936: 1A08 C0         DB      11000000B   ; ON VOICE A
4937: 1A09 9F1F     DW      HOME
4938: 1A0B          EXIT
4938: 1A0B 02       DB      XINTC
4938: 0000          INTPCC DEF 0
4938: 1A0C          ENDM
4938:
4939:                ; *****
4940:                ; ONCE A ROUND START UP ROUTINE
4941:                ; *****
4942: 1A0C F3       STRND  DI
4943: 1A0D          SYSTEM INTPC
4943: 1A0D FF       RST    56
4943: 1A0E 00       DB      INTPC
4943: 0001          IF      INTPC = INTPC
4943: 0001          INTPCC DEF 1
4943:                ENDIF
4943: 1A0F          ENDM
4943:
4944:                ; INIT HANDLES, BULLETS, TIMERS
4945: 1A0F          DO      MOVE
4945: 1A0F 5F       DB      MOVE+1
4945: 1A10          ENDM
4945:
4946: 1A10 DA4F     DW      CT5
4947: 1A12 0C00     DW      12
4948: 1A14 CB1D     DW      SINIT
4949:                ; COLOR BANNER
4950: 1A16          FILLq  NORMEM, BYTEPL*ALINE, 0FFH
4950: 1A16 1B       DB      FILL+1
4950: 1A17 0040     DW      NORMEM
4950: 1A19 6801     DW      BYTEPL*ALINE
4950: 1A1B FF       DB      0FFH
4950: 1A1C          ENDM

```

Home Video Game System Library - Gunfight

```

4950:
4951:      ; ERASE SCREEN
4952: 1A1C      FILLq NORMEM+BYTEPL*ALINE,BYTEPL*(BLINE-ALINE),0
4952: 1A1C 1B      DB      FILL+1
4952: 1A1D 6841     DW      NORMEM+BYTEPL*ALINE
4952: 1A1F F80C      DW      BYTEPL*(BLINE-ALINE)
4952: 1A21 00      DB      0
4952: 1A22      ENDM
4952:
4953:      ; RESET VECTORS
4954: 1A22      FILLq STRRAM,ENDRAM_-STRRAM,0
4954: 1A22 1B      DB      FILL+1
4954: 1A23 124F     DW      STRRAM
4954: 1A25 8F00     DW      ENDRAM_-STRRAM
4954: 1A27 00      DB      0
4954: 1A28      ENDM
4954:
4955:      ; SHOW SCORES
4956: 1A28      DO      SUCK
4956: 1A28 0D      DB      SUCK+1
4956: 1A29      ENDM
4956:
4957: 1A29 10      DB      00010000B
4958: 1A2A 0D02     DW      FNTSML
4959: 1A2C      DO      DISNUM
4959: 1A2C 37      DB      DISNUM+1
4959: 1A2D      ENDM
4959:
4960: 1A2D 08      DB      LNX
4961: 1A2E 02      DB      BSY
4962: 1A2F 0B      DB      TIME
4963: 1A30 C4      DB      0C4H      ; ZERO SUPRS,SMALL
4964: 1A31 A24F     DW      LSCORE
4965: 1A33      DO      DISNUM
4965: 1A33 37      DB      DISNUM+1
4965: 1A34      ENDM
4965:
4966: 1A34 88      DB      RNX
4967: 1A35 02      DB      BSY
4968: 1A36 0B      DB      TIME
4969: 1A37 C4      DB      0C4H
4970: 1A38 A64F     DW      RSCORE
4971:      ; CHECK FOR END GAME
4972: 1A3A      DO      RCALL
4972: 1A3A 05      DB      RCALL+1
4972: 1A3B      ENDM
4972:
4973: 1A3B 2C1B     DW      ENDGAM
4974: 1A3D      TEXTD GETRDY,GRX,GRY,LARGE
4974: 1A3D 35      DB      STRDIS+1
4974: 1A3E 2C      DB      GRX
4974: 1A3F 01      DB      GRY
4974: 1A40 0B      DB      LARGE
4974: 1A41 7A1D     DW      GETRDY
4974: 1A43      ENDM
4974:
4975: 1A43      EXIT

```

Home Video Game System Library - Gunfight

```

4975: 1A43 02          DB      XINTC
4975: 0000          INTPCC DEF 0
4975: 1A44          ENDM
4975:
4976: 1A44 AF          XOR    A          ; SET UP WAGON
4977: 1A45 32904F      LD     (WAGON),A  ; STOP WAGON
4978:          ; PUT UP PLAY FIELD
4979: 1A48 3AA14F      LD     A,(RFIELD) ; NUMBER OF CACTII
4980: 1A4B 1E58          LD     E,RCACX    ; RIGHT CAC COLUMN
4981: 1A4D 01BE1D      LD     BC,RFTAB   ; POSITIONS OF TABLE OF CACTII
4982: 1A50 CD8819      CALL  FIELD      ; PUT THE CACTII UP
4983: 1A53 3AA54F      LD     A,(LFIELD)
4984: 1A56 1E40          LD     E,LCACX
4985: 1A58 01B91D      LD     BC,LFTAB
4986: 1A5B CD8819      CALL  FIELD
4987:          ; INITIALIZE Q POINTERS
4988: 1A5E 3E4F          INITQ LD     A,LCOWB SHR 8
4989: 1A60 32144F      LD     (WRITQ+2),A
4990: 1A63 32174F      LD     (VECQ+2),A
4991:          ; SETUP VECTORS SO COWBOYS WALK OUT
4992: 1A66 DD21614F      LD     IX,LCOWB   ; LEFT COWBOY VECTOR
4993: 1A6A DD360010      LD     (IX+VBMR),10H
4994: 1A6E 21154F      LD     HL,VECQ
4995: 1A71 CD301D      CALL  COWINT
4996: 1A74 DD21784F      LD     IX,RCOWB   ; RIGHT COWBOY VECTOR
4997: 1A78 DD360050      LD     (IX+VBMR),50H
4998: 1A7C CD301D      CALL  COWINT
4999: 1A7F 3A904F      LD     A,(WAGON) ; IF WAGON IS ON
5000: 1A82 B7           OR     A
5001: 1A83 281D          JR     Z,$+(MIDC-$)
5002: 1A85 DD218F4F      LD     IX,WAGVEC  ; THEN ACTIVATE WAGON
5003: 1A89 DD360010      LD     (IX+VBMR),10H
5004: 1A8D DD360C03      LD     (IX+VBYCHK),3
5005: 1A91 DD360840      LD     (IX+VBDYL),40H
5006: 1A95 DD360648      LD     (IX+VBXH),72
5007: 1A99 DD360B0A      LD     (IX+VBYH),TLINE
5008: 1A9D CD501D      CALL  ADDTQ
5009: 1AA0 180B          JR     $+(BORG-$)
5010: 1AA2 3E08          MIDC  LD     A,8
5011: 1AA4 D319          OUT   (XPAND),A
5012: 1AA6          SYSSUK WRITP    ; ELSE PUT UP A CACTUS
5012: 1AA6 FF          RST   56
5012: 1AA7 23          DB    WRITP+1
5012: 0000          IF    WRITP = INTPC
5012:          ENDIF
5012: 1AA8          ENDM
5012:
5013: 1AA8 4C          DB    CCACX
5014: 1AA9 2A          DB    MCACY
5015: 1AAA 08          DB    8          ; EXPAND
5016: 1AAB F41E          DW    CACTUS
5017:          ; INITIALIZE BULLET VECTORS
5018: 1AAD 111200      BORG  LD     DE,BULVSZ
5019: 1AB0 DD21184F      LD     IX,BULV1
5020: 1AB4 012004      LD     BC,4*256+20H
5021: 1AB7 3E02          LD     A,2
5022: 1AB9 B8          BULLP CP    B

```

Home Video Game System Library - Gunfight

```

5023: 1ABA 2002          JR    NZ, $(TIYU-$)
5024: 1ABC 0E60          LD    C,60H
5025: 1ABE DD7100          TIYU  LD    (IX+VBMR),C
5026: 1AC1 DD360701        LD    (IX+VBXCHK),1
5027: 1AC5 DD360C03        LD    (IX+VBYCHK),3
5028: 1AC9 DD19          ADD   IX,DE
5029: 1ACB 10EC          DJNZ  $(BULLP-$)
5030:                ; FIRE UP INTERRUPTS
5031: 1ACD 3E1D          LD    A,INTTBL SHR 8
5032: 1ACF ED47          LD    I,A
5033:                ;          IM      2          ; DONE IN MENU
5034: 1AD1 3E74          LD    A,LFRVEC AND 0FFH
5035: 1AD3 D30D          OUT   (INFBK),A
5036:                ; ***
5037:                ; LET COWBOYS WALK OUT
5038:                ; ***
5039: 1AD5                WALK  SYSSUK PAWS
5039: 1AD5 FF                RST   56
5039: 1AD6 51                DB    PAWS+1
5039: 0000                IF    PAWS = INTPC
5039:                ENDIF
5039: 1AD7                ENDM
5039:
5040: 1AD7 64                DB    100
5041: 1AD8 F3                DI
5042: 1AD9 DD210D02          LD    IX,FNTSML
5043: 1ADD                SYSTEM INTPC
5043: 1ADD FF                RST   56
5043: 1ADE 00                DB    INTPC
5043: 0001                IF    INTPC = INTPC
5043: 0001                INTPCC DEFL 1
5043:                ENDIF
5043: 1ADF                ENDM
5043:
5044:                ; ERASE GET READY
5045: 1ADF                DO    BLANK
5045: 1ADF 2B                DB    BLANK+1
5045: 1AE0                ENDM
5045:
5046: 1AE0 12                DB    18          ; X SIZE
5047: 1AE1 08                DB    8           ; Y SIZE
5048: 1AE2 FF                DB    0FFH       ; DATA TO BLANK TO
5049: 1AE3 3340             DW    4033H      ; BLANK ADDRESS
5050: 1AE5                TEXTD DRAW,DRX,GRY,LARGE
5050: 1AE5 35                DB    STRDIS+1
5050: 1AE6 40                DB    DRX
5050: 1AE7 01                DB    GRY
5050: 1AE8 0B                DB    LARGE
5050: 1AE9 871D             DW    DRAW
5050: 1AEB                ENDM
5050:
5051: 1AEB                DO    CHRDIS
5051: 1AEB 33                DB    CHRDIS+1
5051: 1AEC                ENDM
5051:
5052: 1AEC 20                DB    LBULX
5053: 1AED 02                DB    BSY

```

Home Video Game System Library - Gunfight

```

5054: 1AEE 0B          DB    BULT
5055: 1AEF BB          DB    0BBH          ; BULLET
5056: 1AF0            DO    MCALL          ; 5 MORE
5056: 1AF0 07          DB    MCALL+1
5056: 1AF1            ENDM
5056:
5057: 1AF1 531B         DW    BULRIT
5058: 1AF3            DO    SUCK
5058: 1AF3 0D          DB    SUCK+1
5058: 1AF4            ENDM
5058:
5059: 1AF4 01            DB    00000001B
5060: 1AF5 68            DB    RBULX          ; DO THE RIGHT ONES
5061: 1AF6            DONT  CHRDIS          ; DISPAY THE FIRST ONE
5061: 1AF6 32            DB    CHRDIS
5061: 1AF7            ENDM
5061:
5062: 1AF7            DO    MCALL          ; DISPLAY THR OTHER FIVE
5062: 1AF7 07          DB    MCALL+1
5062: 1AF8            ENDM
5062:
5063: 1AF8 531B         DW    BULRIT
5064: 1AFA            DO    PAWS
5064: 1AFA 51            DB    PAWS+1
5064: 1AFB            ENDM
5064:
5065: 1AFB 3C            DB    60
5066: 1AFC            DO    BLANK
5066: 1AFC 2B            DB    BLANK+1
5066: 1AFD            ENDM
5066:
5067: 1AFD 08            DB    8              ; X SIZE
5068: 1AFE 08            DB    8              ; Y SIZE
5069: 1AFF FF            DB    0FFH          ; DATA TO BLANK TO
5070: 1B00 3840         DW    4038H          ; BLANK ADDRESS
5071: 1B02            EXIT
5071: 1B02 02            DB    XINTC
5071: 0000            INTPCC  DEFL    0
5071: 1B03            ENDM
5071:
5072:                ;
5073:                ; *****
5074:                ; MAIN LOOP DURING ROUND
5075:                ; GETS HANDLES, SETS VECTORS AND CHECKS BULLETS
5076: 1B03            LOOP    SYSTEM INTPC
5076: 1B03 FF            RST    56
5076: 1B04 00            DB    INTPC
5076: 0001            IF    INTPC = INTPC
5076: 0001            INTPCC  DEFL    1
5076:                ENDF
5076: 1B05            ENDM
5076:
5077: 1B05            DO    SENTRY
5077: 1B05 43            DB    SENTRY+1
5077: 1B06            ENDM
5077:
5078: 1B06 1402         DW    ALKEYS

```

Home Video Game System Library - Gunfight

```

5079: 1B08          DO      DOIT
5079: 1B08  45       DB      DOIT+1
5079: 1B09          ENDM
5079:
5080: 1B09  341B     DW      DTAB
5081: 1B0B          EXIT
5081: 1B0B  02       DB      XINTC
5081: 0000          INTPCC DEF  0
5081: 1B0C          ENDM
5081:
5082:          ;
5083:          ; CHECK FOR DEATHS
5084: 1B0C  DD21184F  DEATH  LD      IX,BULV1
5085: 1B10  111200     LD      DE,BULVSZ
5086: 1B13  0604     LD      B,4
5087: 1B15  C5       LPPP2  PUSH  BC
5088: 1B16  D5       PUSH  DE
5089: 1B17  CDCE18   CALL  HITCHK
5090: 1B1A  D1       POP   DE
5091: 1B1B  C1       POP   BC
5092: 1B1C  DD19     ADD   IX,DE
5093: 1B1E  3ADE4F   LD      A,(SEMI4S) ; CHECK IF DEATH MODE
5094: 1B21  3D       DEC   A
5095: 1B22  28DF     JR    Z,$+(LOOP-$)
5096: 1B24  10EF     DJNZ  $+(LPPP2-$)
5097: 1B26  18DB     JR    $+(LOOP-$)
5098:          ;
5099: 1B28          ENDRND EXIT
5099: 1B28  02       DB      XINTC
5099: 0000          INTPCC DEF  0
5099: 1B29          ENDM
5099:
5100: 1B29  C30C1A   JP     STRND
5101:          ;
5102: 1B2C  3AF84F   ENDGAM LD      A,(GAMSTB)
5103: 1B2F  CB7F     BIT   GSBEND,A
5104: 1B31  C8       RET   Z
5105: 1B32          SYSTEM QUIT
5105: 1B32  FF       RST   56
5105: 1B33  78       DB      QUIT
5105: 0000          IF    QUIT = INTPC
5105:          ENDIF
5105: 1B34          ENDM
5105:
5106:          ;
5107: 1B34          DTAB  JMPd  SCT7, ENDRND, 0
5107: 1B34  08       DB      SCT7
5107: 1B35  281B     DW      ENDRND
5107: 0000          IF    0 = $C0
5107:          ENDIF
5107: 1B37          ENDM
5107:
5108: 1B37          JMPd  SF0, ENDRND, 0
5108: 1B37  09       DB      SF0
5108: 1B38  281B     DW      ENDRND
5108: 0000          IF    0 = $C0
5108:          ENDIF

```

Home Video Game System Library - Gunfight

