

```

0000:0000 ;
0000:0000 ;
0000:0000 ;
0000:0000 ; This file is generated by The Interactive Disassembler (IDA)
0000:0000 ; Licensed to: Unknown User ;-
0000:0000 ; Copyright (c) 1999 by DataRescue sa/nv, <ida@datarescue.com>
0000:0000 ;
0000:0000 ;
0000:0000 ;
0000:0000 ; File Name : E:\Projects\NeoKong\arcade\dkong.bin
0000:0000 ; Format : Binary File
0000:0000 ; Base Address: 0000h Range: 0000h - 4000h Loaded length: 4000h
0000:0000 ;
0000:0000 ; Processor: z80
0000:0000 ; Target assembler: Table Driven Assembler (TASM) by Speech Technology Inc.
0000:0000 ;
0000:0000 ;
0000:0000 ; Segment type: Pure code
0000:0000 ; segment 'ROM'
0000:0000
0000:0000 RESET: ; CODE XREF: 0000:00B2|j
0000:0000 3E 00 ; DATA XREF: 0000:0FCD|o
0000:0000 ld a, 0
0000:0002 32 84 7D ld (nmi_mask), a
0000:0005 C3 66 02 jp INIT
0000:0008 ; SUBROUTINE
0000:0008
0000:0008 return_if_attract_mode: ; CODE XREF: flash_1UP_or_2UP+7|p
0000:0008 3A 07 60 ; add_bonus_and_update_high_score+1|p ...
0000:0008 ld a, (attract_mode_flag)
0000:000B 0F rrca ; in attract mode?
0000:000C D0 ret nc ; no, return
0000:000D 33 inc sp
0000:000E 33 inc sp ; discard return address
0000:000F C9 ret
0000:000F ; End of function return_if_attract_mode
0000:0010 ; SUBROUTINE
0000:0010
0000:0010 return_if_mario_not_alive: ; CODE XREF: sub_0_3A2+3|p
0000:0010 3A 00 62 ; sub_0_2C03+3|p ...
0000:0010 ld a, (mario_alive_flag)
0000:0013 0F rrca ; is mario alive?
0000:0014 D8 ret c ; yes, return
0000:0015 33 inc sp
0000:0016 33 inc sp ; discard return address
0000:0017 C9 ret
0000:0017 ; End of function return_if_mario_not_alive
0000:0018 ; SUBROUTINE
0000:0018
0000:0018 return_NOT_8bit_timeout: ; CODE XREF: return_NOT_16bit_timeout+4|j
0000:0018 21 09 60 ; 0000:0A63|p ...
0000:0018 ld hl, eight_bit_countdown
0000:001B 35 dec (hl)
0000:001C C8 ret z
0000:001D 33 inc sp
0000:001E 33 inc sp ; discard return address
0000:001F C9 ret
0000:001F ; End of function return_NOT_8bit_timeout
0000:0020 ; SUBROUTINE
0000:0020
0000:0020 return_NOT_16bit_timeout: ; CODE XREF: 0000:0763|p
0000:0020 21 08 60 ; 0000:084B|p
0000:0020 ld hl, sixteen_bit_countdown_msb
0000:0023 35 dec (hl)
0000:0024 28 F2 jr z, return_NOT_8bit_timeout
0000:0026 pop_hl_ret: ; CODE XREF: print_message_A+1A|j
0000:0026 E1 ; sub_0_1783+4|j
0000:0026 pop hl ; discard return address
0000:0027 C9 ret
0000:0027 ; End of function return_NOT_16bit_timeout
0000:0028 ; SUBROUTINE
0000:0028
0000:0028 jump_table_go_A: ; CODE XREF: 0000:00C9|p
0000:0028 87 ; 0000:0701|p ...
0000:0028 add a, a ; entries are words
0000:0029 E1 pop hl ; return address is table base
0000:002A 5F ld e, a
0000:002B 16 00 ld d, 0 ; DE = offset
0000:002D C3 32 00 jp loc_0_32 ; skip vector address
0000:002D ; End of function jump_table_go_A
0000:0030 ; SUBROUTINE
0000:0030
0000:0030 sub_0_30: ; CODE XREF: sub_0_3A2+2|p
0000:0030 18 12 ; 0000:1668|p ...
0000:0030 jr return_if_level_bit_not_set
0000:0032 ;
0000:0032 loc_0_32: ; CODE XREF: jump_table_go_A+5|j
0000:0032 19 add hl, de ; get address of entry
0000:0033 5E ld e, (hl)
0000:0034 23 inc hl
0000:0035 56 ld d, (hl) ; DE = jump address
0000:0036 EB ex de, hl ; HL = jump address
0000:0037 E9 jp (hl) ; go
0000:0038 ;
0000:0038
0000:0038
0000:0038

```

```

0000:0038      add_c_sprite_register_x10:                ; CODE XREF: animate_kong_and_pauline+F|p
0000:0038 11 04 00      ld      de, 4                ; animate_kong_and_pauline+65|p ...
0000:0038      ld      b, 10              ; every 4th byte
0000:003B 06 0A      ld      b, 10              ; loop 10 times
0000:003D      loc_0_3D:                ; CODE XREF: sub_0_30+11|j
0000:003D 79      ; 0000:0D9A|p ...
0000:003D      ld      a, c
0000:003E 86      add     a, (hl)
0000:003F 77      ld      (hl), a            ; (HL)+=C
0000:0040 19      add     hl, de              ; next byte
0000:0041 10 FA      djnz    loc_0_3D           ; loop
0000:0043 C9      ret
0000:0044      ;
0000:0044      return_if_level_bit_not_set:        ; CODE XREF: sub_0_30|j
0000:0044 21 27 62      ld      hl, level_type
0000:0047 46      ld      b, (hl)            ; get level type
0000:0048      loc_0_48:                ; CODE XREF: sub_0_30+19|j
0000:0048 0F      rrca
0000:0049 10 FD      djnz    loc_0_48           ; get bit of A for level
0000:004B D8      ret     c                  ; bit set, return
0000:004C E1      pop     hl                  ; discard return address
0000:004D C9      ret
0000:004D      ; End of function sub_0_30
0000:004D      ;
0000:004E      ; SUBROUTINE
0000:004E      ;
0000:004E      copy_sprites_2_11_data:            ; CODE XREF: animate_kong_and_pauline+4D|p
0000:004E 11 08 69      ld      de, soft_sprite_ram+8      ; animate_kong_and_pauline+77|p ...
0000:0051 01 28 00      ld      bc, 40                ; ptr sprite #2
0000:0054 ED B0      ldir                                ; 10 4-byte sprites to copy
0000:0056 C9      ret                            ; copy 40 bytes of sprite data
0000:0056      ; End of function copy_sprites_2_11_data
0000:0056      ;
0000:0057      ; SUBROUTINE
0000:0057      ;
0000:0057      rand:                        ; CODE XREF: 0000:00B9|p
0000:0057 3A 18 60      ld      a, (random_no)         ; sub_0_2523+22|p ...
0000:0057      ld      hl, gen_purpose_timer
0000:005A 21 1A 60      add     a, (hl)
0000:005D 86      ;
0000:005E      loc_0_5E:
0000:005E 21 19 60      ld      hl, random_no+1
0000:0061 86      add     a, (hl)
0000:0062 32 18 60      ld      (random_no), a
0000:0065 C9      ret
0000:0065      ; End of function rand
0000:0065      ;
0000:0066      ;
0000:0066      nmi:
0000:0066 F5      push    af
0000:0067 C5      push    bc
0000:0068 D5      push    de
0000:0069 E5      push    hl
0000:006A DD E5      push    ix
0000:006C FD E5      push    iy
0000:006E AF      xor     a
0000:006F 32 84 7D      ld      (nmi_mask), a        ; disable_nmi
0000:0072 3A 00 7D      ld      a, (in2_snd_latch)   ; IN2
0000:0075 E6 01      and     1                    ; bit 0 set?
0000:0077 C2 00 40      jp      nz, 4000h             ; yes, boom! (not valid code)
0000:007A 21 38 01      ld      hl, dma_reg_tbl
0000:007D CD 41 01      call    dma_sprite_data_to_hw ; update sprites
0000:0080 3A 07 60      ld      a, (attract_mode_flag)
0000:0083 A7      and     a                    ; in attract mode?
0000:0084 C2 B5 00      jp      nz, loc_0_B5          ; yes, skip reading inputs
0000:0087 3A 26 60      ld      a, (upright)
0000:008A A7      and     a
0000:008B C2 98 00      jp      nz, loc_0_98          ; player 2?
0000:008E 3A 0E 60      ld      a, (current_player_E) ; (cocktail)
0000:0091 A7      and     a                    ; yes, skip
0000:0092 3A 80 7C      ld      a, (in1)
0000:0095 C2 9B 00      jp      nz, loc_0_9B
0000:0098      loc_0_98:                ; CODE XREF: 0000:008B|j
0000:0098 3A 00 7C      ld      a, (in0)              ; (upright)
0000:009B      loc_0_9B:                ; CODE XREF: 0000:0095|j
0000:009B 47      ld      b, a                  ; store IN0/1
0000:009C E6 0F      and     0Fh                  ; joystick only
0000:009E 4F      ld      c, a                  ; store
0000:009F 3A 11 60      ld      a, (last_raw_in)      ; last raw input
0000:00A2 2F      cpl                                ; negate
0000:00A3 A0      and     b                      ; rising-edge detect
0000:00A4 E6 10      and     10h                  ; button
0000:00A6 17      rla
0000:00A7 17      rla
0000:00A8 17      rla
0000:00A9 B1      or      c                      ; bit 7
0000:00AA 60      ld      h, b                  ; add joystick bits
0000:00AB 6F      ld      l, a                  ; raw controller input
0000:00AC 22 10 60      ld      (controller_in), hl   ; joystick and button press
0000:00AF 78      ld      a, b                  ; store
0000:00B0 CB 77      bit     6, a                  ; reset input?
0000:00B2 C2 00 00      jp      nz, RESET
0000:00B5      loc_0_B5:                ; CODE XREF: 0000:0084|j
0000:00B5 21 1A 60      ld      hl, gen_purpose_timer   ; general purpose timer tick
0000:00B8 35      dec     (hl)                  ; randomise
0000:00B9 CD 57 00      call    rand
0000:00BC CD 7B 01      call    sub_0_17B
0000:00BF CD E0 00      call    update_sounds
0000:00C2 21 D2 00      ld      hl, nmi_exit          ; IRQ resume address
0000:00C5 E5      push    hl
0000:00C6 3A 05 60      ld      a, (nmi_sequencer)
0000:00C9 EF      rst     28h                  ; go!
0000:00C9      ;

```

```

0000:00CA C3 01      .dw init_machine_settings          ; Jump table (nmi sequencer)
0000:00CC 3C 07      .dw chk_credits_and_vector_on_attrac
0000:00CE B2 08      .dw vector_on_credit_sequencer
0000:00D0 FE 06      .dw vector_on_ingame_sequencer
0000:00D2
0000:00D2
0000:00D2
nmi_exit:
0000:00D2 FD E1      pop      iy
0000:00D4 DD E1      pop      ix
0000:00D6 E1      pop      hl
0000:00D7 D1      pop      de
0000:00D8 C1      pop      bc
0000:00D9 3E 01      ld      a, 1
0000:00DB 32 84 7D      ld      (nmi_mask), a
0000:00DE F1      pop      af
0000:00DF C9      ret
0000:00E0
0000:00E0
0000:00E0
0000:00E0
0000:00E0
; SUBROUTINE
0000:00E0
update_sounds:
0000:00E0 21 80 60      ld      hl, digital_snd_tmr_walk    ; CODE XREF: 0000:00BF|p
0000:00E3 11 00 7D      ld      de, in2_snd_latch          ; base of digital sound triggers
0000:00E6 3A 07 60      ld      a, (attract_mode_flag)
0000:00E9 A7      and     a
0000:00EA C0      ret      nz
0000:00EB 06 08      ld      b, 8
0000:00ED
loc_0_ED:
0000:00ED 7E      ld      a, (hl)
0000:00EE A7      and     a
0000:00EF CA F5 00      jp      z, loc_0_F5
0000:00F2 35      dec     (hl)
0000:00F3 3E 01      ld      a, 1
0000:00F5
loc_0_F5:
0000:00F5 12      ld      (de), a
0000:00F6 1C      inc     e
0000:00F7 2C      inc     l
0000:00F8 10 F3      djnz    loc_0_ED
0000:00FA 21 8B 60      ld      hl, unk_0_608B
0000:00FD 7E      ld      a, (hl)
0000:00FE A7      and     a
0000:00FF C2 08 01      jp      nz, loc_0_108
0000:0102 2D      dec     l
0000:0103 2D      dec     l
0000:0104 7E      ld      a, (hl)
0000:0105 C3 0B 01      jp      set_bg_sound_music
0000:0108
0000:0108
loc_0_108:
0000:0108 35      dec     (hl)
0000:0109 2D      dec     l
0000:010A 7E      ld      a, (hl)
0000:010B
set_bg_sound_music:
0000:010B 32 00 7C      ld      (in0), a
0000:010E 21 88 60      ld      hl, unk_0_6088
0000:0111 AF      xor     a
0000:0112 BE      cp      (hl)
0000:0113 CA 18 01      jp      z, loc_0_118
0000:0116 35      dec     (hl)
0000:0117 3C      inc     a
0000:0118
loc_0_118:
0000:0118 32 80 7D      ld      (dsw_audio_irq), a
0000:011B C9      ret
; End of function update_sounds
0000:011B
0000:011C
0000:011C
0000:011C
0000:011C
0000:011C
stop_sound:
0000:011C 06 08      ld      b, 8
0000:011E AF      xor     a
0000:0121 21 00 7D      ld      hl, in2_snd_latch
0000:0122 11 80 60      ld      de, digital_snd_tmr_walk
0000:0125
loc_0_125:
0000:0125 77      ld      (hl), a
0000:0126 12      ld      (de), a
0000:0127 2C      inc     l
0000:0128 1C      inc     e
0000:0129 10 FA      djnz    loc_0_125
0000:012B 06 04      ld      b, 4
0000:012D
loc_0_12D:
0000:012D 12      ld      (de), a
0000:012E 1C      inc     e
0000:012F 10 FC      djnz    loc_0_12D
0000:0131 32 80 7D      ld      (dsw_audio_irq), a
0000:0134 32 00 7C      ld      (in0), a
0000:0137 C9      ret
; End of function stop_sound
0000:0137
0000:0137
0000:0137
dma_reg_tbl:
0000:0138 53      .db 53h
0000:0138
0000:0139 00 69      .dw soft_sprite_ram
0000:013B 80 41      .dw 4180h
0000:013D 00 70      .dw SPRAM_start
0000:013F 80 81      .dw 8180h
0000:0141
0000:0141
0000:0141
0000:0141
0000:0141
dma_sprite_data_to_hw:
0000:0141 AF      xor     a
0000:0142 32 85 7D      ld      (p8257_drq), a
0000:0145 7E      ld      a, (hl)
0000:0146 32 08 78      ld      (i8257_io+8), a
0000:0149 23      inc     hl
0000:014A 7E      ld      a, (hl)
0000:014B 32 00 78      ld      (i8257_io), a

```

```

0000:014E 23      inc    hl
0000:014F 7E      ld     a, (hl)                ; 0x69
0000:0150 32 00 78    ld     (i8257_io), a          ; CH0 DMA address MSB
0000:0153 23      inc    hl
0000:0154 7E      ld     a, (hl)                ; 0x80
0000:0155 32 01 78    ld     (i8257_io+1), a        ; terminal count LSB
0000:0158 23      inc    hl
0000:0159 7E      ld     a, (hl)                ; 0x41
0000:015A 32 01 78    ld     (i8257_io+1), a        ; terminal count MSB
0000:015D 23      inc    hl
0000:015E 7E      ld     a, (hl)                ; 0x00
0000:015F 32 02 78    ld     (i8257_io+2), a        ; CH1 DMA address LSB
0000:0162 23      inc    hl
0000:0163 7E      ld     a, (hl)                ; 0x70
0000:0164 32 02 78    ld     (i8257_io+2), a        ; CH1 DMA address MSB
0000:0167 23      inc    hl
0000:0168 7E      ld     a, (hl)                ; 0x80
0000:0169 32 03 78    ld     (i8257_io+3), a        ; CH1 terminal count LSB
0000:016C 23      inc    hl
0000:016D 7E      ld     a, (hl)                ; 0x81
0000:016E 32 03 78    ld     (i8257_io+3), a        ; CH1 terminal count MSB
0000:0171 3E 01      ld     a, 1
0000:0173 32 85 7D    ld     (p8257_drq), a         ; assert DRQ0&1
0000:0176 AF      xor    a
0000:0177 32 85 7D    ld     (p8257_drq), a         ; deassert DRQ0&1
0000:017A C9      ret
0000:017A      ; End of function dma_sprite_data_to_hw
0000:017A
0000:017B      ; ██████████ SUBROUTINE ██████████
0000:017B
0000:017B      sub_0_17B:
0000:017B 3A 00 7D      ld     a, (in2_snd_latch)      ; CODE XREF: 0000:00BC↑p
0000:017E CB 7F      bit    7, a                   ; read IN2
0000:0180 21 03 60      ld     hl, unk_0_6003         ; coin?