```

5108: 1B3A                                ENDM
5108:
5109: 1B3A                                RC    SP0, PPOT0, 0
5109: 1B3A 5C                              DB    SP0+$40
5109: 1B3B B918                            DW    PPOT0
5109: 0000                                IF    0 = $C0
5109:                                ENDF
5109: 1B3D                                ENDM
5109:
5110: 1B3D                                RC    SP1, PPOT1, 0
5110: 1B3D 5D                              DB    SP1+$40
5110: 1B3E B118                            DW    PPOT1
5110: 0000                                IF    0 = $C0
5110:                                ENDF
5110: 1B40                                ENDM
5110:
5111: 1B40                                RC    SJ0, JOY0, 0
5111: 1B40 55                              DB    SJ0+$40
5111: 1B41 8F18                            DW    JOY0
5111: 0000                                IF    0 = $C0
5111:                                ENDF
5111: 1B43                                ENDM
5111:
5112: 1B43                                RC    SJ1, JOY1, 0
5112: 1B43 57                              DB    SJ1+$40
5112: 1B44 9518                            DW    JOY1
5112: 0000                                IF    0 = $C0
5112:                                ENDF
5112: 1B46                                ENDM
5112:
5113: 1B46                                MC    SKYD, NBRK, 0
5113: 1B46 93                              DB    SKYD+$80
5113: 1B47 8D18                            DW    NBRK
5113: 0000                                IF    0 = $C0
5113:                                ENDF
5113: 1B49                                ENDM
5113:
5114: 1B49                                RC    ST0, FIRE0, 0
5114: 1B49 54                              DB    ST0+$40
5114: 1B4A FF17                            DW    FIRE0
5114: 0000                                IF    0 = $C0
5114:                                ENDF
5114: 1B4C                                ENDM
5114:
5115: 1B4C                                RC    ST1, FIRE1, 0
5115: 1B4C 56                              DB    ST1+$40
5115: 1B4D 0A18                            DW    FIRE1
5115: 0000                                IF    0 = $C0
5115:                                ENDF
5115: 1B4F                                ENDM
5115:
5116: 1B4F                                RC    SSEC, DCLOCK, ENDx
5116: 1B4F 51                              DB    SSEC+$40
5116: 1B50 E117                            DW    DCLOCK
5116: 0001                                IF    ENDx = $C0
5116: 1B52 C0                              DB    ENDx
5116:                                ENDF

```

Home Video Game System Library - Gunfight

```

5116: 1B53          ENDM
5116:
5117: 1B53          BULRIT  DONT  CHRDIS
5117: 1B53  32          DB    CHRDIS
5117: 1B54          ENDM
5117:
5118: 1B54          DONT  CHRDIS
5118: 1B54  32          DB    CHRDIS
5118: 1B55          ENDM
5118:
5119: 1B55          DONT  CHRDIS
5119: 1B55  32          DB    CHRDIS
5119: 1B56          ENDM
5119:
5120: 1B56          DONT  CHRDIS
5120: 1B56  32          DB    CHRDIS
5120: 1B57          ENDM
5120:
5121: 1B57          DONT  CHRDIS
5121: 1B57  32          DB    CHRDIS
5121: 1B58          ENDM
5121:
5122: 1B58          DONT  MRET
5122: 1B58  08          DB    MRET
5122: 1B59          ENDM
5122:
5123:
5124:
5125:                ; *****
5126:                ; * GUNFIGHT WRITE INTERRUPT ROUTINE *
5127:                ; *****
5127: 1B59  08          GFWRIT  EX    AF,AF'
5128: 1B5A  D9          EXX
5129: 1B5B  DDE5        PUSH  IX
5130: 1B5D  3E74        BEGINT LD    A,LFRVEC AND 0FFH ; ESTABLISH TICKS INT
5131: 1B5F  D30D        OUT   (INFBK),A
5132: 1B61  3EC8        LD    A,LFRLIN
5133: 1B63  D30F        OUT   (INLIN),A
5134: 1B65  21124F      LD    HL,WRITQ           ; GET FIRST WRITE Q ENTRY
5135: 1B68  CD671D      CALL  FIRST
5136: 1B6B  CD251D      CALL  DELQ               ; DROP FROM WRITE Q
5137: 1B6E  AF           XOR   A
5138: 1B6F  32FF0F      LD    (WASTE),A
5139: 1B72  DDCB0146    BIT   VBSWAG,(IX+VBSTAT) ; WAGON?
5140: 1B76  2028        JR    NZ,$+(GFWRT1-$)   ; JUMP IF YES
5141:                ; GUNFIGHTER - BLANKETH HIM
5142: 1B78  110514      LD    DE,1405H           ; LOAD BLANKING PARMS
5143: 1B7B          SYSTEM VBLANK           ; CALL BLANKER
5143: 1B7B  FF          RST   56
5143: 1B7C  28          DB    VBLANK
5143: 0000          IF    VBLANK = INTPC
5143:                ENDF
5143: 1B7D          ENDM
5143:
5144: 1B7D  261E        LD    H,LEG0 SHR 8       ; WRITE LEG PATTERN
5145: 1B7F  DD6E12      LD    L,(IX+VBLEG)
5146: 1B82  2C          INC   L                   ; SKIP OVER LINK AND TIME
5147: 1B83  2C          INC   L

```

Home Video Game System Library - Gunfight

```

5148: 1B84          SYSTEM VWRITR          ; AND WRITE LEG
5148: 1B84 FF          RST      56
5148: 1B85 1E          DB      VWRITR
5148: 0000          IF      VWRITR = INTPC
5148:                ENDIF
5148: 1B86          ENDM
5148:
5149:                ; IS GUNFIGHTER DEAD?
5150: 1B86 DDCB016E      BIT      VBSINT,(IX+VBSTAT)
5151: 1B8A 2030          JR      NZ,$+(GFWRT5-$) ; JUMP IF SO
5152: 1B8C 21D71D      LD      HL,ARMTBL      ; LOOKUP ARM PATTERN
5153: 1B8F 1600          LD      D,0
5154: 1B91 DD5E0F      LD      E,(IX+VBARM)
5155: 1B94 19          ADD     HL,DE
5156: 1B95 5E          LD      E,(HL)
5157: 1B96 23          INC     HL
5158: 1B97 56          LD      D,(HL)
5159: 1B98 EB          EX     DE,HL
5160: 1B99          SYSTEM VWRITR          ; WRITE ARM PATTERN
5160: 1B99 FF          RST      56
5160: 1B9A 1E          DB      VWRITR
5160: 0000          IF      VWRITR = INTPC
5160:                ENDIF
5160: 1B9B          ENDM
5160:
5161: 1B9B 210C1F      LD      HL,GFBODY      ; LOAD BODY PATTERN
5162: 1B9E 1808          JR      $+(GFWRT2-$)
5163:                ; BLANK THE WAGON
5164: 1BA0 110416      GFWRT1 LD      DE,1604H      ; LOAD WAGON SIZE
5165: 1BA3          SYSTEM VBLANK
5165: 1BA3 FF          RST      56
5165: 1BA4 28          DB      VBLANK
5165: 0000          IF      VBLANK = INTPC
5165:                ENDIF
5165: 1BA5          ENDM
5165:
5166: 1BA5 213C1F      LD      HL,WAGPAT
5167: 1BA8          GFWRT2 SYSTEM VWRITR          ; NOW WRITE
5167: 1BA8 FF          RST      56
5167: 1BA9 1E          DB      VWRITR
5167: 0000          IF      VWRITR = INTPC
5167:                ENDIF
5167: 1BAA          ENDM
5167:
5168: 1BAA DD720E      GFWRT4 LD      (IX+VBOAH),D
5169: 1BAD DD730D      LD      (IX+VBOAL),E
5170: 1BB0 21154F      GFWRT3 LD      HL,VECQ      ; ADD VECTOR TO VECTOR Q
5171: 1BB3 CD501D      CALL   ADDTQ
5172: 1BB6 DDE1          POP     IX
5173: 1BB8 08          EX     AF,AF'
5174: 1BB9 D9          EXX
5175: 1BBA FB          EIRE   EI
5176: 1BBB C9          RET
5177: 1BBC 21081F      GFWRT5 LD      HL,NULPAT
5178: 1BBF 18E7          JR      $+(GFWRT2-$)
5179:                ; *****
5180:                ; * GUNFIGHT LOW FOREGROUND ROUTINE *

```

Home Video Game System Library - Gunfight

```

5181:                ; *****
5182: 1BC1  F5          GFLFR  PUSH  AF
5183: 1BC2  C5          PUSH  BC
5184: 1BC3  D5          PUSH  DE
5185: 1BC4  E5          PUSH  HL
5186: 1BC5  DDE5        PUSH  IX
5187:                ; BUMP TIME BASES OF ACTIVE OR INTERCEPTED VECTORS
5188: 1BC7  21194F       LD    HL,BULV1+VBSTAT
5189: 1BCA  111100       LD    DE,BULVSZ-1
5190: 1BCD  0604         LD    B,4
5191: 1BCF  CD1A1D       CALL  TBUMP
5192: 1BD2  23           INC   HL                ; SKIP LINK FIELD
5193: 1BD3  111600       LD    DE,GFVSIZ-1
5194: 1BD6  0603         LD    B,3
5195: 1BD8  CD1A1D       CALL  TBUMP
5196:                ; LOOP TO UNWRITE, THE WRITE ALL FOUR BULLETS
5197:                ; BUT FIRST, A WORD TO OUR SHIFTER
5198: 1BDB  AF           XOR   A
5199: 1BDC  32FF0F       LD    (WASTE),A
5200: 1BDF  0604         LD    B,4
5201: 1BE1  DD21184F     LD    IX,BULV1
5202:                ; UNWRITE THIS GUY?
5203: 1BE5  DDCB0176     WRBUL1 BIT  VBBLNK,(IX+VBSTAT)
5204: 1BE9  2811         JR    Z,$+(WRBUL2-$)    ; JUMP IF NOT
5205: 1BEB  DD660E       LD    H,(IX+VBOAH)
5206: 1BEE  DD6E0D       LD    L,(IX+VBOAL)
5207: 1BF1  DD7E0F       LD    A,(IX+VBARM)     ; GET LAST MR
5208: 1BF4  D30C        OUT  (MAGIC),A
5209: 1BF6  36C0         LD    (HL),0C0H        ; UNWRITE BULLET
5210: 1BF8  DDCB01B6     RES  VBBLNK,(IX+VBSTAT) ; CLEAR BLANK BIT
5211:                ; SHALL WE WRITE THIS GUY?
5212: 1BFC  DDCB017E     WRBUL2 BIT  VBSACT,(IX+VBSTAT)
5213: 1C00  282B         JR    Z,$+(WRBUL4-$)
5214: 1C02  DD560B       LD    D,(IX+VBYH)
5215: 1C05  DD5E06       LD    E,(IX+VBXH)
5216: 1C08  DD7E00       LD    A,(IX+VBMR)
5217: 1C0B                SYSTEM RELABS
5217: 1C0B  FF           RST   56
5217: 1C0C  38           DB    RELABS
5217: 0000                IF    RELABS = INTPC
5217:                ENDIF
5217: 1C0D                ENDM
5217:
5218: 1C0D  DD720E       LD    (IX+VBOAH),D
5219: 1C10  DD730D       LD    (IX+VBOAL),E
5220: 1C13  DD770F       LD    (IX+VBARM),A
5221: 1C16  210040      LD    HL,NORMEM-SCREEN
5222: 1C19  19           ADD  HL,DE
5223: 4FFF                DIFER EQU  WASTE-SCREEN+NORMEM
5224: 1C1A  7E           LD    A,(HL)
5225: 1C1B  EB           EX   DE,HL
5226: 1C1C  36C0         LD    (HL),0C0H
5227: 1C1E  B7           OR   A
5228: 1C1F  2808         JR    Z,$+(WRBUL3-$)    ; JUMP IF NOT
5229: 1C21  DDCB01BE     RES  VBSACT,(IX+VBSTAT) ; KILL ACTIVE BIT
5230: 1C25  DDCB01EE     SET  VBSINT,(IX+VBSTAT) ; SET INTERCEPT LINE
5231: 1C29  DDCB01F6     WRBUL3 SET  VBBLNK,(IX+VBSTAT) ; SET BLANK BIT

```

Home Video Game System Library - Gunfight