0000:0183 C2 89 01      jp     nz, loc_0_189          ; yes, skip
0000:0186 36 01      ld     (hl), 1
0000:0188 C9      ret
0000:0189
0000:0189      ;
0000:0189      loc_0_189:
0000:0189 7E      ld     a, (hl)                ; CODE XREF: sub_0_17B+8↑j
0000:018A A7      and    a
0000:018B C8      ret    z
0000:018C E5      push   hl
0000:018D 3A 05 60      ld     a, (nmi_sequencer)
0000:0190 FE 03      cp     3                      ; in credit sequence?
0000:0192 CA 9D 01      jp     z, loc_0_19D           ; yes, skip
0000:0195 CD 1C 01      call   stop_sound
0000:0198 3E 03      ld     a, 3                   ; tmr = 3
0000:019A 32 83 60      ld     (digital_snd_tmr_coin_spring), a
0000:019D
0000:019D      loc_0_19D:
0000:019D E1      pop     hl                    ; CODE XREF: sub_0_17B+17↑j
0000:019E 36 00      ld     (hl), 0
0000:01A0 2B      dec    hl
0000:01A1 34      inc    (hl)
0000:01A2 11 24 60      ld     de, coinage+2
0000:01A5 1A      ld     a, (de)
0000:01A6 96      sub    (hl)
0000:01A7 C0      ret    nz
0000:01A8 77      ld     (hl), a
0000:01A9 13      inc    de
0000:01AA 2B      dec    hl
0000:01AB EB      ex     de, hl
0000:01AC 1A      ld     a, (de)
0000:01AD FE 90      cp     90h ; 'É'
0000:01AF D0      ret    nc
0000:01B0 86      add    a, (hl)
0000:01B1 27      daa
0000:01B2 12      ld     (de), a
0000:01B3 11 00 04      ld     de, 400h
0000:01B6 CD 9F 30      call   sub_0_309F
0000:01B9 C9      ret
0000:01B9      ; End of function sub_0_17B
0000:01B9
0000:01B9      ;
0000:01B9      ;
0000:01BA 00 37 00      init_scores_and_high_score: .db 0, 37h, 0 ; DATA XREF: 0000:01C6↑o
0000:01BA      ; Initial score and high score on bootup
0000:01BD AA AA AA      byte_0_1BD: .db 0AAh, 0AAh, 0AAh ; DATA XREF: 0000:159D↑o
0000:01C0 50 76 00      ;
0000:01C3
0000:01C3      init_machine_settings:
0000:01C3 CD 74 08      call   clear_visible_area_and_sprites ; DATA XREF: 0000:00CA↑o
0000:01C6 21 BA 01      ld     hl, init_scores_and_high_score ; copy in ROM
0000:01C9 11 B2 60      ld     de, pl_score            ; RAM location
0000:01CC 01 09 00      ld     bc, 9                   ; 9 bytes to copy
0000:01CF ED B0      ldir                                ; copy scores to RAM
0000:01D1 3E 01      ld     a, 1
0000:01D3 32 07 60      ld     (attract_mode_flag), a ; set attract mode flag
0000:01D6 32 29 62      ld     (level), a
0000:01D9 32 28 62      ld     (lives_left), a
0000:01DC CD B8 06      call   display_lives_and_level
0000:01DF CD 07 02      call   read_dips_and_high_score_tbl
0000:01E2 3E 01      ld     a, 1
0000:01E4 32 82 7D      ld     (flipscreen), a
0000:01E7 32 05 60      ld     (nmi_sequencer), a ; next sequence
0000:01EA 32 27 62      ld     (level_type), a
0000:01ED AF      xor    a
0000:01EE 32 0A 60      ld     (main_sequencer), a ; game screen sequencer
0000:01F1 CD 53 0A      call   display_lup
0000:01F4 11 04 03      ld     de, 304h
0000:01F7 CD 9F 30      call   sub_0_309F
0000:01FA 11 02 02      ld     de, 202h
0000:01FD CD 9F 30      call   sub_0_309F
0000:0200 11 00 02      ld     de, 200h
0000:0203 CD 9F 30      call   sub_0_309F
0000:0206 C9      ret
0000:0207
0000:0207      ; ██████████ SUBROUTINE ██████████
0000:0207
0000:0207      read_dips_and_high_score_tbl:
0000:0207 3A 80 7D      ld     a, (dsw_audio_irq)      ; CODE XREF: 0000:01DF↑p
0000:0207      ; read DIPSW

```

```

0000:020A 4F      ld      c, a                ; store
0000:020B 21 20 60    ld      hl, lives_per_game
0000:020E E6 03      and     3                  ; lives setting
0000:0210 C6 03      add     a, 3               ; init no. of lives
0000:0212 77      ld      (hl), a            ; store no. of lives
0000:0213 23      inc     hl
0000:0214 79      ld      a, c               ; DIPSW
0000:0215 0F      rrca
0000:0216 0F      rrca
0000:0217 E6 03      and     3                  ; bonus life setting
0000:0219 47      ld      b, a
0000:021A 3E 07      ld      a, 7               ; 7,000?
0000:021C CA 26 02  jp      z, loc_0_226       ; yes, skip
0000:021F 3E 05      ld      a, 5               ; 5,000?
0000:0221      loc_0_221:
0000:0221 C6 05      add     a, 5               ; CODE XREF: read_dips_and_high_score_tbl+1D|j
0000:0223 27      daa
0000:0224 10 FB      djnz    loc_0_221         ; calculate 10/15/20K points
0000:0226      loc_0_226:
0000:0226 77      ld      (hl), a            ; CODE XREF: read_dips_and_high_score_tbl+15|j
0000:0227 23      inc     hl                 ; bonus_setting
0000:0228 79      ld      a, c               ; DIPSW
0000:0229 01 01 01  ld      bc, 101h           ; 1C P1
0000:022C 11 02 01  ld      de, 102h           ; 1C P2
0000:022F E6 70      and     70h ; 'p'         ; coinage setting
0000:0231 17      rla
0000:0232 17      rla
0000:0233 17      rla
0000:0234 17      rla
0000:0235 CA 47 02  jp      z, loc_0_247       ; coinage 0-7
0000:0238 DA 41 02  jp      c, loc_0_241       ; 1C1C
0000:023B 3C      inc     a                  ; 2-5 coins
0000:023C 4F      ld      c, a               ; no. credits
0000:023D 5A      ld      e, d               ; C = credits
0000:023E C3 47 02  jp      loc_0_247         ; D = coins
0000:0241      ; -----
0000:0241      loc_0_241:
0000:0241 C6 02      add     a, 2               ; CODE XREF: read_dips_and_high_score_tbl+31|j
0000:0243 47      ld      b, a               ; no. coins
0000:0244 57      ld      d, a               ; B = coins
0000:0245 87      add     a, a               ; D = coins
0000:0246 5F      ld      e, a               ; E = coins x2
0000:0247      loc_0_247:
0000:0247 72      ; CODE XREF: read_dips_and_high_score_tbl+2E|j
0000:0247 72      ; read_dips_and_high_score_tbl+37|j
0000:0247      ld      (hl), d
0000:0248 23      inc     hl
0000:0249 73      ld      (hl), e
0000:024A 23      inc     hl
0000:024B 70      ld      (hl), b
0000:024C 23      inc     hl
0000:024D 71      ld      (hl), c
0000:024E 23      inc     hl
0000:024F 3A 80 7D  ld      a, (dsw_audio_irq) ; read DIPSW
0000:0252 07      rlca                       ; upright?