```

5232:                ; STEP TO NEXT BULLET VECTOR    LOOP BACK IF DONE
5233: 1C2D  111200      WRBUL4  LD    DE,BULVSZ
5234: 1C30  DD19                ADD    IX,DE
5235: 1C32  10B1                DJNZ  $+(WRBUL1-$)
5236:                ; GET NEXT PATTERN TO WRITE, AND SCHEDULE HIM
5237: 1C34  21124F                LD    HL,WRITQ
5238: 1C37  CD671D                CALL  FIRST
5239: 1C3A  2812                JR    Z,$+(WRBL5A-$)      ; JUMP IF EMPTY Q
5240: 1C3C  3E76                LD    A,WRTVEC AND OFFH  ; SET FEEDBACK REG
5241: 1C3E  D30D                OUT   (INFBK),A
5242: 1C40  DD7E0B                LD    A,(IX+VBYH)        ; WHICH WINDOW TO USE?
5243: 1C43  FE32                CP    WINBND             ; COMPARE TO WINDOW
BOUNDRY
5244: 1C45  3E00                LD    A,BOTLIN           ; ASSUME BOTTOM LINE
5245: 1C47  3002                JR    NC,$+(WRBUL5-$)    ; JUMP IF GOOD GUESS
5246: 1C49  3E6A                LD    A,TOPLIN           ; WRONG - USE TOP
5247: 1C4B  D30F                WRBUL5 OUT (INLIN),A     ; SET LINE REGISTER
5248: 1C4D  FB                    EI
5249:                ; LOOP THRU VECTORING THOSE BULLETS
5250: 1C4E  DD21184F      WRBL5A LD    IX,BULV1
5251: 1C52  0604                LD    B,4
5252: 1C54  218B1D                LD    HL,BULLMT         ; HL = BULLET LIMITS TABLE
5253: 1C57  111200                LD    DE,BULVSZ
5254: 1C5A  DDCB017E      WRBUL6 BIT  VBSACT,(IX+VBSTAT) ; ACTIVE BULLET?
5255: 1C5E  280C                JR    Z,$+(WRBUL7-$)
5256: 1C60                SYSTEM VECT
5256: 1C60  FF                    RST   56
5256: 1C61  3E                    DB    VECT
5256: 0000                IF    VECT = INTPC
5256:                ENDIF
5256: 1C62                ENDM
5256:
5257: 1C62  DDCB075E                BIT  VBCLAT,(IX+VBXCHK) ; DID Y HIT EDGE?
5258: 1C66  2804                JR    Z,$+(WRBUL7-$)    ; NOPE
5259: 1C68  DDCB01BE                RES  VBSACT,(IX+VBSTAT) ; DEACTIVATE BULLET
5260: 1C6C  DD19                WRBUL7 ADD  IX,DE
5261: 1C6E  10EA                DJNZ  $+(WRBUL6-$)      ; LOOP BACK
5262:                ; NOW PUT SOMETHING ON THE WRITE Q
5263: 1C70  0602                LD    B,2                ; MAX 2 TIMES THRU
5264: 1C72  21154F                LD    HL,VECQ
5265: 1C75  CD671D                GVECT CALL  FIRST         ; FIRST VECTOR Q ENTRY
5266: 1C78  CAF81C                JP    Z,GVECT4           ; JUMP IS Q EMPTY
5267: 1C7B  CD251D                CALL  DELQ               ; DROP FROM VECTOR Q
5268: 1C7E  FB                    EI
5269:                ; WAGON?
5270: 1C7F  DDCB0146                BIT  VBSWAG,(IX+VBSTAT)
5271: 1C83  C2031D                JP    NZ,GVECT5         ; JUMP ON WAGON
5272:                ; DEAD?
5273: 1C86  DDCB016E                BIT  VBSINT,(IX+VBSTAT)
5274: 1C8A  2025                JR    NZ,$+(GVECT1-$)
5275:                ; ZERO VELOCITY?
5276: 1C8C  DD7E03                LD    A,(IX+VBDXL)
5277: 1C8F  DDB604                OR    (IX+VBDXH)
5278: 1C92  DDB608                OR    (IX+VBDYL)
5279: 1C95  DDB609                OR    (IX+VBDYH)
5280: 1C98  2017                JR    NZ,$+(GVECT1-$)   ; GVECT1 IF NONZERO
5281: 1C9A  DD7702                LD    (IX+VBTIMB),A     ; ZERO TIME BASE

```

Home Video Game System Library - Gunfight

```

5282: 1C9D DDCB0166          BIT   VBSNOM,(IX+VBSTAT) ; ALREADY STATIONARY
5283: 1CA1 2036              JR    NZ,$+(GVEC3A-$)
5284:                ; SET STATIONARY LEGS
5285: 1CA3 DD36124B          LD    (IX+VBLEG),LEGO AND 0FFH
5286: 1CA7 DDCB01DE          SET   VBSCHG,(IX+VBSTAT) ; SET CHANGED
5287: 1CAB DDCB01E6          SET   VBSNOM,(IX+VBSTAT) ; AND STATIONARY
5288: 1CAF 1828              JR    $+(GVEC3A-$)      ; JUMP TO ARM CHECK
5289:                ; MOVING GUNFIGHTER
5290:                ; VECTOR
5291: 1CB1 21831D          GVECT1 LD  HL,GUNLMT          ; LOAD GF LIMITS
5292: 1CB4                SYSTEM VECT
5292: 1CB4 FF                RST   56
5292: 1CB5 3E                DB    VECT
5292: 0000                IF    VECT = INTPC
5292:                ENDF
5292: 1CB6                ENDM
5292:
5293: 1CB6 2808              JR    Z,$+(GVECT2-$)
5294: 1CB8 DDCB01DE          SET   VBSCHG,(IX+VBSTAT) ; SET CHANGED BIT
5295: 1CBC DDCB01A6          RES   VBSNOM,(IX+VBSTAT) ; CLEAR NOT MOVING STATUS
5296:                ; NEED WE GO TO NEXT CELL IN ANIMATION SEQUENCE?
5297: 1CC0 DD7E11          GVECT2 LD  A,(IX+VBLEGT)    ; A = ANIMATION TIMER
5298: 1CC3 91                SUB   C                    ; SUBTRACT TIME BASE
5299: 1CC4 F2D61C          JP    P,GVECT3            ; JUMP IF NOT COUNTED
DOWN
5300:                ; GET NEXT CELL
5301: 1CC7 DD5E12          LD    E,(IX+VBLEG)        ; GET LINK
5302: 1CCA 161E          LD    D,LEGO SHR 8        ; SET H.0. PART
5303: 1CCC 1A                LD    A,(DE)              ; A = NEXT
5304: 1CCD DD7712          LD    (IX+VBLEG),A
5305: 1CD0 13                INC   DE                    ; STEP TO TIMER
5306: 1CD1 1A                LD    A,(DE)              ; GET NEW TIMER
5307: 1CD2 DDCB01DE          SET   VBSCHG,(IX+VBSTAT) ; SET CHANGED BIT
5308: 1CD6 DD7711          GVECT3 LD  (IX+VBLEGT),A  ; STORE BACK TIMER
5309:                ; DID ARM CHANGE?
5310: 1CD9 DD7E0F          GVEC3A LD  A,(IX+VBARM)
5311: 1CDC DDBE10          CP    (IX+VBOARM)         ; COMPARE TO OLD ARM
5312: 1CDF 2807              JR    Z,$+(GVEC3B-$)      ; JUMP IF NO CHANGE
5313: 1CE1 DDCB01DE          SET   VBSCHG,(IX+VBSTAT) ; SET CHANGED BIT
5314: 1CE5 DD7710          LD    (IX+VBOARM),A
5315:                ; ADD ITEM TO WRITE Q?
5316: 1CE8 DDCB015E          GVEC3B BIT  VBSCHG,(IX+VBSTAT)
5317: 1CEC 2020              JR    NZ,$+(GVECT6-$)    ; YES GVECT6
5318:                ; NO CHANGE - LINK TO VECTOR Q
5319: 1CEE 21154F          LD    HL,VECQ
5320: 1CF1 CD501D          CALL  ADDTQ
5321: 1CF4 05                DEC   B
5322: 1CF5 C2751C          JP    NZ,GVECT            ; SUB FOR DJNZ
5323: 1CF8 FB                GVECT4 EI
5324: 1CF9 CD0002          CALL  STIMER
5325: 1CFC DDE1              POP   IX
5326: 1CFE E1                POP   HL
5327: 1CFF D1                POP   DE
5328: 1D00 C1                POP   BC
5329: 1D01 F1                POP   AF
5330: 1D02 C9                RET
5331:                ; VECTOR AND Q WAGON

```

Home Video Game System Library - Gunfight

```

5332: 1D03 21781D      GVECT5 LD    HL,WAGLMT
5333: 1D06                SYSTEM VECT
5333: 1D06 FF                RST    56
5333: 1D07 3E                DB     VECT
5333: 0000                IF     VECT = INTPC
5333:                ENDIF
5333: 1D08                ENDM
5333:
5334: 1D08 21154F          LD    HL,VE CQ
5335: 1D0B CD251D          CALL DELQ                ; REMOVE FROM VECTOR Q
5336: 1D0E DDCB019E      GVECT6 RES    VBSC HG,(IX+VBSTAT)
5337: 1D12 21124F          LD    HL,WRITQ
5338: 1D15 CD501D          CALL ADDTQ
5339: 1D18 18DE                JR    $+(GVECT4-$)      ; JUMP BACK TO QUIT
5340:                ; ROUTINE TO BUMP TIME BASE OF VECTORS
5341: 1D1A 7E                TBUMP LD    A,(HL)        ; GET STATUS
5342: 1D1B 23                INC    HL
5343: 1D1C E6A0          AND    0A0H            ; ACTIVE OR INTERCEPTED?
5344: 1D1E 2801          JR    Z,$+(TBUMP1-$)   ; NO - TBUMP1
5345: 1D20 34                INC    (HL)            ; BUMP THE TIME BASE
5346: 1D21 19                TBUMP1 ADD   HL,DE
5347: 1D22 10F6          DJNZ  $+(TBUMP-$)
5348: 1D24 C9                RET
5349:                ; SUBROUTINE TO DELETE ENTRY AT FRONT OF Q
5350:                ; ENTRY:      HL = HEAD-TAIL, IX = OBJECT, A = CLOBBERE
5351: 1D25 F3                DELQ  DI
5352: 1D26 DD7EFF          LD    A,(IX+NEXT)      ; HEAD = NEXT(OBJECT)
5353: 1D29 77                LD    (HL),A
5354: 1D2A A7                AND    A                ; IS HEAD NOW NIL?
5355: 1D2B C0                RET    NZ                ; QUIT IF NOT
5356: 1D2C 23                INC    HL                ; YES - SET TAIL = NIL
TOO
5357: 1D2D 77                LD    (HL),A
5358: 1D2E 2B                DEC    HL
5359: 1D2F C9                RET
5360: 1D30 DD360332      COWINT LD    (IX+VBDXL),50    ; SLOW WALK OUT
5361: 1D34 DD360180          LD    (IX+VBSTAT),80H  ; ACTIVE
5362: 1D38 DD360701          LD    (IX+VBXCHK),1
5363: 1D3C DD360C01          LD    (IX+VBYCHK),1
5364: 1D40 DD360604          LD    (IX+VBXH),4
5365: 1D44 DD360B28          LD    (IX+VBYH),40
5366: 1D48 DD360F06          LD    (IX+VBARM),6     ; SET ARM STRAIGHT
5367: 1D4C DD36124B          LD    (IX+VBLEG),LEGO AND 0FFH
5368:                ; JP    ADDTQ
5369:                ; SUBROUTINE TO APPEND ENTRY TO END OF Q
5370:                ; ENTRY:      HL = HEAD-TAIL BYTES, IX = OBJECT, A,DE C
5371: 1D50 DDE5          ADDTQ PUSH  IX                ; DE = ENTRY
5372: 1D52 D1                POP   DE
5373: 1D53 F3                DI
5374: 1D54 DD36FF00          LD    (IX+NEXT),0      ; NEXT(OBJ)=NIL
5375: 1D58 23                INC    HL
5376: 1D59 7E                LD    A,(HL)          ; A= OLD TAIL
5377: 1D5A 73                LD    (HL),E          ; SET TAIL = .OBJ
5378: 1D5B A7                AND    A                ; WAS OLD TAIL NIL?
5379: 1D5C 2806          JR    Z,$+(ADDTQ1-$)  ; JUMP IF SO
5380:                ; NONNIL OLD TAIL SET NEXT(OLDTAIL)=.OBJ
5381: 1D5E 5F                LD    E,A                ; DE = .NEXT(OLDTAIL)

```

Home Video Game System Library - Gunfight

```

5382: 1D5F 7E          LD    A,(HL)          ; A = OBJ (FROM NEW TAIL)
5383: 1D60 2B          DEC    HL
5384: 1D61 1B          DEC    DE
5385: 1D62 12          LD    (DE),A
5386: 1D63 C9          RET
5387:                ; NIL OLD TAIL CASE
5388: 1D64 2B          ADDTQ1 DEC    HL          ; BACKUP TO HEAD
5389: 1D65 73          LD    (HL),E          ; HEAD = .OBJ
5390: 1D66 C9          RET
5391:                ; SUBROUTINE TO POINT IX AT FIRST ENTRY ON A Q
5392:                ; ENTRY:      HL = Q HEAD-TAIL
5393:                ; EXIT:      IX,DE = OBJECT, A = L.O. BYTE OF OBJECT
5394:                ;                NONZERO STATUS SET IF Q NOT EMPTY
5395: 1D67 F3          FIRST  DI
5396: 1D68 5E          LD    E,(HL)
5397: 1D69 23          INC    HL
5398: 1D6A 23          INC    HL
5399: 1D6B 56          LD    D,(HL)          ; D = H.O. ADDR. BYTE
5400: 1D6C 2B          DEC    HL
5401: 1D6D 2B          DEC    HL
5402: 1D6E 7B          LD    A,E            ; E = HEAD OF Q
5403: 1D6F A7          AND    A
5404: 1D70 D5          PUSH  DE
5405: 1D71 DDE1        POP    IX
5406: 1D73 C9          RET
5407:                ;
5408:                ; *****
5409:                ; * GUNFIGHT CONSTANTS *
5410:                ; *****
5411: 1D74                ORG    ($+1) AND 0FFFEH
5412: 1D74                INTTBL
5413: 1D74 C11B        LFRVEC DW    GFLFR
5414: 1D76 591B        WRTVEC DW    GFWRIT
5415:                ; WAGON LIMITS TABLE
5416: 1D78 0A          WAGLMT DB    TLINE
5417: 1D79 44          DB    BLINE-24
5418: 1D7A 47455420    GETRDY DB    'G','E','T',' ','R','E','A','D','Y'
5419:                52454144
5420:                59
5419:                ; GUNFIGHTER LIMITS
5420: 1D83 00          GUNLMT DB    0
5421: 1D84 2F          DB    LCACX-17
5422: 1D85 0A          DB    TLINE
5423: 1D86 48          DB    BLINE-20
5424: 1D87 44524157    DRAW  DB    'D','R','A','W'
5425:                ; BULLET LIMITS
5426: 1D8B 00          BULLMT DB    0
5427: 1D8C 9F          DB    159
5428: 1D8D 09          DB    ALINE
5429: 1D8E 5B          DB    BLINE-1
5430:                BN    MACRO DX, ARMX, DY, ARMY
5431:                DW    DX
5432:                DB    ARMX
5433:                DW    DY
5434:                DB    ARMY
5435:                ENDM
5436: 1D8F                BULTAB BN    768, 15, 768, 15

```

Home Video Game System Library - Gunfight

5436:	1D8F	0003	DW	768
5436:	1D91	0F	DB	15
5436:	1D92	0003	DW	768
5436:	1D94	0F	DB	15
5436:	1D95		ENDM	
5436:				
5437:	1D95		BN	1024, 15, 512, 12
5437:	1D95	0004	DW	1024
5437:	1D97	0F	DB	15
5437:	1D98	0002	DW	512
5437:	1D9A	0C	DB	12
5437:	1D9B		ENDM	
5437:				
5438:	1D9B		BN	1024, 15, 256, 11
5438:	1D9B	0004	DW	1024
5438:	1D9D	0F	DB	15
5438:	1D9E	0001	DW	256
5438:	1DA0	0B	DB	11
5438:	1DA1		ENDM	
5438:				
5439:	1DA1		BN	1024, 15, 0, 8
5439:	1DA1	0004	DW	1024
5439:	1DA3	0F	DB	15
5439:	1DA4	0000	DW	0
5439:	1DA6	08	DB	8
5439:	1DA7		ENDM	
5439:				
5440:	1DA7		BN	1024, 15, -256, 6
5440:	1DA7	0004	DW	1024
5440:	1DA9	0F	DB	15
5440:	1DAA	00FF	DW	-256
5440:	1DAC	06	DB	6
5440:	1DAD		ENDM	
5440:				
5441:	1DAD		BN	1024, 15, -512, 4
5441:	1DAD	0004	DW	1024
5441:	1DAF	0F	DB	15
5441:	1DB0	00FE	DW	-512
5441:	1DB2	04	DB	4
5441:	1DB3		ENDM	
5441:				
5442:	1DB3		BN	768, 15, -768, 3
5442:	1DB3	0003	DW	768
5442:	1DB5	0F	DB	15
5442:	1DB6	00FD	DW	-768
5442:	1DB8	03	DB	3
5442:	1DB9		ENDM	
5442:				
5443:	1DB9		LFTAB	DEF5 72, 22, 44, 67, 14
5443:	1DB9	48	DB	72
5443:	1DBA	16	DB	22
5443:	1DBB	2C	DB	44
5443:	1DBC	43	DB	67
5443:	1DBD	0E	DB	14
5443:	1DBE		ENDM	
5443:				
5444:	1DBE		RFTAB	DEF5 18, 68, 40, 13, 63

Home Video Game System Library - Gunfight