0000:0253 3E 01      ld      a, 1
0000:0255 DA 59 02  jp      c, loc_0_259       ; yes, skip
0000:0258 3D      dec     a
0000:0259      loc_0_259:
0000:0259 77      ; CODE XREF: read_dips_and_high_score_tbl+4E|j
0000:0259 77      ; store cocktail/upright
0000:025A 21 65 35  ld      hl, high_score_tbl
0000:025D 11 00 61  ld      de, high_score_tbl_ram
0000:0260 01 AA 00  ld      bc, 0AAh ; '-'
0000:0263 ED B0  ldir
0000:0265 C9      ret                         ; copy to ram
0000:0265      ; End of function read_dips_and_high_score_tbl
0000:0265      ; -----
0000:0266      INIT:
0000:0266      ld      b, 16              ; CODE XREF: 0000:0005|j
0000:0266 06 10      ld      hl, RAM_start     ; start of RAM
0000:0268 21 00 60  xor     a                  ; zero byte
0000:026B AF      loc_0_26C:
0000:026C      ld      c, a               ; CODE XREF: 0000:0272|j
0000:026C 4F      loc_0_26D:
0000:026D      ld      (hl), a            ; CODE XREF: 0000:0270|j
0000:026D 77      ; zero memory
0000:026E 23      inc     hl                 ; next location
0000:026E 23      dec     c
0000:026F 0D      jr      nz, loc_0_26D      ; clear 256 bytes
0000:0270 20 FB      djnz    loc_0_26C         ; clear 4K bytes
0000:0272 10 F8      ld      b, 4
0000:0274 06 04      ld      hl, SPRAM_start   ; start of sprite RAM
0000:0276 21 00 70  loc_0_279:
0000:0279      ld      c, a               ; CODE XREF: 0000:027F|j
0000:0279 4F      loc_0_27A:
0000:027A      ld      (hl), a            ; CODE XREF: 0000:027D|j
0000:027A 77      ; zero memory
0000:027B 23      inc     hl                 ; next location
0000:027C 0D      dec     c
0000:027C 0D      jr      nz, loc_0_27A      ; clear 256 bytes
0000:027D 20 FB      djnz    loc_0_279         ; clear 1K bytes
0000:027F 10 F8      ld      b, 4
0000:0281 06 04      ld      a, 10h            ; space character
0000:0283 3E 10      ld      hl, VRAM_start    ; start of VRAM
0000:0285 21 00 74  loc_0_288:
0000:0288      ld      c, 0               ; CODE XREF: 0000:028F|j
0000:0288 0E 00      loc_0_28A:
0000:028A      ld      (hl), a            ; CODE XREF: 0000:028D|j
0000:028A 77      ; clear memory
0000:028B 23      inc     hl                 ; next location
0000:028C 0D      dec     c
0000:028C 0D      jr      nz, loc_0_28A      ; clear 256 bytes
0000:028D 20 FB      djnz    loc_0_288         ; clear 1K bytes
0000:028F 10 F7      ld      hl, fg_vector_fn_params
0000:0291 21 C0 60  ld      b, 64
0000:0294 06 40      ld      a, 0FFh           ; fill 64 bytes
0000:0296 3E FF      ld      a, 0FFh           ; fill byte
0000:0298

```

```

0000:0298      loc_0_298:      ld      (hl), a          ; CODE XREF: 0000:029A|j
0000:0298 77          inc      hl              ; set to $FF
0000:0298 23          djnz    loc_0_298      ; next location
0000:029A 10 FC      ld      a, 0C0h ; 'L' ; set 64 bytes
0000:029C 3E C0      ld      (unk_0_60B0), a
0000:029E 32 B0 60   ld      (unk_0_60B1), a
0000:02A1 32 B1 60   ld      a
0000:02A4 AF      xor      a
0000:02A5 32 83 7D   ld      (spritebank), a
0000:02A8 32 86 7D   ld      (palette_bank), a          ; b0=0
0000:02AB 32 87 7D   ld      (palette_bank+1), a        ; b1=0
0000:02AE 3C          inc      a
0000:02AF 32 82 7D   ld      (flipscreen), a
0000:02B2 31 00 6C   ld      sp, 6C00h
0000:02B5 CD 1C 01   call    stop_sound
0000:02B8 3E 01      ld      a, 1
0000:02BA 32 84 7D   ld      (nmi_mask), a          ; enable interrupts
0000:02BD      ;
0000:02BD      main_loop:          ; CODE XREF: 0000:02D8|j
0000:02BD 26 60      ; 0000:02E1|j
0000:02BD      ; DATA XREF: ...
0000:02BD      ; MSB of parameter table in RAM
0000:02BF 3A B1 60   ld      h, 60h ; ''
0000:02C2 6F      ld      a, (unk_0_60B1)
0000:02C3 7E      ld      l, a
0000:02C4 87      ld      a, (hl)
0000:02C5 30 1C   add     a, a
0000:02C7 CD 15 03   jr      nc, loc_0_2E3
0000:02CA CD 50 03   call    flash_1UP_or_2UP
0000:02CD 21 19 60   call    check_and_award_bonus
0000:02D0 34      ld      hl, random_no+1          ; random LSB
0000:02D1 21 83 63   inc     (hl)                    ; INC
0000:02D4 3A 1A 60   ld      hl, unk_0_6383
0000:02D7 BE      ld      a, (gen_purpose_timer)
0000:02D8 28 E3      cp      (hl)                    ; same?
0000:02DA 77      jr      z, main_loop           ; yes, loop
0000:02DB CD 7F 03   ld      (hl), a                ; generate LSB from timer
0000:02DE CD A2 03   call    sub_0_37F
0000:02E1 18 DA      call    sub_0_3A2              ; do something with a sprite
0000:02E3      ;
0000:02E3      ;
0000:02E3      loc_0_2E3:          ; CODE XREF: 0000:02C5|j
0000:02E3 E6 1F      and     1Fh
0000:02E5 5F      ld      e, a
0000:02E6 16 00   ld      d, 0
0000:02E8 36 FF      ld      (hl), 0FFh
0000:02EA 2C      inc     l
0000:02EB 4E      ld      c, (hl)
0000:02EC 36 FF      ld      (hl), 0FFh
0000:02EE 2C      inc     l
0000:02EF 7D      ld      a, l
0000:02F0 FE C0   cp      0C0h ; 'L'
0000:02F2 30 02   jr      nc, loc_0_2F6
0000:02F4 3E C0      ld      a, 0C0h ; 'L'
0000:02F6      ;
0000:02F6      loc_0_2F6:          ; CODE XREF: 0000:02F2|j
0000:02F6 32 B1 60   ld      (unk_0_60B1), a
0000:02F9 79      ld      a, c
0000:02FA 21 BD 02   ld      hl, main_loop
0000:02FD E5      push    hl
0000:02FE 21 07 03   ld      hl, foreground_vector_table ; $02BD is the return address
0000:0301 19      add     hl, de                  ; jump table
0000:0302 5E      ld      e, (hl)                ; entry index
0000:0303 23      inc     hl
0000:0304 56      ld      d, (hl)                ; get entry
0000:0305 EB      ex      de, hl
0000:0306 E9      jp      (hl)                    ; jump
0000:0306      ;
0000:0307 1C 05      foreground_vector_table: .dw add_bonus_and_update_high_score ; DATA XREF: 0000:02FE|o
0000:0307      ; jump table
0000:0309 9B 05      .dw zero_score_or_high_score
0000:030B C6 05      .dw display_score_or_high_score
0000:030D E9 05      .dw print_message_A
0000:030F 11 06      .dw display_credits_if_attract_mode
0000:0311 2A 06      .dw update_bonus_timer
0000:0313 B8 06      .dw display_lives_and_level
0000:0315      ;
0000:0315      ; SUBROUTINE
0000:0315      ;
0000:0315      flash_1UP_or_2UP:   ; CODE XREF: 0000:02C7|p
0000:0315 3A 1A 60   ld      a, (gen_purpose_timer)
0000:0318 47      ld      b, a
0000:0319 E6 0F      and     0Fh
0000:031B C0      ret     nz
0000:031C CF      rst     8                      ; return if attract mode
0000:031D 3A 0D 60   ld      a, (current_player_D)
0000:0320 CD 47 03   call    get_1UP_or_2UP_screen_location
0000:0323 11 E0 FF   ld      de, 0FFE0h
0000:0326 CB 60      bit     4, b                    ; column address offset
0000:0328 28 14      jr      z, loc_0_33E           ; unhide 1UP/2UP?
0000:032A 3E 10      ld      a, 10h
0000:032C 77      ld      (hl), a
0000:032D 19      add     hl, de                  ; wipe "1" or "2"
0000:032E 77      ld      (hl), de                ; next column
0000:032F 19      add     hl, de                  ; wipe "U"
0000:0330 77      ld      (hl), de                ; next column
0000:0331 3A 0F 60   ld      a, (two_players)
0000:0334 A7      and     a
0000:0335 C8      ret     z                      ; 1 player?
0000:0336 3A 0D 60   ld      a, (current_player_D)
0000:0339 EE 01      xor     1
0000:033B CD 47 03   call    get_1UP_or_2UP_screen_location ; yes, return
0000:033E      ;
0000:033E      loc_0_33E:          ; CODE XREF: flash_1UP_or_2UP+13|j
0000:033E 3C          inc     a
0000:033F 77      ld      (hl), a
0000:0340 19      add     hl, de                  ; "1" or "2"
0000:0341 36 25      ld      (hl), 25h ; '%'        ; next column
0000:0343 19      add     hl, de                  ; "U"
0000:0344 36 20      ld      (hl), 20h ; ' '        ; next column
0000:0346 C9      ret
0000:0346      ; End of function flash_1UP_or_2UP
0000:0347      ;
0000:0347      ; SUBROUTINE
0000:0347      ;

```

```

0000:0347
0000:0347
0000:0347
0000:0347 21 40 77 get_LUP_or_2UP_screen_location: ; CODE XREF: flash_LUP_or_2UP+B|p
; flash_LUP_or_2UP+26|p
0000:0347 ld hl, VRAM_start+340h ; ptr "LUP" screen location
0000:034A A7 and a ; player 1?
0000:034B C8 ret z ; yes, return
0000:034C 21 E0 74 ld hl, VRAM_start+0E0h ; ptr "2UP" screen location
0000:034F C9 ret
; End of function get_LUP_or_2UP_screen_location
0000:034F
0000:0350
0000:0350 ; SUBROUTINE
0000:0350
0000:0350
0000:0350 check_and_award_bonus: ; CODE XREF: 0000:02CA|p
0000:0350 3A 2D 62 ld a, (awarded_bonus_life)
0000:0353 A7 and a ; already got bonus life?
0000:0354 C0 ret nz ; yes, return
0000:0355 21 B3 60 ld hl, p1_score+1
0000:0358 3A 0D 60 ld a, (current_player_D)
0000:035B A7 and a ; player 1?
0000:035C 28 03 jr z, loc_0_361 ; yes, skip
0000:035E 21 B6 60 ld hl, p2_score+1
0000:0361
0000:0361 loc_0_361: ; CODE XREF: check_and_award_bonus+C|j
0000:0361 7E ld a, (hl) ; get hundreds from score
0000:0362 E6 F0 and 0F0h ; only thousands
0000:0364 47 ld b, a ; save
0000:0365 23 inc hl ; next score byte
0000:0366 7E ld a, (hl) ; get tens of thousands
0000:0367 E6 0F and 0Fh ; only tens of thousands
0000:0369 B0 or b ; B = thousands (and tens of)
0000:036A 0F rrca
0000:036B 0F rrca
0000:036C 0F rrca
0000:036D 0F rrca ; swap nibbles
0000:036E 21 21 60 ld hl, bonus_setting
0000:0371 BE cp (hl) ; reached bonus score?
0000:0372 D8 ret c ; no, return
0000:0373 3E 01 ld a, 1
0000:0375 32 2D 62 ld (awarded_bonus_life), a ; flag that we've got the bonus
0000:0378 21 28 62 ld hl, lives_left
0000:037B 34 inc (hl) ; extra life
0000:037C C3 B8 06 jp display_lives_and_level
; End of function check_and_award_bonus
0000:037C
0000:037F
0000:037F ; SUBROUTINE
0000:037F
0000:037F
0000:037F sub_0_37F: ; CODE XREF: 0000:02DB|p
0000:037F 21 84 63 ld hl, unk_0_6384
0000:0382 7E ld a, (hl)
0000:0383 34 inc (hl)
0000:0384 A7 and a
0000:0385 C0 ret nz
0000:0386 21 81 63 ld hl, unk_0_6381
0000:0389 7E ld a, (hl)
0000:038A 47 ld b, a
0000:038B 34 inc (hl)
0000:038C E6 07 and 7
0000:038E C0 ret nz
0000:038F 78 ld a, b
0000:0390 0F rrca
0000:0391 0F rrca
0000:0392 0F rrca
0000:0393 47 ld b, a
0000:0394 3A 29 62 ld a, (level)
0000:0397 80 add a, b
0000:0398 FE 05 cp 5
0000:039A 38 02 jr c, loc_0_39E
0000:039C 3E 05 ld a, 5
0000:039E
0000:039E loc_0_39E: ; CODE XREF: sub_0_37F+1B|j
0000:039E 32 80 63 ld (unk_0_6380), a
0000:03A1 C9 ret
; End of function sub_0_37F
0000:03A1
0000:03A2
0000:03A2 ; SUBROUTINE
0000:03A2
0000:03A2
0000:03A2 sub_0_3A2: ; CODE XREF: 0000:02DE|p
0000:03A2 3E 03 ld a, 3
0000:03A4 F7 rst 30h ; return if level bit not set
0000:03A5 D7 rst 10h ; return if mario not alive
0000:03A6 3A 50 63 ld a, (unk_0_6350)
0000:03A9 0F rrca
0000:03AA D8 ret c
0000:03AB 21 B8 62 ld hl, unk_0_62B8
0000:03AE 35 dec (hl)
0000:03AF C0 ret nz
0000:03B0 36 04 ld (hl), 4
0000:03B2 3A B9 62 ld a, (unk_0_62B9)
0000:03B5 0F rrca
0000:03B6 D0 ret nc
0000:03B7 21 29 6A ld hl, soft_sprite_ram+129h ; sprite #173, flipy & code
0000:03BA 06 40 ld b, 40h ; '@'
0000:03BC DD 21 A0 66 ld ix, unk_0_66A0
0000:03C0 0F rrca
0000:03C1 D2 E4 03 jp nc, loc_0_3E4
0000:03C4 DD 36 09 02 ld (ix+9), 2
0000:03C8 DD 36 0A 02 ld (ix+0Ah), 2
0000:03CC 04 inc b
0000:03CD 04 inc b
0000:03CE CD F2 03 call sub_0_3F2
0000:03D1 21 BA 62 ld hl, unk_0_62BA
0000:03D4 35 dec (hl)
0000:03D5 C0 ret nz
0000:03D6 3E 01 ld a, 1
0000:03D8 32 B9 62 ld (unk_0_62B9), a
0000:03DB 32 A0 63 ld (unk_0_63A0), a
0000:03DE
0000:03DE loc_0_3DE: ; CODE XREF: sub_0_3A2+4D|j
0000:03DE 3E 10 ld a, 10h

```

```

0000:03E0 32 BA 62      ld      (unk_0_62BA), a
0000:03E3 C9          ret
0000:03E4          ;
0000:03E4          loc_0_3E4:
0000:03E4          ; CODE XREF: sub_0_3A2+1F↑j
0000:03E4 DD 36 09 02      ld      (ix+9), 2
0000:03E8 DD 36 0A 00      ld      (ix+0Ah), 0
0000:03EC CD F2 03      call   sub_0_3F2
0000:03EF C3 DE 03      jp      loc_0_3DE
0000:03EF          ; End of function sub_0_3A2
0000:03EF
0000:03F2          ;
0000:03F2          ; SUBROUTINE
0000:03F2
0000:03F2          sub_0_3F2:
0000:03F2          ; CODE XREF: sub_0_3A2+2C↑p
0000:03F2 70          ; sub_0_3A2+4A↑p
0000:03F2          ld      (hl), b
0000:03F3 3A 19 60      ld      a, (random_no+1)
0000:03F6 0F          rrca
0000:03F7 D8          ret      c
0000:03F8 04          inc     b
0000:03F9 70          ld      (hl), b
0000:03FA C9          ret
0000:03FA          ; End of function sub_0_3F2
0000:03FA
0000:03FB          ;
0000:03FB          ; SUBROUTINE
0000:03FB
0000:03FB          animate_kong_and_pauline:
0000:03FB          ; CODE XREF: 0000:19B0↑p
0000:03FB 3A 27 62      ld      a, (level_type)
0000:03FE FE 02      cp      2
0000:0400 C2 13 04      jp      nz, loc_0_413
0000:0403 21 08 69      ld      hl, soft_sprite_ram+8
0000:0406 3A A3 63      ld      a, (unk_0_63A3)
0000:0409 4F          ld      c, a
0000:040A FF          rst      38h
0000:040B 3A 10 69      ld      a, (soft_sprite_ram+10h)
0000:040E D6 3B      sub     59
0000:0410 32 B7 63      ld      (unk_0_63B7), a
0000:0413          loc_0_413:
0000:0413          ; CODE XREF: animate_kong_and_pauline+5↑j
0000:0413 3A 91 63      ld      a, (kong_thrash_flag)
0000:0416 A7          and     a
0000:0417 C2 26 04      jp      nz, loc_0_426
0000:041A 3A 1A 60      ld      a, (gen_purpose_timer)
0000:041D A7          and     a
0000:041E C2 86 04      jp      nz, animate_pauline
0000:0421 3E 01      ld      a, 1
0000:0423 32 91 63      ld      (kong_thrash_flag), a
0000:0426          loc_0_426:
0000:0426          ; CODE XREF: animate_kong_and_pauline+1C↑j
0000:0426 21 90 63      ld      hl, kong_thrash_tmr
0000:0429 34          inc     (hl)
0000:042A 7E          ld      a, (hl)
0000:042B FE 80      cp      128