```

5444: 1DBE 12          DB      18
5444: 1DBF 44          DB      68
5444: 1DC0 28          DB      40
5444: 1DC1 0D          DB      13
5444: 1DC2 3F          DB      63
5444: 1DC3              ENDM
5444:
5445: 1DC3 A2          GFCOLS DB    0A2H   ; WAS 9DH - NOW MATCHES 3159 ROM
5446: 1DC4 7D          DB    7DH     ; WAS 76H - NOW MATCHES 3159 ROM
5447: 1DC5 0B          DB    0BH     ; WAS FCH - NOW MATCHES 3159 ROM
5448: 1DC6 87          DB    87H
5449: 1DC7 A2          DB    0A2H   ; WAS 9DH - NOW MATCHES 3159 ROM
5450: 1DC8 7D          DB    7DH     ; WAS 76H - NOW MATCHES 3159 ROM
5451: 1DC9 6C          DB    6CH
5452: 1DCA 87          DB    87H
5453: 1DCB              SINIT  DEF8    6, 6, 0, 0, 0, 30H, 30H, 0
5453: 1DCB 06          DB      6
5453: 1DCC 06          DB      6
5453: 1DCD 00          DB      0
5453: 1DCE 00          DB      0
5453: 1DCF 00          DB      0 ; 'GE' reserved, so used '0'
5453: 1DD0 30          DB    30H
5453: 1DD1 30          DB    30H
5453: 1DD2 00          DB      0
5453: 1DD3              ENDM
5453:
5454: 1DD3              DEF4    0, 80H, 0FH, 0FH
5454: 1DD3 00          DB      0
5454: 1DD4 80          DB    80H
5454: 1DD5 0F          DB    0FH
5454: 1DD6 0F          DB    0FH
5454: 1DD7              ENDM
5454:
5455: 0007              NUMB   EQU    00000111B          ; COLOR MASK
5456: 000B              BULT   EQU    00001011B
5457: 000B              TIME   EQU    00001011B
5458: 000B              LARGE  EQU    00001011B
5459: 000C              LARG2  EQU    00001100B
5460:                  ;
5461:                  ; *****
5462:                  ; * GUN FIGHT PATTERNS *
5463:                  ; *****
5464:                  ;
5465:                  ; PATTERN TABLES
5466: 1DD7 F81D          ARMTBL DW    ARM0
5467: 1DD9 061E          DW    ARM1
5468: 1DDB 101E          DW    ARM2
5469: 1DDD 181E          DW    ARM3
5470: 1DDF 241E          DW    ARM4
5471: 1DE1 321E          DW    ARM5
5472: 1DE3 421E          DW    ARM6
5473:                  ; PATTERN DEFINITION MACROS
5474: DEF02  MACRO AA, BB
5475:          DB    $AA
5476:          DB    $BB
5477:          ENDM
5478: DEF03  MACRO AA, BB, CC

```

Home Video Game System Library - Gunfight

```

5479:          DB      $AA
5480:          DB      $BB
5481:          DB      $CC
5482:          ENDM
5483:          DEF04  MACRO AA, BB, CC, DD
5484:          DB      $AA
5485:          DB      $BB
5486:          DB      $CC
5487:          DB      $DD
5488:          ENDM
5489: 1DE5          TREE  DEF2  1, 17
5489: 1DE5  01          DB      1
5489: 1DE6  11          DB      17
5489: 1DE7          ENDM
5489:
5490: 1DE7  08          DB      00001000B
5491: 1DE8  1C          DB      00011100B
5492: 1DE9  3E          DB      00111110B
5493: 1DEA  6B          DB      01101011B
5494: 1DEB  08          DB      00001000B
5495: 1DEC  08          DB      00001000B
5496: 1DED  3C          DB      00111100B
5497: 1DEE  7E          DB      01111110B
5498: 1DEF  A9          DB      10101001B
5499: 1DF0  08          DB      00001000B
5500: 1DF1  3C          DB      00111100B
5501: 1DF2  7E          DB      01111110B
5502: 1DF3  EB          DB      11101011B
5503: 1DF4  89          DB      10001001B
5504: 1DF5  08          DB      00001000B
5505: 1DF6  1C          DB      00011100B
5506: 1DF7  AE          DB      10101110B
5507: 1DF8          ARMO  DEF04  0A, 0A, 2, 5
5507: 1DF8  0A          DB      $0A
5507: 1DF9  0A          DB      $0A
5507: 1DFA  02          DB      $2
5507: 1DFB  05          DB      $5
5507: 1DFC          ENDM
5507:
5508: 1DFC          DEF02  40, 00
5508: 1DFC  40          DB      $40
5508: 1DFD  00          DB      $00
5508: 1DFE          ENDM
5508:
5509: 1DFE          DEF02  51, 00
5509: 1DFE  51          DB      $51
5509: 1DFE  00          DB      $00
5509: 1E00          ENDM
5509:
5510: 1E00          DEF02  04, 00
5510: 1E00  04          DB      $04
5510: 1E01  00          DB      $00
5510: 1E02          ENDM
5510:
5511: 1E02          DEF02  01, 00
5511: 1E02  01          DB      $01
5511: 1E03  00          DB      $00

```

Home Video Game System Library - Gunfight

```

5511: 1E04          ENDM
5511:
5512: 1E04          DEF02 00, 40
5512: 1E04 00       DB    $00
5512: 1E05 40       DB    $40
5512: 1E06          ENDM
5512:
5513: 1E06          ARM1 DEF04 0A, 0A, 2, 3
5513: 1E06 0A       DB    $0A
5513: 1E07 0A       DB    $0A
5513: 1E08 02       DB    $2
5513: 1E09 03       DB    $3
5513: 1E0A          ENDM
5513:
5514: 1E0A          DEF02 50, 00
5514: 1E0A 50       DB    $50
5514: 1E0B 00       DB    $00
5514: 1E0C          ENDM
5514:
5515: 1E0C          DEF02 14, 00
5515: 1E0C 14       DB    $14
5515: 1E0D 00       DB    $00
5515: 1E0E          ENDM
5515:
5516: 1E0E          DEF02 01, 40
5516: 1E0E 01       DB    $01
5516: 1E0F 40       DB    $40
5516: 1E10          ENDM
5516:
5517: 1E10          ARM2 DEF04 0A, 0A, 2, 2
5517: 1E10 0A       DB    $0A
5517: 1E11 0A       DB    $0A
5517: 1E12 02       DB    $2
5517: 1E13 02       DB    $2
5517: 1E14          ENDM
5517:
5518: 1E14          DEF02 54, 00
5518: 1E14 54       DB    $54
5518: 1E15 00       DB    $00
5518: 1E16          ENDM
5518:
5519: 1E16          DEF02 55,40
5519: 1E16 55       DB    $55
5519: 1E17 40       DB    $40
5519: 1E18          ENDM
5519:
5520: 1E18          ARM3 DEF04 0A, 7, 2, 4
5520: 1E18 0A       DB    $0A
5520: 1E19 07       DB    $7
5520: 1E1A 02       DB    $2
5520: 1E1B 04       DB    $4
5520: 1E1C          ENDM
5520:
5521: 1E1C          DEF02 10, 00
5521: 1E1C 10       DB    $10
5521: 1E1D 00       DB    $00
5521: 1E1E          ENDM

```

Home Video Game System Library - Gunfight

5521:			
5522:	1E1E		DEF02 05, 40
5522:	1E1E	05	DB \$05
5522:	1E1F	40	DB \$40
5522:	1E20		ENDM
5522:			
5523:	1E20		DEF02 54, 00
5523:	1E20	54	DB \$54
5523:	1E21	00	DB \$00
5523:	1E22		ENDM
5523:			
5524:	1E22		DEF02 50, 00
5524:	1E22	50	DB \$50
5524:	1E23	00	DB \$00
5524:	1E24		ENDM
5524:			
5525:	1E24		DEF04 0A, 6, 2, 5
5525:	1E24	0A	DB \$0A
5525:	1E25	06	DB \$6
5525:	1E26	02	DB \$2
5525:	1E27	05	DB \$5
5525:	1E28		ENDM
5525:			
5526:	1E28		DEF02 00, 40
5526:	1E28	00	DB \$00
5526:	1E29	40	DB \$40
5526:	1E2A		ENDM
5526:			
5527:	1E2A		DEF02 45, 00
5527:	1E2A	45	DB \$45
5527:	1E2B	00	DB \$00
5527:	1E2C		ENDM
5527:			
5528:	1E2C		DEF02 10, 00
5528:	1E2C	10	DB \$10
5528:	1E2D	00	DB \$00
5528:	1E2E		ENDM
5528:			
5529:	1E2E		DEF02 50, 00
5529:	1E2E	50	DB \$50
5529:	1E2F	00	DB \$00
5529:	1E30		ENDM
5529:			
5530:	1E30		DEF02 40, 00
5530:	1E30	40	DB \$40
5530:	1E31	00	DB \$00
5530:	1E32		ENDM
5530:			
5531:	1E32		DEF04 0A, 5, 2, 6
5531:	1E32	0A	DB \$0A
5531:	1E33	05	DB \$5
5531:	1E34	02	DB \$2
5531:	1E35	06	DB \$6
5531:	1E36		ENDM
5531:			
5532:	1E36		DEF02 00, 40
5532:	1E36	00	DB \$00

Home Video Game System Library - Gunfight