0000:042D CA 64 04      jp      z, draw_kong_mouth_closed
0000:0430 3A 93 63      ld      a, (barrel_deployment)
0000:0433 A7          and     a
0000:0434 C2 86 04      jp      nz, animate_pauline
0000:0437 7E          ld      a, (hl)
0000:0438 47          ld      b, a
0000:0439 E6 1F          and     31
0000:043B C2 86 04      jp      nz, animate_pauline
0000:043E 21 CF 39      ld      hl, dk_thrash_right_spr
0000:0441 CB 68      bit     5, b
0000:0443 20 03      jr      nz, do_kong_thrash
0000:0445 21 F7 39      ld      hl, dk_thrash_left_spr
0000:0448          do_kong_thrash:
0000:0448          ; CODE XREF: animate_kong_and_pauline+48↑j
0000:0448 CD 4E 00      call   copy_sprites_2_11_data
0000:044B 3E 03      ld      a, 3
0000:044D 32 82 60      ld      (digital_snd_tmr_thump), a
0000:0450          loc_0_450:
0000:0450          ; CODE XREF: animate_kong_and_pauline+7A↑j
0000:0450 3A 27 62      ld      a, (level_type)
0000:0453 0F          rrca
0000:0454 D2 78 04      jp      nc, loc_0_478
0000:0457 0F          rrca
0000:0458 DA 86 04      jp      c, animate_pauline
0000:045B 21 0B 69      ld      hl, soft_sprite_ram+0Bh
0000:045E 0E FC      ld      c, 0FCh ; '3'
0000:0460 FF          rst      38h
0000:0461 C3 86 04      jp      subtract_4_from_x_for_10_sprites
0000:0464          ;
0000:0464          draw_kong_mouth_closed:
0000:0464          ; CODE XREF: animate_kong_and_pauline+32↑j
0000:0464 AF          xor     a
0000:0465 77          ld      (hl), a
0000:0466 23          inc     hl
0000:0467 77          ld      (hl), a
0000:0468 3A 93 63      ld      a, (barrel_deployment)
0000:046B A7          and     a
0000:046C C2 86 04      jp      nz, animate_pauline
0000:046F 21 5C 38      ld      hl, kong_normal_spr
0000:0472 CD 4E 00      call   copy_sprites_2_11_data
0000:0475 C3 50 04      jp      loc_0_450
0000:0478          ;
0000:0478          loc_0_478:
0000:0478          ; CODE XREF: animate_kong_and_pauline+59↑j
0000:0478          ; ptr sprite #2 (x coord)
0000:0478 21 08 69      ld      hl, soft_sprite_ram+8
0000:047B 0E 44      ld      c, 44h ; 'D'
0000:047D 0F          rrca
0000:047E D2 85 04      jp      nc, loc_0_485
0000:0481 3A B7 63      ld      a, (unk_0_63B7)
0000:0484 4F          ld      c, a
0000:0485          loc_0_485:
0000:0485          ; CODE XREF: animate_kong_and_pauline+83↑j
0000:0485          ; add C to y coord of 10 sprites
0000:0485 FF          rst      38h
0000:0486          animate_pauline:
0000:0486          ; CODE XREF: animate_kong_and_pauline+23↑j
0000:0486          ; animate_kong_and_pauline+39↑j ...
0000:0486          ld      a, (kong_thrash_tmr)
0000:0489 4F          ld      c, a
0000:048A 11 20 00      ld      de, 20h ; ' '

```



```

0000:048D 3A 27 62      ld      a, (level_type)
0000:0490 FE 04          cp      4
0000:0492 CA BE 04      jp      z, display_help_rivets_level
0000:0495 79             ld      a, c
0000:0496 A7             and     a
0000:0497 CA A1 04      jp      z, wipe_help
0000:049A 3E EF          ld      a, 0EFh ; ''
0000:049C CB 71          bit     6, c
0000:049E C2 A3 04      jp      nz, display_or_wipe_help
0000:04A1
0000:04A1      wipe_help:
0000:04A1 3E 10      ld      a, 10h
0000:04A3
0000:04A3      display_or_wipe_help:
0000:04A3 21 C4 75      ld      hl, VRAM_start+1C4h
0000:04A6 CD 14 05      call    display_3_tiles_HL
0000:04A9 3A 05 69      ld      a, (soft_sprite_ram+5)
0000:04AC
0000:04AC      make_pauline_run:
0000:04AC 32 05 69      ld      (soft_sprite_ram+5), a
0000:04AC CB 71          bit     6, c
0000:04B1 C8          ret     z
0000:04B2 47          ld      b, a
0000:04B3 79          ld      a, c
0000:04B4 E6 07      and     7
0000:04B6 C0          ret     nz
0000:04B7 78          ld      a, b
0000:04B8 EE 03      xor     3
0000:04BA 32 05 69      ld      (soft_sprite_ram+5), a
0000:04BD C9          ret
0000:04BE
0000:04BE
0000:04BE      display_help_rivets_level:
0000:04BE 3E 10      ld      a, 10h
0000:04C0 21 23 76      ld      hl, VRAM_start+223h
0000:04C3 CD 14 05      call    display_3_tiles_HL
0000:04C6 21 83 75      ld      hl, VRAM_start+183h
0000:04C9 CD 14 05      call    display_3_tiles_HL
0000:04CC CB 71          bit     6, c
0000:04CE CA 09 05      jp      z, loc_0_509
0000:04D1 3A 03 62      ld      a, (mario_y_coord)
0000:04D4 FE 80          cp      80h ; 'Ç'
0000:04D6 D2 F1 04      jp      nc, display_help_right
0000:04D9 3E DF          ld      a, 0DFh ; '■'
0000:04DB 21 23 76      ld      hl, VRAM_start+223h
0000:04DE CD 14 05      call    display_3_tiles_HL
0000:04E1
0000:04E1      display_pauline_left:
0000:04E1 3A 01 69      ld      a, (soft_sprite_ram+1)
0000:04E4 F6 80          or      80h ; 'Ç'
0000:04E6 32 01 69      ld      (soft_sprite_ram+1), a
0000:04E9 3A 05 69      ld      a, (soft_sprite_ram+5)
0000:04EC F6 80          or      80h ; 'Ç'
0000:04EE C3 AC 04      jp      make_pauline_run
0000:04F1
0000:04F1
0000:04F1      display_help_right:
0000:04F1 3E EF          ld      a, 0EFh ; ''
0000:04F3 21 83 75      ld      hl, VRAM_start+183h
0000:04F6 CD 14 05      call    display_3_tiles_HL
0000:04F9
0000:04F9      display_pauline_right:
0000:04F9 3A 01 69      ld      a, (soft_sprite_ram+1)
0000:04FC E6 7F          and     7Fh ; ''
0000:04FE 32 01 69      ld      (soft_sprite_ram+1), a
0000:0501 3A 05 69      ld      a, (soft_sprite_ram+5)
0000:0504 E6 7F          and     7Fh ; ''
0000:0506 C3 AC 04      jp      make_pauline_run
0000:0509
0000:0509
0000:0509      loc_0_509:
0000:0509 3A 03 62      ld      a, (mario_y_coord)
0000:050C FE 80          cp      80h ; 'Ç'
0000:050E D2 F9 04      jp      nc, display_pauline_right
0000:0511 C3 E1 04      jp      display_pauline_left
0000:0511
0000:0511      ; End of function animate_kong_and_pauline
0000:0511
0000:0514      ; ██████████ SUBROUTINE ██████████
0000:0514
0000:0514      display_3_tiles_HL:
0000:0514 06 03      ld      b, 3
0000:0516
0000:0516      loc_0_516:
0000:0516 77          ld      (hl), a
0000:0517 19          add     hl, de
0000:0518 3D          dec     a
0000:0519 10 FB      djnz    loc_0_516
0000:051B C9          ret
0000:051B
0000:051B      ; End of function display_3_tiles_HL
0000:051B
0000:051C      ; ██████████ SUBROUTINE ██████████
0000:051C
0000:051C
0000:051C      add_bonus_and_update_high_score:
0000:051C 4F          ; CODE XREF: 0000:0698|p
0000:051C          ; 0000:06A5|j
0000:051C          ; DATA XREF: ...