```

5532: 1E37 40          DB    $40
5532: 1E38          ENDM
5532:
5533: 1E38          DEF02 01, 00
5533: 1E38 01          DB    $01
5533: 1E39 00          DB    $00
5533: 1E3A          ENDM
5533:
5534: 1E3A          DEF02 05, 00
5534: 1E3A 05          DB    $05
5534: 1E3B 00          DB    $00
5534: 1E3C          ENDM
5534:
5535: 1E3C          DEF02 14, 00
5535: 1E3C 14          DB    $14
5535: 1E3D 00          DB    $00
5535: 1E3E          ENDM
5535:
5536: 1E3E          DEF02 54, 00
5536: 1E3E 54          DB    $54
5536: 1E3F 00          DB    $00
5536: 1E40          ENDM
5536:
5537: 1E40          DEF02 50, 00
5537: 1E40 50          DB    $50
5537: 1E41 00          DB    $00
5537: 1E42          ENDM
5537:
5538: 1E42          ARM6  DEF04 0A, 5, 1, 5
5538: 1E42 0A          DB    $0A
5538: 1E43 05          DB    $5
5538: 1E44 01          DB    $1
5538: 1E45 05          DB    $5
5538: 1E46          ENDM
5538:
5539: 1E46 01          DB    01H
5540: 1E47 44          DB    44H
5541: 1E48 10          DB    10H
5542: 1E49 40          DB    40H
5543: 1E4A 40          DB    40H
5544:                ; **** NOTE ****
5545:                ; THE FOLLOWING PATTERNS ARE CONSTRAINED TO EXIST ON THE
5546:                ; PAGE. THE FOLLOWING 'ORG' WILL DO IT FOR EXPERIMENTAL
5547:                ; PATTERNS ARE: LEG0, LEG1, LEG2, KIL1, KIL2
5548:                ;      ORG      ($+255) AND 0FF00H ; ***
5549: 1E4B 60          LEG0  DB    LEG1 AND 0FFH
5550: 1E4C 04          DB    4
5551: 1E4D          DEF04 0, 0F, 3, 5
5551: 1E4D 00          DB    $0
5551: 1E4E 0F          DB    $0F
5551: 1E4F 03          DB    $3
5551: 1E50 05          DB    $5
5551: 1E51          ENDM
5551:
5552: 1E51          DEF03 01, 55, 0
5552: 1E51 01          DB    $01
5552: 1E52 55          DB    $55

```

Home Video Game System Library - Gunfight

5552:	1E53	00		DB	\$0
5552:	1E54			ENDM	
5552:					
5553:	1E54			DEF03	05, 45, 40
5553:	1E54	05		DB	\$05
5553:	1E55	45		DB	\$45
5553:	1E56	40		DB	\$40
5553:	1E57			ENDM	
5553:					
5554:	1E57			DEF03	15, 01, 40
5554:	1E57	15		DB	\$15
5554:	1E58	01		DB	\$01
5554:	1E59	40		DB	\$40
5554:	1E5A			ENDM	
5554:					
5555:	1E5A			DEF03	50, 01, 40
5555:	1E5A	50		DB	\$50
5555:	1E5B	01		DB	\$01
5555:	1E5C	40		DB	\$40
5555:	1E5D			ENDM	
5555:					
5556:	1E5D			DEF03	15, 00, 54
5556:	1E5D	15		DB	\$15
5556:	1E5E	00		DB	\$00
5556:	1E5F	54		DB	\$54
5556:	1E60			ENDM	
5556:					
5557:	1E60	70	LEG1	DB	LEG2 AND 0FFH
5558:	1E61	04		DB	4
5559:	1E62			DEF04	2, 0F, 2, 5
5559:	1E62	02		DB	\$2
5559:	1E63	0F		DB	\$0F
5559:	1E64	02		DB	\$2
5559:	1E65	05		DB	\$5
5559:	1E66			ENDM	
5559:					
5560:	1E66			DEF02	15, 50
5560:	1E66	15		DB	\$15
5560:	1E67	50		DB	\$50
5560:	1E68			ENDM	
5560:					
5561:	1E68			DEF02	54, 50
5561:	1E68	54		DB	\$54
5561:	1E69	50		DB	\$50
5561:	1E6A			ENDM	
5561:					
5562:	1E6A			DEF02	50, 50
5562:	1E6A	50		DB	\$50
5562:	1E6B	50		DB	\$50
5562:	1E6C			ENDM	
5562:					
5563:	1E6C			DEF02	50, 50
5563:	1E6C	50		DB	\$50
5563:	1E6D	50		DB	\$50
5563:	1E6E			ENDM	
5563:					
5564:	1E6E			DEF02	55, 15

Home Video Game System Library - Gunfight

5564:	1E6E	55		DB	\$55
5564:	1E6F	15		DB	\$15
5564:	1E70			ENDM	
5564:					
5565:	1E70	4B	LEG2	DB	LEG0 AND 0FFH
5566:	1E71	04		DB	4
5567:	1E72			DEF04	3, 0F, 2, 5
5567:	1E72	03		DB	\$3
5567:	1E73	0F		DB	\$0F
5567:	1E74	02		DB	\$2
5567:	1E75	05		DB	\$5
5567:	1E76			ENDM	
5567:					
5568:	1E76			DEF02	55, 00
5568:	1E76	55		DB	\$55
5568:	1E77	00		DB	\$00
5568:	1E78			ENDM	
5568:					
5569:	1E78			DEF02	15, 00
5569:	1E78	15		DB	\$15
5569:	1E79	00		DB	\$00
5569:	1E7A			ENDM	
5569:					
5570:	1E7A			DEF02	15, 00
5570:	1E7A	15		DB	\$15
5570:	1E7B	00		DB	\$00
5570:	1E7C			ENDM	
5570:					
5571:	1E7C			DEF02	14, 00
5571:	1E7C	14		DB	\$14
5571:	1E7D	00		DB	\$00
5571:	1E7E			ENDM	
5571:					
5572:	1E7E			DEF02	05, 40
5572:	1E7E	05		DB	\$05
5572:	1E7F	40		DB	\$40
5572:	1E80			ENDM	
5572:					
5573:	1E80	D2	KIL1	DB	KIL2 AND 0FFH
5574:	1E81	14		DB	20
5575:	1E82			DEF04	0, 1, 4, 13
5575:	1E82	00		DB	\$0
5575:	1E83	01		DB	\$1
5575:	1E84	04		DB	\$4
5575:	1E85	13		DB	\$13
5575:	1E86			ENDM	
5575:					
5576:	1E86			DEF04	01, 10, 00, 00
5576:	1E86	01		DB	\$01
5576:	1E87	10		DB	\$10
5576:	1E88	00		DB	\$00
5576:	1E89	00		DB	\$00
5576:	1E8A			ENDM	
5576:					
5577:	1E8A			DEF04	45, 54, 40, 00
5577:	1E8A	45		DB	\$45
5577:	1E8B	54		DB	\$54

Home Video Game System Library - Gunfight

5577: 1E8C	40	DB	\$40
5577: 1E8D	00	DB	\$00
5577: 1E8E		ENDM	
5577:			
5578: 1E8E		DEF04	55, 55, 40, 00
5578: 1E8E	55	DB	\$55
5578: 1E8F	55	DB	\$55
5578: 1E90	40	DB	\$40
5578: 1E91	00	DB	\$00
5578: 1E92		ENDM	
5578:			
5579: 1E92		DEF04	0A, A8, 00, 00
5579: 1E92	0A	DB	\$0A
5579: 1E93	A8	DB	\$A8
5579: 1E94	00	DB	\$00
5579: 1E95	00	DB	\$00
5579: 1E96		ENDM	
5579:			
5580: 1E96		DEF04	0A, A2, 00, 01
5580: 1E96	0A	DB	\$0A
5580: 1E97	A2	DB	\$A2
5580: 1E98	00	DB	\$00
5580: 1E99	01	DB	\$01
5580: 1E9A		ENDM	
5580:			
5581: 1E9A		DEF04	0A, AA, 80, 14
5581: 1E9A	0A	DB	\$0A
5581: 1E9B	AA	DB	\$AA
5581: 1E9C	80	DB	\$80
5581: 1E9D	14	DB	\$14
5581: 1E9E		ENDM	
5581:			
5582: 1E9E		DEF04	02, AA, 00, 50
5582: 1E9E	02	DB	\$02
5582: 1E9F	AA	DB	\$AA
5582: 1EA0	00	DB	\$00
5582: 1EA1	50	DB	\$50
5582: 1EA2		ENDM	
5582:			
5583: 1EA2		DEF04	00, A8, 05, 40
5583: 1EA2	00	DB	\$00
5583: 1EA3	A8	DB	\$A8
5583: 1EA4	05	DB	\$05
5583: 1EA5	40	DB	\$40
5583: 1EA6		ENDM	
5583:			
5584: 1EA6		DEF04	05, 55, 54, 00
5584: 1EA6	05	DB	\$05
5584: 1EA7	55	DB	\$55
5584: 1EA8	54	DB	\$54
5584: 1EA9	00	DB	\$00
5584: 1EAA		ENDM	
5584:			
5585: 1EAA		DEF04	15, 55, 50, 00
5585: 1EAA	15	DB	\$15
5585: 1EAB	55	DB	\$55
5585: 1EAC	50	DB	\$50

Home Video Game System Library - Gunfight

5585:	1EAD	00	DB	\$00
5585:	1EAE		ENDM	
5585:				
5586:	1EAE		DEF04	54, 55, 50, 00
5586:	1EAE	54	DB	\$54
5586:	1EAF	55	DB	\$55
5586:	1EB0	50	DB	\$50
5586:	1EB1	00	DB	\$00
5586:	1EB2		ENDM	
5586:				
5587:	1EB2		DEF04	50, 05, 54, 00
5587:	1EB2	50	DB	\$50
5587:	1EB3	05	DB	\$05
5587:	1EB4	54	DB	\$54
5587:	1EB5	00	DB	\$00
5587:	1EB6		ENDM	
5587:				
5588:	1EB6		DEF04	50, 01, 55, 00
5588:	1EB6	50	DB	\$50
5588:	1EB7	01	DB	\$01
5588:	1EB8	55	DB	\$55
5588:	1EB9	00	DB	\$00
5588:	1EBA		ENDM	
5588:				
5589:	1EBA		DEF04	10, 01, 55, 40
5589:	1EBA	10	DB	\$10
5589:	1EBB	01	DB	\$01
5589:	1EBC	55	DB	\$55
5589:	1EBD	40	DB	\$40
5589:	1EBE		ENDM	
5589:				
5590:	1EBE		DEF04	10, 00, 05, 50
5590:	1EBE	10	DB	\$10
5590:	1EBF	00	DB	\$00
5590:	1EC0	05	DB	\$05
5590:	1EC1	50	DB	\$50
5590:	1EC2		ENDM	
5590:				
5591:	1EC2		DEF04	00, 00, 01, 50
5591:	1EC2	00	DB	\$00
5591:	1EC3	00	DB	\$00
5591:	1EC4	01	DB	\$01
5591:	1EC5	50	DB	\$50
5591:	1EC6		ENDM	
5591:				
5592:	1EC6		DEF04	00, 00, 00, 40
5592:	1EC6	00	DB	\$00
5592:	1EC7	00	DB	\$00
5592:	1EC8	00	DB	\$00
5592:	1EC9	40	DB	\$40
5592:	1ECA		ENDM	
5592:				
5593:	1ECA		DEF04	00, 00, 01, 40
5593:	1ECA	00	DB	\$00
5593:	1ECB	00	DB	\$00
5593:	1ECC	01	DB	\$01
5593:	1ECD	40	DB	\$40

Home Video Game System Library - Gunfight

5593:	1ECE		ENDM
5593:			
5594:	1ECE		DEF04 00, 00, 00, 54
5594:	1ECE	00	DB \$00
5594:	1ECF	00	DB \$00
5594:	1ED0	00	DB \$00
5594:	1ED1	54	DB \$54
5594:	1ED2		ENDM
5594:			
5595:	1ED2	D2	KIL2 DB KIL2 AND 0FFH
5596:	1ED3	3C	DB 60
5597:	1ED4		DEF04 0, D, 4, 7
5597:	1ED4	00	DB \$0
5597:	1ED5	0D	DB \$D
5597:	1ED6	04	DB \$4
5597:	1ED7	07	DB \$7
5597:	1ED8		ENDM
5597:			
5598:	1ED8		DEF04 01, 10, 00, 00
5598:	1ED8	01	DB \$01
5598:	1ED9	10	DB \$10
5598:	1EDA	00	DB \$00
5598:	1EDB	00	DB \$00
5598:	1EDC		ENDM
5598:			
5599:	1EDC		DEF04 45, 54, 40, 00
5599:	1EDC	45	DB \$45
5599:	1EDD	54	DB \$54
5599:	1EDE	40	DB \$40
5599:	1EDF	00	DB \$00
5599:	1EE0		ENDM
5599:			
5600:	1EE0		DEF04 55, 55, 40, 00
5600:	1EE0	55	DB \$55
5600:	1EE1	55	DB \$55
5600:	1EE2	40	DB \$40
5600:	1EE3	00	DB \$00
5600:	1EE4		ENDM
5600:			
5601:	1EE4		DEF04 0A, A8, 00, 00
5601:	1EE4	0A	DB \$0A
5601:	1EE5	A8	DB \$A8
5601:	1EE6	00	DB \$00
5601:	1EE7	00	DB \$00
5601:	1EE8		ENDM
5601:			
5602:	1EE8		DEF04 0A, 88, 15, 01
5602:	1EE8	0A	DB \$0A
5602:	1EE9	88	DB \$88
5602:	1EEA	15	DB \$15
5602:	1EEB	01	DB \$01
5602:	1EEC		ENDM
5602:			
5603:	1EEC		DEF04 16, A5, 55, 41
5603:	1EEC	16	DB \$16
5603:	1EED	A5	DB \$A5
5603:	1EEE	55	DB \$55

Home Video Game System Library - Gunfight

5603:	1EEF	41		DB	\$41
5603:	1EF0			ENDM	
5603:					
5604:	1EF0			DEF04	15, 55, 55, 55
5604:	1EF0	15		DB	\$15
5604:	1EF1	55		DB	\$55
5604:	1EF2	55		DB	\$55
5604:	1EF3	55		DB	\$55
5604:	1EF4			ENDM	
5604:					
5605:	1EF4		CACTUS	DEF2	1, 12
5605:	1EF4	01		DB	1
5605:	1EF5	0C		DB	12
5605:	1EF6			ENDM	
5605:					
5606:	1EF6	20		DB	00100000B
5607:	1EF7	30		DB	00110000B
5608:	1EF8	38		DB	00111000B
5609:	1EF9	30		DB	00110000B
5610:	1EFA	B2		DB	10110010B
5611:	1EFB	F2		DB	11110010B
5612:	1EFC	F6		DB	11110110B
5613:	1EFD	3C		DB	00111100B
5614:	1EFE	3C		DB	00111100B
5615:	1EFF	30		DB	00110000B
5616:	1F00	30		DB	00110000B
5617:	1F01	30		DB	00110000B
5618:	1F02	474F5420 4D45	GOTME	DB	'G','O','T',' ','M','E'
5619:	1F08	00	NULPAT	DB	0
5620:	1F09	00		DB	0
5621:	1F0A	01		DB	1
5622:	1F0B	01		DB	1
5623:	1F0C		GFBODY	DEF04	0, 0, 3, F
5623:	1F0C	00		DB	\$0
5623:	1F0D	00		DB	\$0
5623:	1F0E	03		DB	\$3
5623:	1F0F	0F		DB	\$F
5623:	1F10			ENDM	
5623:					
5624:	1F10			DEF03	00, 44, 00
5624:	1F10	00		DB	\$00
5624:	1F11	44		DB	\$44
5624:	1F12	00		DB	\$00
5624:	1F13			ENDM	
5624:					
5625:	1F13			DEF03	11, 55, 10
5625:	1F13	11		DB	\$11
5625:	1F14	55		DB	\$55
5625:	1F15	10		DB	\$10
5625:	1F16			ENDM	
5625:					
5626:	1F16			DEF03	15, 55, 50
5626:	1F16	15		DB	\$15
5626:	1F17	55		DB	\$55
5626:	1F18	50		DB	\$50
5626:	1F19			ENDM	

Home Video Game System Library - Gunfight

```

5626:
5627: 1F19          DEF03 02, AA, 00
5627: 1F19 02       DB    $02
5627: 1F1A AA       DB    $AA
5627: 1F1B 00       DB    $00
5627: 1F1C          ENDM
5627:
5628: 1F1C          DEF03 02, A2, 00
5628: 1F1C 02       DB    $02
5628: 1F1D A2       DB    $A2
5628: 1F1E 00       DB    $00
5628: 1F1F          ENDM
5628:
5629: 1F1F          DEF03 02, AA, 80
5629: 1F1F 02       DB    $02
5629: 1F20 AA       DB    $AA
5629: 1F21 80       DB    $80
5629: 1F22          ENDM
5629:
5630: 1F22          DEF03 00, AA, 00
5630: 1F22 00       DB    $00
5630: 1F23 AA       DB    $AA
5630: 1F24 00       DB    $00
5630: 1F25          ENDM
5630:
5631: 1F25          DEF03 00, A8, 00
5631: 1F25 00       DB    $00
5631: 1F26 A8       DB    $A8
5631: 1F27 00       DB    $00
5631: 1F28          ENDM
5631:
5632: 1F28          DEF03 15, 55, 00
5632: 1F28 15       DB    $15
5632: 1F29 55       DB    $55
5632: 1F2A 00       DB    $00
5632: 1F2B          ENDM
5632:
5633: 1F2B          DEF03 55, 55, 50
5633: 1F2B 55       DB    $55
5633: 1F2C 55       DB    $55
5633: 1F2D 50       DB    $50
5633: 1F2E          ENDM
5633:
5634: 1F2E          DEF03 51, 55, 50
5634: 1F2E 51       DB    $51
5634: 1F2F 55       DB    $55
5634: 1F30 50       DB    $50
5634: 1F31          ENDM
5634:
5635: 1F31          DEF03 41, 55, 00
5635: 1F31 41       DB    $41
5635: 1F32 55       DB    $55
5635: 1F33 00       DB    $00
5635: 1F34          ENDM
5635:
5636: 1F34          DEF03 41, 55, 00
5636: 1F34 41       DB    $41

```

Home Video Game System Library - Gunfight

5636:	1F35	55		DB	\$55
5636:	1F36	00		DB	\$00
5636:	1F37			ENDM	
5636:					
5637:	1F37			DEF03	45, 55, 00
5637:	1F37	45		DB	\$45
5637:	1F38	55		DB	\$55
5637:	1F39	00		DB	\$00
5637:	1F3A			ENDM	
5637:					
5638:	1F3A	01		DB	01H
5639:	1F3B	55		DB	55H
5640:	1F3C		WAGPAT	DEF04	0, 0, 4, 16
5640:	1F3C	00		DB	\$0
5640:	1F3D	00		DB	\$0
5640:	1F3E	04		DB	\$4
5640:	1F3F	16		DB	\$16
5640:	1F40			ENDM	
5640:					
5641:	1F40			DEF04	00, 05, 50, 00
5641:	1F40	00		DB	\$00
5641:	1F41	05		DB	\$05
5641:	1F42	50		DB	\$50
5641:	1F43	00		DB	\$00
5641:	1F44			ENDM	
5641:					
5642:	1F44			DEF04	00, 55, 55, 00
5642:	1F44	00		DB	\$00
5642:	1F45	55		DB	\$55
5642:	1F46	55		DB	\$55
5642:	1F47	00		DB	\$00
5642:	1F48			ENDM	
5642:					
5643:	1F48			DEF04	01, 55, 55, 40
5643:	1F48	01		DB	\$01
5643:	1F49	55		DB	\$55
5643:	1F4A	55		DB	\$55
5643:	1F4B	40		DB	\$40
5643:	1F4C			ENDM	
5643:					
5644:	1F4C			DEF04	05, 55, 55, 50
5644:	1F4C	05		DB	\$05
5644:	1F4D	55		DB	\$55
5644:	1F4E	55		DB	\$55
5644:	1F4F	50		DB	\$50
5644:	1F50			ENDM	
5644:					
5645:	1F50			DEF04	15, 54, 15, 54
5645:	1F50	15		DB	\$15
5645:	1F51	54		DB	\$54
5645:	1F52	15		DB	\$15
5645:	1F53	54		DB	\$54
5645:	1F54			ENDM	
5645:					
5646:	1F54			DEF04	15, 50, 05, 54
5646:	1F54	15		DB	\$15
5646:	1F55	50		DB	\$50

Home Video Game System Library - Gunfight

5646:	1F56	05	DB	\$05
5646:	1F57	54	DB	\$54
5646:	1F58		ENDM	
5646:				
5647:	1F58		DEF04	15, 40, 01, 54
5647:	1F58	15	DB	\$15
5647:	1F59	40	DB	\$40
5647:	1F5A	01	DB	\$01
5647:	1F5B	54	DB	\$54
5647:	1F5C		ENDM	
5647:				
5648:	1F5C		DEF04	15, 40, 01, 54
5648:	1F5C	15	DB	\$15
5648:	1F5D	40	DB	\$40
5648:	1F5E	01	DB	\$01
5648:	1F5F	54	DB	\$54
5648:	1F60		ENDM	
5648:				
5649:	1F60		DEF04	15, 50, 05, 54
5649:	1F60	15	DB	\$15
5649:	1F61	50	DB	\$50
5649:	1F62	05	DB	\$05
5649:	1F63	54	DB	\$54
5649:	1F64		ENDM	
5649:				
5650:	1F64		DEF04	05, 54, 15, 50
5650:	1F64	05	DB	\$05
5650:	1F65	54	DB	\$54
5650:	1F66	15	DB	\$15
5650:	1F67	50	DB	\$50
5650:	1F68		ENDM	
5650:				
5651:	1F68		DEF04	01, 55, 55, 40
5651:	1F68	01	DB	\$01
5651:	1F69	55	DB	\$55
5651:	1F6A	55	DB	\$55
5651:	1F6B	40	DB	\$40
5651:	1F6C		ENDM	
5651:				
5652:	1F6C		DEF04	00, 55, 55, 00
5652:	1F6C	00	DB	\$00
5652:	1F6D	55	DB	\$55
5652:	1F6E	55	DB	\$55
5652:	1F6F	00	DB	\$00
5652:	1F70		ENDM	
5652:				
5653:	1F70		DEF04	00, 15, 54, 00
5653:	1F70	00	DB	\$00
5653:	1F71	15	DB	\$15
5653:	1F72	54	DB	\$54
5653:	1F73	00	DB	\$00
5653:	1F74		ENDM	
5653:				
5654:	1F74		DEF04	02, AA, AA, 80
5654:	1F74	02	DB	\$02
5654:	1F75	AA	DB	\$AA
5654:	1F76	AA	DB	\$AA

Home Video Game System Library - Gunfight

5654:	1F77	80	DB	\$80
5654:	1F78		ENDM	
5654:				
5655:	1F78		DEF04	00, AA, AA, 00
5655:	1F78	00	DB	\$00
5655:	1F79	AA	DB	\$AA
5655:	1F7A	AA	DB	\$AA
5655:	1F7B	00	DB	\$00
5655:	1F7C		ENDM	
5655:				
5656:	1F7C		DEF04	12, AA, AA, 84
5656:	1F7C	12	DB	\$12
5656:	1F7D	AA	DB	\$AA
5656:	1F7E	AA	DB	\$AA
5656:	1F7F	84	DB	\$84
5656:	1F80		ENDM	
5656:				
5657:	1F80		DEF04	10, A8, 2A, 04
5657:	1F80	10	DB	\$10
5657:	1F81	A8	DB	\$A8
5657:	1F82	2A	DB	\$2A
5657:	1F83	04	DB	\$04
5657:	1F84		ENDM	
5657:				
5658:	1F84		DEF04	10, 20, 08, 04
5658:	1F84	10	DB	\$10
5658:	1F85	20	DB	\$20
5658:	1F86	08	DB	\$08
5658:	1F87	04	DB	\$04
5658:	1F88		ENDM	
5658:				
5659:	1F88		DEF04	52, AA, AA, 85
5659:	1F88	52	DB	\$52
5659:	1F89	AA	DB	\$AA
5659:	1F8A	AA	DB	\$AA
5659:	1F8B	85	DB	\$85
5659:	1F8C		ENDM	
5659:				
5660:	1F8C		DEF04	10, 20, 08, 04
5660:	1F8C	10	DB	\$10
5660:	1F8D	20	DB	\$20
5660:	1F8E	08	DB	\$08
5660:	1F8F	04	DB	\$04
5660:	1F90		ENDM	
5660:				
5661:	1F90		DEF04	10, 00, 00, 04
5661:	1F90	10	DB	\$10
5661:	1F91	00	DB	\$00
5661:	1F92	00	DB	\$00
5661:	1F93	04	DB	\$04
5661:	1F94		ENDM	
5661:				
5662:	1F94		DEF04	10, 00, 00, 04
5662:	1F94	10	DB	\$10
5662:	1F95	00	DB	\$00
5662:	1F96	00	DB	\$00
5662:	1F97	04	DB	\$04

Home Video Game System Library - Gunfight

```

5662: 1F98          ENDM
5662:
5663: 1F98 00          FUDG4  DB    0
5664: 1F99          MSET  MASTER 11H
5664: 1F99 80          DB    $80
5664: 1F9A 11          DB    11H
5664: 1F9B          ENDM
5664:
5665: 1F9B          VOLUME 09H, 0H
5665: 1F9B B0          DB    $B0
5665: 1F9C 09          DB    09H
5665: 1F9D 00          DB    0H
5665: 1F9E          ENDM
5665:
5666: 1F9E C9          RET
5667:          ; HOME ON THE RANGE
5668: 1F9F CD991F        HOME  CALL  MSET
5669: 1FA2          NOTE1 36, G1
5669: 1FA2 24          DB    (36)&($7F)
5669: 1FA3 7E          DB    G1
5669: 1FA4          ENDM
5669:
5670: 1FA4          NOTE1 12, F1
5670: 1FA4 0C          DB    (12)&($7F)
5670: 1FA5 8D          DB    F1
5670: 1FA6          ENDM
5670:
5671: 1FA6          NOTE1 18, E1
5671: 1FA6 12          DB    (18)&($7F)
5671: 1FA7 96          DB    E1
5671: 1FA8          ENDM
5671:
5672: 1FA8          NOTE1 6, D1
5672: 1FA8 06          DB    (6)&($7F)
5672: 1FA9 A8          DB    D1
5672: 1FAA          ENDM
5672:
5673: 1FAA          NOTE1 36, E1
5673: 1FAA 24          DB    (36)&($7F)
5673: 1FAB 96          DB    E1
5673: 1FAC          ENDM
5673:
5674: 1FAC          QUIET
5674: 1FAC F0          DB    $F0
5674: 1FAD          ENDM
5674:
5675:          ; TAPS
5676: 1FAD          TAPS
5677: 1FAD CD991F        CALL  MSET
5678: 1FB0          NOTE1 18, C1
5678: 1FB0 12          DB    (18)&($7F)
5678: 1FB1 BD          DB    C1
5678: 1FB2          ENDM
5678:
5679: 1FB2          NOTE1 06, C1
5679: 1FB2 06          DB    (06)&($7F)
5679: 1FB3 BD          DB    C1

```

Home Video Game System Library - Gunfight

```

5679: 1FB4          ENDM
5679:
5680: 1FB4          NOTE1 36, F1
5680: 1FB4 24        DB      (36)&($7F)
5680: 1FB5 8D        DB      F1
5680: 1FB6          ENDM
5680:
5681: 1FB6          NOTE1 18, C1
5681: 1FB6 12        DB      (18)&($7F)
5681: 1FB7 BD        DB      C1
5681: 1FB8          ENDM
5681:
5682: 1FB8          NOTE1 6, F1
5682: 1FB8 06        DB      (6)&($7F)
5682: 1FB9 8D        DB      F1
5682: 1FBA          ENDM
5682:
5683: 1FBA          NOTE1 36, A1
5683: 1FBA 24        DB      (36)&($7F)
5683: 1FBB 70        DB      A1
5683: 1FBC          ENDM
5683:
5684: 1FBC          QUIET
5684: 1FBC F0        DB      $F0
5684: 1FBD          ENDM
5684:
5685:                ; FUNERAL
5686: 1FBD CD991F    FUNERL CALL MSET
5687: 1FC0          NOTE1 24, A0
5687: 1FC0 18        DB      (24)&($7F)
5687: 1FC1 E1        DB      A0
5687: 1FC2          ENDM
5687:
5688: 1FC2          NOTE1 18, A0
5688: 1FC2 12        DB      (18)&($7F)
5688: 1FC3 E1        DB      A0
5688: 1FC4          ENDM
5688:
5689: 1FC4          NOTE1 6, A0
5689: 1FC4 06        DB      (6)&($7F)
5689: 1FC5 E1        DB      A0
5689: 1FC6          ENDM
5689:
5690: 1FC6          NOTE1 24, A0
5690: 1FC6 18        DB      (24)&($7F)
5690: 1FC7 E1        DB      A0
5690: 1FC8          ENDM
5690:
5691: 1FC8          NOTE1 18, C1
5691: 1FC8 12        DB      (18)&($7F)
5691: 1FC9 BD        DB      C1
5691: 1FCA          ENDM
5691:
5692: 1FCA          NOTE1 6, B0
5692: 1FCA 06        DB      (6)&($7F)
5692: 1FCB C8        DB      B0
5692: 1FCC          ENDM

```

Home Video Game System Library - Gunfight

```

5692:
5693: 1FCC          NOTE1 18, B0
5693: 1FCC 12       DB      (18)&($7F)
5693: 1FCD C8       DB      B0
5693: 1FCE          ENDM
5693:
5694: 1FCE          NOTE1 6, A0
5694: 1FCE 06       DB      (6)&($7F)
5694: 1FCF E1       DB      A0
5694: 1FD0          ENDM
5694:
5695: 1FD0          NOTE1 18, A0
5695: 1FD0 12       DB      (18)&($7F)
5695: 1FD1 E1       DB      A0
5695: 1FD2          ENDM
5695:
5696: 1FD2          NOTE1 6, GS0
5696: 1FD2 06       DB      (6)&($7F)
5696: 1FD3 EE       DB      GS0
5696: 1FD4          ENDM
5696:
5697: 1FD4          NOTE1 18, A0
5697: 1FD4 12       DB      (18)&($7F)
5697: 1FD5 E1       DB      A0
5697: 1FD6          ENDM
5697:
5698: 1FD6          QUIET
5698: 1FD6 F0       DB      $F0
5698: 1FD7          ENDM
5698:
5699: 1FD7          GUNSHOT OUTPUT 18H, 0F0H, 0F5H, 0FDH, 0FFH, 0, 3FH, 0FFH,
0EFH
5699:          ;          IF      NOT (18H=$18)
5699: 0000          IF      18H!=$18
5699:          ENDIF
5699: 0001          IF      18H=$18
5699: 1FD7 88       DB      $88
5699: 1FD8          DEF8  0EFH, 0FFH ,3FH, 0, 0FFH, 0FDH, 0F5H, 0F0H
5699: 1FD8 EF       DB      0EFH
5699: 1FD9 FF       DB      0FFH
5699: 1FDA 3F       DB      3FH
5699: 1FDB 00       DB      0
5699: 1FDC FF       DB      0FFH ; 'GE' reserved, so used '0FFH'
5699: 1FDD FD       DB      0FDH
5699: 1FDE F5       DB      0F5H
5699: 1FDF F0       DB      0F0H
5699: 1FE0          ENDM
5699:
5699:          ENDIF
5699: 1FE0          ENDM
5699:
5700: 1FE0          LEGSTA
5700: 1FE0 E0       DB      $E0
5700: 1FE1          ENDM
5700:
5701: 1FE1          VOLUME 0FFH, 03FH
5701: 1FE1 B0       DB      $B0

```

Home Video Game System Library - Gunfight