0000:051C      ld      c, a
0000:051D CF      rst     8
0000:051E CD 5F 05      call    current_player_score_DE
0000:0521 79          ld      a, c
0000:0522 81          add     a, c
0000:0523 81          add     a, c
0000:0524 4F          ld      c, a
0000:0525 21 29 35      ld      hl, Bonus_Points_Tbl
0000:0528 06 00          ld      b, 0
0000:052A 09          add     hl, bc
0000:052B A7          and     a
0000:052C 06 03      ld      b, 3
0000:052E
0000:052E      loc_0_52E:
0000:052E 1A          ld      a, (de)
0000:052E          ; CODE XREF: add_bonus_and_update_high_score+18|j
0000:052E          ; get score BCD pair

```

```

0000:052F 8E      adc     a, (hl)                ; add bonus BCD pair
0000:0530 27      daa                     ; adjust for BCD
0000:0531 12      ld      (de), a              ; update score BCD pair
0000:0532 13      inc     de
0000:0533 23      inc     hl                  ; next byte
0000:0534 10 F8    djnz    loc_0_52E           ; loop through score
0000:0536 D5      push    de
0000:0537 1B      dec     de                  ; ptr score
0000:0538 3A 0D 60  ld      a, (current_player_D)
0000:053B CD 6B 05  call    display_player_A_score
0000:053E D1      pop     de
0000:053F 1B      dec     de
0000:0540 21 BA 60  ld      hl, high_score+2      ; MSB
0000:0543 06 03    ld      b, 3                 ; 3 bytes to compare
0000:0545
0000:0545 loc_0_545:                  ; CODE XREF: add_bonus_and_update_high_score+31|j
0000:0545 1A      ld      a, (de)              ; get byte from score
0000:0546 BE      cp      (hl)              ; less than high score?
0000:0547 D8      ret     c                   ; yes, return
0000:0548 C2 50 05  jp      nz, new_high_score    ; greater, we have a high score
0000:054B 1B      dec     de
0000:054C 2B      dec     hl
0000:054D 10 F6    djnz    loc_0_545           ; same, check next byte
0000:054F C9      ret
0000:0550
0000:0550
0000:0550 new_high_score:              ; CODE XREF: add_bonus_and_update_high_score+2C|j
0000:0550 CD 5F 05  call    current_player_score_DE
0000:0553 21 B8 60  ld      hl, high_score
0000:0556
0000:0556 update_high_score:          ; CODE XREF: add_bonus_and_update_high_score+3E|j
0000:0556 1A      ld      a, (de)              ; get score byte
0000:0557 77      ld      (hl), a              ; copy to high score
0000:0558 13      inc     de
0000:0559 23      inc     hl                  ; next location
0000:055A 10 FA    djnz    update_high_score    ; loop through 3 bytes
0000:055C C3 DA 05  jp      display_high_score
0000:055C ; End of function add_bonus_and_update_high_score
0000:055F
0000:055F ; SUBROUTINE
0000:055F
0000:055F current_player_score_DE:    ; CODE XREF: add_bonus_and_update_high_score+2|p
0000:055F 11 B2 60  ld      de, p1_score          ; add_bonus_and_update_high_score+34|p
0000:0562 3A 0D 60  ld      a, (current_player_D)
0000:0565 A7      and     a
0000:0566 C8      ret     z                 ; player one?
0000:0567 11 B5 60  ld      de, p2_score          ; yes, return
0000:056A C9      ret
0000:056A ; End of function current_player_score_DE
0000:056A
0000:056B ; SUBROUTINE
0000:056B
0000:056B
0000:056B display_player_A_score:      ; CODE XREF: add_bonus_and_update_high_score+1F|p
0000:056B DD 21 81 77  ld      ix, VRAM_start+381h  ; display_score_or_high_score+11|j
0000:056B A7      and     a
0000:0570 28 0A    jr      z, display_score_HL
0000:0572 DD 21 21 75  ld      ix, VRAM_start+121h
0000:0576 18 04    jr      display_score_HL
0000:0578
0000:0578 display_score_at_hs_location: ; CODE XREF: display_score_or_high_score+17|j
0000:0578 DD 21 41 76  ld      ix, VRAM_start+241h  ; screen position for score
0000:057C
0000:057C display_score_HL:           ; CODE XREF: display_player_A_score+5|j
0000:057C EB      ex      de, hl          ; display_player_A_score+B|j ...
0000:057C
0000:057D 11 E0 FF  ld      de, 0FFE0h        ; column address delta
0000:0580 01 04 03  ld      bc, 304h            ; 3=6 digits
0000:0583
0000:0583 display_B_bcd_digit_pairs:  ; CODE XREF: display_player_A_score+25|j
0000:0583 7E      ld      a, (hl)              ; display_credits+11|j
0000:0583 0F      rrca                     ; get bcd digit pair
0000:0584 0F      rrca
0000:0585 0F      rrca
0000:0586 0F      rrca
0000:0587 0F      rrca                      ; shift high nibble
0000:0588 CD 93 05  call    display_score_digit
0000:058B 7E      ld      a, (hl)              ; low nibble
0000:058C CD 93 05  call    display_score_digit
0000:058F 2B      dec     hl                  ; next digit pair
0000:0590 10 F1    djnz    display_B_bcd_digit_pairs ; loop through 6 digits
0000:0592 C9      ret
0000:0592 ; End of function display_player_A_score
0000:0593
0000:0593 ; SUBROUTINE
0000:0593
0000:0593 display_score_digit:        ; CODE XREF: display_player_A_score+1D|p
0000:0593 E6 0F    and     0Fh                ; display_player_A_score+21|p
0000:0593 DD 77 00  ld      (ix+0), a            ; low nibble only
0000:0598 DD 19    add     ix, de            ; display digit
0000:059A C9      ret                    ; next column
0000:059A ; End of function display_score_digit
0000:059A
0000:059B ; SUBROUTINE
0000:059B
0000:059B zero_score_or_high_score:  ; CODE XREF: zero_score_or_high_score+24|p
0000:059B FE 03    cp      3                  ; DATA XREF: 0000:0309|o
0000:059B D2 BD 05  jp      nc, loc_0_5BD        ; zero all scores?
0000:059D F5      push    af                ; yes, skip
0000:05A1 21 B2 60  ld      hl, p1_score
0000:05A4 A7      and     a
0000:05A5 CA AB 05  jp      z, loc_0_5AB
0000:05A8 21 B5 60  ld      hl, p2_score
0000:05AB

```

```

0000:05AB      loc_0_5AB:                                ; CODE XREF: zero_score_or_high_score+A|j
0000:05AB FE 02      cp      2
0000:05AD C2 B3 05      jp      nz, loc_0_5B3
0000:05B0 21 B8 60      ld      hl, high_score
0000:05B3
0000:05B3      loc_0_5B3:                                ; CODE XREF: zero_score_or_high_score+12|j
0000:05B3 AF          xor      a
0000:05B4 77          ld      (hl), a
0000:05B5 23          inc     hl
0000:05B6 77          ld      (hl), a
0000:05B7 23          inc     hl
0000:05B8 77          ld      (hl), a
0000:05B9 F1          pop     af
0000:05BA C3 C6 05      jp      display_score_or_high_score
0000:05BD
0000:05BD      loc_0_5BD:                                ; CODE XREF: zero_score_or_high_score+2|j
0000:05BD 3D          dec     a                                ; zero_score_or_high_score+29|j
0000:05BD          push    af                                ; next score to zero
0000:05BE F5          call    zero_score_or_high_score
0000:05BF CD 9B 05      pop     af
0000:05C2 F1          ret     z                                ; return when done
0000:05C3 C8          jr      loc_0_5BD                        ; zero next score
0000:05C4 18 F7      ; End of function zero_score_or_high_score
0000:05C4
0000:05C6      ; SUBROUTINE
0000:05C6
0000:05C6      display_score_or_high_score:              ; CODE XREF: zero_score_or_high_score+1F|j
0000:05C6 FE 03      ; display_score_or_high_score+1C|p
0000:05C6          ; DATA XREF: ...