```

5701: 1FE2 FF          DB      0FFH
5701: 1FE3 3F          DB      03FH
5701: 1FE4             ENDM
5701:
5702: 1FE4             REST    5
5702: 1FE4 E1          DB      $E1
5702: 1FE5 05          DB      5
5702: 1FE6             ENDM
5702:
5703: 1FE6             NOTE1   5, 8FH
5703: 1FE6 05          DB      (5)&($7F)
5703: 1FE7 8F          DB      8FH
5703: 1FE8             ENDM
5703:
5704: 1FE8             NOTE1   5, 4CH
5704: 1FE8 05          DB      (5)&($7F)
5704: 1FE9 4C          DB      4CH
5704: 1FEA             ENDM
5704:
5705: 1FEA             QUIET
5705: 1FEA F0          DB      $F0
5705: 1FEB             ENDM
5705:
5706: 1FEB             LASTB  EQU  $
5707: ;
5708: ; *****
5709: ; * RAM CELLS *
5710: ; *****
5711: ;          ORG  NORMEM+0E70H
5712: ;          DEFS 150             ; ALLOW BIG STACK
5713: 4F06          STACK  EQU  $4F06             ; START STACK HERE
5714: ;          DEFS 12
5715: 4F12          MSTACK EQU  $4F12
5716: 4F12          STRRAM EQU  $4F12
5717: 4F12          WRITQ  EQU  $4F12             ; WRUTE Q HEADER
5718: 4F15          VECQ   EQU  $4F15             ; VECTOR Q HEADER
5719: 4F18          VECSTR EQU  $4F18
5720: 4F18          BULV1  EQU  $4F18             ; BULLET VECTOR 1
5721: 4F2A          BULV2  EQU  $4F2A             ; BULLET VECTOR 2
5722: 4F3C          BULV3  EQU  $4F3C             ; BULLET VECTOR 3
5723: 4F4E          BULV4  EQU  $4F4E             ; BULLET VECTOR 4
5724: ;          DEFS 1             ; LEFT COMWBOY LINK
5725: 4F61          LCOWB  EQU  $4F61             ; LEFT GUNFIGHTER
5726: ;          DEFS 1             ; RIGHT COWBOY LINK
5727: 4F78          RCOWB  EQU  $4F78             ; RIGHT GUNFIGHTER
5728: 4F8F          WAGVEC EQU  $4F8F             ; WAGON VECTOR
5729: 4F90          WAGON  EQU  $4F90             ; WAGVEC+VBSTAT
5730: 4FA1          ENDRAM_ EQU  $4FA1
5731: 4FDA          LBULS  EQU  $4FDA             ; CT5
5732: 4FDB          RBULS  EQU  $4FDB             ; CT6
5733: 4FA1          RFIELD EQU  $4FA1
5734: 4FA2          LSCORE EQU  $4FA2
5735: 4FA5          LFIELD EQU  $4FA5
5736: 4FA6          RSCORE EQU  $4FA6
5737: 1FEF          LEND   EQU  $1FEF             ; LASTB
5738: ;
5739: ;          END OF GUNFIGHT PROGRAM

```

Home Video Game System Library - Gunfight

```
5740:          ;
5741:          ; TWENTY-ONE FILLER BYTES BETWEEN GUNFIGHT AND
5742:          ; END OF ROM TO MATCH 3159 ROM
5743: 1FEB  FFFFFFFF          DB  0FFH, 0FFH, 0FFH, 0FFH, 0FFH
      FF
5744: 1FF0  FFFFFFFF          DB  0FFH, 0FFH, 0FFH, 0FFH, 0FFH
      FF
5745: 1FF5  FFFFFFFF          DB  0FFH, 0FFH, 0FFH, 0FFH, 0FFH
      FF
5746: 1FFA  FFFFFFFF          DB  0FFH, 0FFH, 0FFH, 0FFH, 0FFH, 0FFH
      FFFF
5747:
**** ballyrom.asm ****
```

Home Video Game System Library - Symbol Table

Statistics:

```

1243      symbols
8192      bytes

505       macro calls
3263     macro bytes
0         invented symbols
    
```

Symbol Table:

```

a0          = e1      gs3          = 1d+    pncm          dd3
a1          = 70      gs4          = e+      pngf          dca
a2          = 37+    gsbend       = 7       pnote0       1744+
a3          = 1b+    gsbscr       = 1       pnotel       174e+
a4          = d+     gsbtim       = 0       pnote2       1758+
a5          = 6+     gt01         = 4f4     pnote3       1762+
actbit      = 7       gt02         = 4f9     pnsch        de8
actint      = e       gtimer       = 4e0     pointp       =4edc+
action      146b    gtmins       =4fee     pot0          = 1c
active      = 80      gtplix       = 13c4    pot1          = 1d+
actnum      118a    gtsecs       =4fed     pot2          = 1e+
addtb       130f    gunlmt       = 1d83    pot3          = 1f+
addtbl      1313    gunlnk       = 218     ppacks       1328+
adtd        1311+   gunshot      = 1fd7    ppadr0       1748+
addtq       1d50    gvec3a       = 1cd9    ppadr1       1752+
addtq1      1d64    gvec3b       = 1ce8    ppadr2       175c+
adig1       1lab    gvect        = 1c75    ppadr3       1766+
adig2       11c1    gvect1       = 1cb1    ppat0        1799
adig3       11cd    gvect2       = 1cc0    ppat1        179d
adig4       11d5    gvect3       = 1cd6    ppat2        17a1
adig5       11d6    gvect4       = 1cf8    ppat3        17a5
adig6       11dd    gvect5       = 1d03    ppath        = 5
adig7       11de    gvect6       = 1d0e    ppat1        = 4
adig8       11ed    handle       = 453     ppot         18be
adig9       11fd    height       = 4       ppot0        18b9
adown       1791    hit          = 18e6    ppot1        18b1
akeys       214    hit1         = 18fb    prior         =4ff9
aleft       1795    hit2         = 1920    psdop        = 9
aline       = 9     hitchk       = 18ce    psdop0       174d+
alkeys      = 214    home         = 1f9f    psdop1       1757+
allbyt      = d20    horaf        = f+      psdop2       1761+
allhum      13b7    horcb        = 9       psdop3       176b+
amove       = 8     human        = 40     pspos0       174b+
aniarr      15ff    humanr       = 40     pspos1       1755+
animax      = 3+    humbit       = 6       pspos2       175f+
anymov      16a0    humchk       = 163b+   pspos3       1769+
arg1        =4ec4    incix        = 146e    psposx       = 7
arg1md      = 0     inclop       = c18     psposy       = 8
arg2        =4ecf    incscr       = 54     pstat        = 6
arg2md      = 1     indexb       = 5c     pswcy        = 0+
aright      178d    indexn       = 56     pswpv        = 2+
arm0        1df8    indexw       = 5a     pswsgn       = 7+
arm1        1e06    infbk        = d       pswzro       = 6
arm2        1e10    inicol       = 100c    pushl        5d
    
```

Home Video Game System Library - Symbol Table

arm3	1e18	init	19e8	pushel	122d
arm4	1e24	initq	1a5e+	pushen	1216
arm5	1e32	inlin	= f	pushn	93c+
arm6	1e42	inmod	= e	putnb1	ba5
armtbl	1dd7	intipp	13ba+	putnb2	ba8
arot	= 3	intpc	= 0	putnib	b90
arrx	= 4	intpcc	= 0+	putvec	19d3
arry	= 5	intpe	4e	pvolab	=4fd2
as0	= d4	intst	= 8+	pvolmc	=4fd3
as1	= 6a	inttbl	1d74	pwrap	c61
as2	= 34+	inxnib	b76	pwrap1	c95
as3	= 1a+	itab	34	qfrog	ad1
aup	1789	iter4	eaa	quiet	a6c
b0	= c8	iter41	eb3	quit	= 78
b1	= 64+	itret	eba	r1	3a2
b2	= 31+	jmpd	54f	r2	3a6
b3	= 18+	joy0	188f	r3	3a9
badmov	1701	joy1	1895	ramtbl	1676
bcacy	= 46	joys	471	ranfin	1697+
bcdad	321	jusjoy	= f+	ranged	= 76
bcdadd	= 62	kart	18ca	ranmov	167e
bcdchs	= 6a	kctasc	= 40	ransht	=4fef
bcdcs	364	kctatb	ad5	rantst	14cf
bcdd0	bee	key0	= 14+	rbuls	=4fdb
bcdd1	bfd	key1	= 15+	rbulx	= 68
bcdd2	c05	key2	= 16+	rc	4e0
bcdd3	c0b	key3	= 17	rcacx	= 58
bcdd4	c11	keyb0	e69	rcall	= 4
bcddiv	= 68	keyb1	e8c	rcowb	=4f78
bcddv	284	keyget	1043	rectan	= 1c
bcdisp	beb	keygt1	1095	relabl	= 3a
bcdml	2de	keymes	e65	relabs	= 38
bcdmul	= 66	keysex	=4fe3	reld	68
bcdneg	= 6c	kill	1e80	relta	b08
bcdng	341	kil2	1ed2	reltal	b4e
bcdng1	34d	killst	159c	relta2	b56+
bcdsb	31f	kiltr1	1309	relta3	b5f
bcdsub	= 64	kiltrz	1303	reenter	7c
begin	e19+	klrksx	e9c	repeat	c3b
begint	1b5d+	ksctrv	= 14	rest	a38
begram	=4fce	larg2	= c	restom	158d
belp	1859	large	= b	restor	= 2e
berase	183a	lastb	=1feb+	retn_	27a
bitspl	= a0+	lastmv	= 1	rfield	=4fal
blank	= 2a	lastsw	= 0	rftab	1dbe
bline	= 5c	lbuls	=4fda	ritb	184f
bmusic	= 12	lbulx	= 20	rlmove	= c
bn	b9c	lcacx	= 40	rmask	=4fa7+
borg	1aad	lcowb	=4f61	rnx	= 88
botlin	= 0	ldplix	165b	romtbl	166e
bsy	= 2	ldpliy	165b	rotmsk	16ae
btreey	= 41+	leg0	1e4b	rscore	=4fa6
bullmt	1d8b	leg1	1e60	rstart	=4fal+
bullp	1ab9	leg2	1e70	save	= 2c
bulrit	1b53	legsta	a16	sblen	= 8
bult	= b	lend	=1fef+	scbl	dc4
bultab	1d8f	lfield	=4fa5	scbst	= e19

Home Video Game System Library - Symbol Table

bulv1	=4f18	lfrlin	= c8	scclr	e20
bulv2	=4f2a+	lfrvec	1d74	scdotb	ea1
bulv3	=4f3c	lftab	1db9	schedr	= c
bulv4	=4f4e+	lnx	= 8	scpsiz	= 1e
bulvsz	= 12	looksq	1707	scrib3	fef
bumpck	15c5	loop	1b03	screen	= 0
bumpem	15a2	loopy	144e	screst	ee5
bytepl	= 28	lowx	= 0	scrld0	1266
c1	= bd	lowy	= b	scrld1	1276
c2	= 5e+	lppp2	1b15	scrld2	1280
c3	= 2e+	lrgchr	8e4	scrldn	125e
c4	= 17+	lscore	=4fa2	scrky	1238
c5	= b+	m06	5da	scrul1	124b
c6	= 5+	m061	5e6	scrul2	124d
c7	= 2+	m81	53a	scroll	= 30
cactus	1ef4	m815	540	scrptr	=4ee5
cacw	19c8	m82	547	scrsav	f2d
calcl	1fa	m83	54b	scrstk	=4ee8
calcln	= 5c	mactin	18b	scrstr	= 16
calclr	131c	magic	= c	scrup1	eda
calcst	=1020	mainlp	e58	scrupd	ec4
calpiz	148a	master	7ec	scrwr1	eff
calram	=4ec0	math	= 56	scrwr2	f1a
calstk	=4ec0	maxstk	= 5c	scrwrt	ef6
cba	= 9	mblan1	7a3	scsadh	= 1d
cbb	= 7	mblan2	7a4	scsadh	= 1c
cbc	= 6	mblank	79e	scsava	= 0+
cbd	= 5	mc	471	sct0	= 1
cbe	= 4	mcac	19a6	sct1	= 2
cbflag	= 8	mcacy	= 2a	sct2	= 3+
cbh	= b	mcall	= 6	sct3	= 4+
cbixh	= 3	mcolor	1db	sct4	= 5+
cbixl	= 2	mdisti	bcc	sct5	= 6+
cbiyh	= 1+	mdola	61c+	sct6	= 7+
cbiyl	= 0+	mdoit	60c	sct7	= 8
cbl	= a	mdoit0	60e	scxc	= 1a
cblen	= 8+	mdoit1	616	scyc	= 1b
cblock	17a9	mdoit2	620	sdabs	356
ccacx	= 4c	mdoit3	621+	sdadd	36e
cctl	3e6	mdoitb	60b	sdadd1	36f
cctlp	3dd	memdig	=4ec3+	sdsmsg	329
ccvt1	12e1	memfun	10d8	sdsmsg1	333
ccvt1a	12f5	memsta	=4ec2	semi4s	=4fde
ccvt2	12f6	mentry	1ac	senflg	=4ffa
ccvt3	1300	menu	= 4a	sentry	= 42
cdcolr	1785+	menucl	13	setb	= 7a
cdeltb	f29	menust	= 218+	setend	c35
cdopt	= 44	mfill	aee	setout	= 16
cdown1	1426	mfill1	aef	setw	= 7c
chdown	= 1	mfrog	b06	sf0	= 9
chgsgn	10de	mgetn	d31	sf1	= a+
chkmov	16bf	mgetn1	d36	sf2	= b+
chleft	= 2	mgetn2	d55	sf3	= c+
chrdis	= 32	mgetn3	d61	sf4	= d+
chrigh	= 3	mgetn4	d6f	sf5	= e+
chtrig	= 4+	mgetn5	d75	sf6	= f+
chup	= 0	mgetn6	d7f	sf7	= 10+

Home Video Game System Library - Symbol Table

cknopl	141e	mgetn7	d86	sh1	3b1
cksum	33+	mgetn8	d93	shftit	1693
cksum2	b0e+	mgetn9	d96	shftul	db1
cksum3	1418+	mgetp	cfb	shiftr	3ac
clear	1020	midc	1aa2	shiftn	= 60
clearf	14b3	mincsc	c15	shonum	1084
clrent	11af	mindb	bbd	showbf	=4edd
clrnum	d99	mindbl	bc5	showck	120a
cmem0	=4ee6	mindw	bac	showmd	= 2
cml	dbe	mint0	84	sinit	1dcb
cmpit	c22	mint1	95	sixy	4cc
cmplop	c30	mint2	9a	siztbl	f26
cmstrt	=1328	mintpc	7b	sj0	= 15
cnohum	=4fa4	mjump	= a	sj1	= 17
cnopl	=4fa2	mkctas	ac9	sj2	= 19
cnt	=4fdd	mmcall	7d	sj3	= 1b
col0l	= 4	mmenu	c97	skyd	= 13
col0r	= 0+	mmenu1	ca2	skyu	= 12
col1l	= 5+	mmenu3	cc8	smlchr	abf
collr	= 1+	mmenu5	cd9	smlfnt	20d
col2l	= 6+	mmenu6	cdf	sndbx	= 18
col2r	= 2+	mmenu7	ce2	snegt	34c
col3l	= 7+	mmjump	ac4	snul	= 0+
col3r	= 3+	mmove	b4b	sp0	= 1c
colbx	= b	mmret	b73	sp1	= 1d
collst	=4fe8	mmtd	240	sp2	= 1e+
colmsk	f22	mmtd1	24e	sp3	= 1f+
colors	=4ee8	mmtd2	24f	ssec	= 11
colset	= 18	mnclr	d19	st0	= 14
concl	264	mngl	= 5+	st1	= 16
conc2	2b	mngl	= 4+	st2	= 18+
concm	= 8	mnh	= 1	st3	= 1a+
concp1	256	mnl	= 0	stack	=4f06
convrt	12c7	mnsah	= 3	stak0	4be
cowint	1d30	mnsal	= 2	stall	161d
cowx	= 60+	mo0	55b	starts	=41b8
crash	1532	mo01	56b	statev	=4ec1
crel	9b4+	mo1	574	sthldc	bb8
cs1	= b2	mo2	57d	stick	1657
cs2	= 59+	mo3	587	stimer	= 200
cs3	= 2c+	mo4	594	stkbot	=4e58
cs4	= 15+	mo40	5a1	stloop	1584
cs5	= a+	mo41	5a5	stmrx	= 4c
ct0	=4fd5	mo43	5b7	stomp	157b
ct1	=4fd6	mo44	5c0	storen	= 58
ct2	=4fd7+	mo45	5c5	storit	1627
ct3	=4fd8+	mo5	5cc	strd1	7ce
ct4	=4fd9+	movany	16a9	strd2	7d4
ct5	=4fda	move	= 5e	strdis	= 34
ct6	=4fdb+	moveit	14bb	stripe	6e2
ct7	=4fdc	movend	172a	strnd	1a0c
ctimer	= 203+	movext	16bd	strnew	7c4
ctlp	3d9	movjoy	1483	strpl	6eb
curscr	=4f96	movtmr	= 4	strram	=4f12
cursw	= 2	movtst	16ab	stsec	1837
d1	= a8	mpaint	6b2	suck	= c
d2	= 54+	mpause	1b	sw0	= 10

Home Video Game System Library - Symbol Table

d3	= 29+	mpizbk	1ba	sw1	= 11+
d4	= 14+	mpt1	6c5	sw2	= 12+
dabs	= 72	mpt2	6cf	sw3	= 13+
dadd	= 6e	mpt3	6d5	swhit	461
dclct1	849	mpt4	6de	swlop	456
dclctb	83e	mquit	c41	sysdpt	cb
dclock	17e1	mquit1	c4a	sysfnt	206
dcout	17f7	mquit2	c56	sysram	=9fcd+
death	1b0c+	mrangle	37f	syssuk	32f
deccts	= 10	mrargt	14b	system	2c2
def02	bf0	mrcall	632	taps	1fad
def03	c23	mrela1	afb	tarrx	=4fa5
def04	c68	mrela2	b00	tarry	=4fa6
def2	0	mrelab	af6	tbump	1d1a
def3	33+	mrest	7ad	tbump1	1d21
def4	97	mrest1	7b5	tcac	199b
def4x	b0d+	mret	= 8	tcacy	= 14
def5	ee	mrflap	= 6	tdopt	= 24
def6	157+	mrlock	=4ff7+	tdown	16f3
def8	1d2	mrrot	= 4	tdpack	1787+
defchr	a8e	mrrot	= 2+	textd	5ba
deload	74	mrshft	= 3+	thetbl	1458
delq	1d25	mrxor	= 5	tickit	1649
die	1930	mrxpnd	= 3	time	= b
die1	194c	msave	3b9	timex	47b
die4	1963	msavel	3c2	timey	47e
difer	=4fff+	mscrll	26b	timez	4a0
digits	=4eda	mscrol	26a	timlp	485
dirtmr	=4ef2	msenk2	43b	timeout	=4fec
dirval	=4ef5	msenke	446	tiyu	labe
discla	7f1	mset	1f99	tkeys	421
disclb	7fe	msetb	36c	tleft	16c9+
dischl	7ed	msetup	3cf	tline	= a
disch2	80a	msetw	23	tmr60	=4feb+
disch3	80d	mshftu	daa	tokejt	10b8
disch4	821	mskl	42c	toketb	10a0
disch5	839	msktd	= 7e	tonea	= 11+
disnum	= 36	mstack	=4f12	toneb	= 12+
dispch	7e1	msuck	a4	tonec	= 13
dispsc	15e2	msuck1	a8	tonmo	= 10
distim	= 52	msuck2	b6	toplin	= 6a
div1	29f	msuck3	bf	tplop	3ff
div2	2a3	msuck5	c6	trchk	3ec
div3	2b1	mult1	2cd	tree	1de5
div4	315	mult2	2e1	tright	16d7
dleft	1942	mult3	2e8	tsex	413
do	42d	mult4	2f0	tstbit	149b
doadd	1162	mult5	309	ttest	1e5
doclre	1186	mult6	31b	ttreey	= f+
dodiv	116b	mult7	313	tup	16e5
doit	= 44	musvol	= 9	udmove	= 3
doitb	= 46	muz999	5f4	umargt	=4ffb
domul	1168	muzak	= 12	uncram	=4f96
dont	450	muzcpl	517	upistr	= 0
dontd	13a1+	muzcpu	514	upmuzk	1490
doop	1146	muzpc	=4fce	usertb	=4ffd
dooptb	115a	muzset	508	vbarm	= f

Home Video Game System Library - Symbol Table

dosub	1165	muzsp	=4fd0	vbblnk	= 6
dowrts	fd3	muzstp	5fc	vbcchk	= 4
dpoint	=4edb+	mvblal	79a	vbch	= 3
draw	1d87	mvblan	77d	vbcl	= 2
drx	= 40	mvct1a	66f	vbclat	= 3
ds1	= 9f	mvect	633	vbclmt	= 0
ds2	= 4f+	mvect1	665	vbcomp	= 13+
ds3	= 27+	mvect2	684	vbcrev	= 1
ds4	= 13+	mvect3	6a4	vbdch	= 1
ds5	= 9+	mvect6	6a6	vbdcl	= 0
ds6	= 4+	mvectc	656	vbdxh	= 4
dsjnz	9e2+	mvwrit	6fe	vbdxl	= 3
dsmg	= 70	mwrit	719	vbdyh	= 9
dtab	1b34	mwrita	71c	vbdyl	= 8
durat	=4fea	mwritp	715	vblank	= 28
e1	= 96	mwritr	70b	vbleg	= 12
e2	= 4a+	mwrt	725	vblegt	= 11
e3	= 25+	mwrtfl	74c	vbm	= 0
e4	= 12+	mwrx	735	vboah	= e
eire	1bba+	mwrx1	736	vboal	= d
elop	1917	mwrx2	739	vboarm	= 10
emusic	= 14	mwxf	766	vbsact	= 7
endchk	15d5	mwxf1	767	vbschg	= 3
endcr	=4f54	mwxf2	76a	vbsint	= 5
endgam	1b2c	mxintc	279	vbsnom	= 4
endram	=4fc4	mxscr	= 21e	vbstat	= 1
endram_	=4fal	nbrk	188d	vbswag	= 0
endrnd	1b28	negt	= 74	vbtimb	= 2
endscr	=4ff4	newmov	168c	vbxchk	= 7
endx	= c0	newway	= 1+	vbxh	= 6
entstg	db7	next	=ffff	vbxl	= 5+
eplop	410	nlines	=4ee4	vbychk	= c
eqrout	1111	nobit	= 2+	vbyh	= b
eqrtl	1120	nocur	14ca	vbyl	= a+
erase	1524	nogame	= 235	vecq	=4f15
erase_	190a	noline	= 60	vecstr	=4f18+
etlp	493	noplay	= 228	vect	= 3e
excols	1780	normem	=4000	vectc	= 3c
exit	3f8	notel	629	veraf	= e+
exloop	153d	note2	664+	verbl	= a+
expat1	176c+	note3	6b1+	vibra	= 14+
expat2	1770+	note4	708+	voicem	908+
expat3	1774+	note5	771+	voices	=4fd4
expat4	1778+	nothum	1419	volab	= 16
expat5	177c+	notxor	15eb	volc	= 15
expats	176c	npbit	= 3+	voln	= 17+
expfin	156c	nulkey	12a9+	volume	96e
expsnd	17b1	nulpat	1f08	vwritr	= 1e
f1	= 8d	numb	= 7+	waglmt	1d78
f2	= 46+	numba	128a	wagon	=4f90
f3	= 22+	numbal	129c	wagpat	1f3c
f4	= 11+	numbas	b	wagvec	=4f8f
f5	= 8+	numchk	116d+	wagvsz	= 12+
field	1988	numply	=4ff3	wagx	= 48
fill	= 1a	nwhdwr	= 1	walk	1ad5+
fillq	39d	nxtfr1	858	waste	= fff
findl3	cf4	nxtfr2	863	waster	= fff+

Home Video Game System Library - Symbol Table

fire0	17ff	nxtfr3	86a	width	= 4
fire1	180a	nxtfrm	84e	winbnd	= 32
fireit	1340	oa1	= 8f+	wponof	= 0+
first	1d67	oa2	= 47+	wpopt	= 1+
firstc	=2000	oa3	= 23+	wppah	= 3+
fntsm1	= 20d	oa4	= 11	wppal	= 2+
fntsys	= 206+	oa5	= 8+	wpxsiz	= 5+
forcem	= f6+	ob0	= fe+	wpysiz	= 4+
fplay	13b5	oc0	= f1+	wrb15a	1c4e
fs1	= 85	od1	= d6+	wrbul1	1be5
fs2	= 42+	oe1	= bf+	wrbul2	1bfc
fs3	= 20+	of1	= b4+	wrbul3	1c29
fs4	= 10+	ogl	= a0+	wrbul4	1c2d
ftbase	= 0	oldway	= 0	wrbul5	1c4b
ftbyte	= 3	onetim	1328+	wrbul6	1c5a
ftfsx	= 1	opl	10f0	wrbul7	1c6c
ftfsy	= 2	op2	1106	wreck	b60+
ftpth	= 6	op2a	1109	wrfl1	751
ftptl	= 5	op3	110b	wrfl2	754
ftsiz	= 4	opcod	=4ec0	writ	= 24
fudg4	1f98+	oploop	51b	writa	= 26
funer1	1fbd	oplp2	592	writor	= 10
g0	= fd	opot0	=4fdf+	writp	= 22
g1	= 7e	opot1	=4fe0+	writq	=4f12
g2	= 3e+	opot2	=4fe1+	writr	= 20
g3	= 1f+	opot3	=4fe2+	wrtl1	88d
g4	= f+	oprout	10e8	wrtl2	898
g5	= 7+	optbl	12c1	wrtl3	8ac
g6	= 3+	osw0	=4fe4	wrtl4	8bf
g7	= 1+	osw1	=4fe5+	wrtl5	8c4
g8	= 0+	osw2	=4fe6+	wrtl6	8d4
gamstb	=4ff8	osw3	=4fe7+	wrtlin	86c
gamstr	df3	output	820	wrtvec	1d76
getdlt	1001	ovferr	12aa	xintc	= 2
getlm	1641	ovfmsg	12b4	xmax	= 98
getnum	= 4e	ovrchk	1175	xmem	10ca
getpar	= 4c	plscp	=4ef0	xmem1	10d0
getrdy	1d7a	p2scp	=4f0e	xnib	b7b
getrot	172e	p3scp	=4f2c	xnib1	b8c
gfbody	1f0c	p4scp	=4f4a	xpand	= 19
gfcols	1dc3	packer	12d8+	xpndon	= 1+
gflfr	1bc1	patdim	= 104+	xtab1	= 28
gfstrt	=17de	patxsz	= 1	xtab2	= 50
gfvsiz	= 17	patysz	= 4	xtab3	= 78
gf writ	1b59	paws	= 50	xyrell	291
gf wrt1	1ba0	pblp	1c7+	ylines	= 15
gf wrt2	1ba8	pcdop	= 6	ymax	= 5b
gf wrt3	1bb0+	pcdop0	174a+	ytab	= 14
gf wrt4	1baa+	pcdop1	1754+	ytab1	= 1f
gf wrt5	1bbc	pcdop2	175e+	ytab2	= 33
gmem1	107b	pcdop3	1768+	ytab3	= 47
gmovr	c57	percen	1123	zerol	fcf+
gnacc	2c0	pfug	= 8	zero2	ffd
gnumd0	d4c	phot	40b	zeropl	f50
gotbit	14a0	pizbrk	= 48	zok	1828
gotit	1644	pjoy	1899	zore	1813
gotme	1f02	play0	=4fa8	zp0	f63

Home Video Game System Library - Symbol Table

gotmov	14f6	play1	=4faf	zpa	f69
gotnpl	13aa	play2	=4fb6	zpb	f77
gotsw	14db	play3	=4fbd	zpc	f7d
gout	502	plix	=4fa3	zpd	f93
grx	= 2c	plrom0	1744	zpsiz	=4ef3
gry	= 1	plrom1	174e	zpstmr	=4ef0
gs0	= ee	plrom2	1758	zsw	14c7+
gs1	= 77	plrom3	1762		
gs2	= 3b+	pncalc	ddd		