0000:05C6      cp      3
0000:05C8 CA E0 05      jp      z, loc_0_5E0
0000:05CB 11 B4 60      ld      de, pl_score+2
0000:05CE A7          and     a
0000:05CF CA D5 05      jp      z, loc_0_5D5
0000:05D2 11 B7 60      ld      de, p2_score+2
0000:05D5
0000:05D5      loc_0_5D5:                                ; CODE XREF: display_score_or_high_score+9|j
0000:05D5 FE 02      cp      2
0000:05D7 C2 6B 05      jp      nz, display_player_A_score
0000:05DA
0000:05DA      display_high_score:                      ; CODE XREF: add_bonus_and_update_high_score+40|j
0000:05DA 11 BA 60      ld      de, high_score+2
0000:05DD C3 78 05      jp      display_score_at_hs_location
0000:05E0
0000:05E0      loc_0_5E0:                                ; CODE XREF: display_score_or_high_score+2|j
0000:05E0 3D          dec     a                                ; display_score_or_high_score+21|j
0000:05E1 F5          push    af
0000:05E2 CD C6 05      call    display_score_or_high_score
0000:05E5 F1          pop     af
0000:05E6 C8          ret     z
0000:05E7 18 F7      jr      loc_0_5E0
0000:05E7      ; End of function display_score_or_high_score
0000:05E7
0000:05E9      ; SUBROUTINE
0000:05E9
0000:05E9      print_message_A:                        ; CODE XREF: display_credits+2|p
0000:05E9 21 4B 36      ; display_start_12P_get_selection+18|p
0000:05E9          ; DATA XREF: ...
0000:05E9      ld      hl, message_table
0000:05EC 87          add     a, a                                ; convert entry to offset
0000:05ED F5          push    af
0000:05EE E6 7F      and     7Fh ; ' '                        ; mask off 'wipe' bit
0000:05F0 5F          ld      e, a
0000:05F1 16 00      ld      d, 0                                ; DE = offset
0000:05F3 19          add     hl, de                                ; pointer to entry
0000:05F4 5E          ld      e, (hl)
0000:05F5 23          inc     hl
0000:05F6 56          ld      d, (hl)                                ; DE = entry (word)
0000:05F7 EB          ex      de, hl
0000:05F8 5E          ld      e, (hl)
0000:05F9 23          inc     hl
0000:05FA 56          ld      d, (hl)                                ; DE = screen address to print
0000:05FB 23          inc     hl                                ; HL = message text
0000:05FC 01 E0 FF      ld      bc, 0FFE0h                        ; screen column address inc value
0000:05FF EB          ex      de, hl                        ; DE = text, HL = screen address
0000:0600
0000:0600      loc_0_600:                                ; CODE XREF: print_message_A+26|j
0000:0600 1A          ld      a, (de)                                ; get message character
0000:0601
0000:0601      loc_0_601:                                ; DATA XREF: 0000:1275|o
0000:0601 FE 3F      cp      3Fh ; '?'                        ; end of message?
0000:0603 CA 26 00      jp      z, pop_hl_ret
0000:0606 77          ld      (hl), a
0000:0607 F1          pop     af
0000:0608 30 02      jr      nc, loc_0_60C
0000:060A 36 10      ld      (hl), 10h
0000:060C
0000:060C      loc_0_60C:                                ; CODE XREF: print_message_A+1F|j
0000:060C F5          push    af                                ; store original entry index
0000:060D 13          inc     de                                ; next message character
0000:060E 09          add     hl, bc
0000:060F 18 EF      jr      loc_0_600                        ; next screen location
0000:060F      ; End of function print_message_A
0000:0611
0000:0611      display_credits_if_attract_mode:          ; DATA XREF: 0000:030F|o
0000:0611 3A 07 60      ld      a, (attract_mode_flag)
0000:0614 0F          rrca
0000:0615 D0          ret     nc                                ; in attract mode?
0000:0616          ; no, return
0000:0616      ; SUBROUTINE
0000:0616
0000:0616      display_credits:                        ; CODE XREF: display_start_12P_get_selection+1B|p
0000:0616 3E 05      ld      a, 5                                ; 0000:141E|p ...
0000:0616          ; "credit"
0000:0618 CD E9 05      call    print_message_A

```

```

0000:061B 21 01 60      ld      hl, no_of_credits
0000:061E 11 E0 FF      ld      de, 0FFE0h          ; column address delta
0000:0621 DD 21 BF 74      ld      ix, VRAM_start+0BFh ; screen position of credits
0000:0625 06 01      ld      b, 1                ; 1=2 digits
0000:0627 C3 83 05      jp      display_B_bcd_digit_pairs
0000:0627      ; End of function display_credits
0000:0627
0000:062A
0000:062A      ;
0000:062A      ; -----
0000:062A      update_bonus_timer:
0000:062A A7      and      a                  ; DATA XREF: 0000:0311|o
0000:062B CA 91 06      jp      z, loc_0_691
0000:062E 3A 8C 63      ld      a, (bonus_timer)
0000:0631 A7      and      a                  ; zero?
0000:0632 C2 A8 06      jp      nz, loc_0_6A8       ; no, skip
0000:0635 3A B8 63      ld      a, (unk_0_63B8)
0000:0638 A7      and      a
0000:0639 C0      ret      nz
0000:063A 3A B0 62      ld      a, (unk_0_62B0)
0000:063D 01 0A 00      ld      bc, 0Ah
0000:0640
0000:0640      loc_0_640:
0000:0640      inc      b                  ; CODE XREF: 0000:0642|j
0000:0641 91      sub      c
0000:0642 C2 40 06      jp      nz, loc_0_640
0000:0645 78      ld      a, b
0000:0646 07      rlca
0000:0647 07      rlca
0000:0648 07      rlca
0000:0649 07      rlca
0000:064A 32 8C 63      ld      (bonus_timer), a    ; set initial bonus timer value
0000:064D 21 4A 38      ld      hl, bonus_graphic_tiles
0000:0650 11 65 74      ld      de, VRAM_start+65h ; screen position for bonus
0000:0653 3E 06      ld      a, 6                ; 6 columns of tiles to display
0000:0655
0000:0655      loc_0_655:
0000:0655 DD 21 1D 00      ld      ix, 1Dh             ; CODE XREF: 0000:0664|j
0000:0659 01 03 00      ld      bc, 3                ; column inc
0000:065C ED B0      ldir                     ; 3 tiles to display
0000:065E DD 19      add      ix, de              ; display bonus tiles
0000:0660 DD E5      push     ix                 ; next column
0000:0662 D1      pop      de
0000:0663 3D      dec      a                  ; screen position
0000:0664 C2 55 06      jp      nz, loc_0_655       ; done?
0000:0667 3A 8C 63      ld      a, (bonus_timer)    ; no, loop
0000:066A
0000:066A      loc_0_66A:
0000:066A      ld      c, a                ; CODE XREF: 0000:06B5|j
0000:066B E6 0F      and      0Fh
0000:066D 47      ld      b, a
0000:066E 79      ld      a, c                ; B=low nibble
0000:066F 0F      rrca
0000:0670 0F      rrca
0000:0671 0F      rrca
0000:0672 0F      rrca
0000:0673 E6 0F      and      0Fh                ; C=high nibble
0000:0675 C2 89 06      jp      nz, display_bonus_digits ; skip if more than 9s left
0000:0678 3E 03      ld      a, 3
0000:067A 32 89 60      ld      (unk_0_6089), a
0000:067D 3E 70      ld      a, 70h ; 'p'        ; purple '0'
0000:067F 32 86 74      ld      (VRAM_start+86h), a ; '0'
0000:0682 32 A6 74      ld      (VRAM_start+0A6h), a ; '0'
0000:0685 80      add      a, b                ; 2nd digit to 'ascii'
0000:0686 47      ld      b, a                ; store
0000:0687 3E 10      ld      a, 10h              ; <space>
0000:0689
0000:0689      display_bonus_digits:
0000:0689      ld      (VRAM_start+0E6h), a ; CODE XREF: 0000:0675|j
0000:068C 78      ld      a, b                ; display 1st digit
0000:068D 32 C6 74      ld      (VRAM_start+0C6h), a ; restore 2nd digit
0000:0690 C9      ret                          ; display 2nd digit
0000:0691
0000:0691      ;
0000:0691      loc_0_691:
0000:0691      ld      a, (bonus_timer)    ; CODE XREF: 0000:062B|j
0000:0694 47      ld      b, a
0000:0695 E6 0F      and      0Fh
0000:0697 C5      push     bc
0000:0698 CD 1C 05      call    add_bonus_and_update_high_score
0000:069B C1      pop      bc
0000:069C 78      ld      a, b
0000:069D 0F      rrca
0000:069E 0F      rrca
0000:069F 0F      rrca
0000:06A0 0F      rrca
0000:06A1 E6 0F      and      0Fh
0000:06A3 C6 0A      add      a, 0Ah
0000:06A5 C3 1C 05      jp      add_bonus_and_update_high_score
0000:06A8
0000:06A8      ;
0000:06A8      loc_0_6A8:
0000:06A8      sub      1                  ; CODE XREF: 0000:0632|j
0000:06AA 20 05      jr      nz, loc_0_6B1
0000:06AC 21 B8 63      ld      hl, unk_0_63B8
0000:06AF 36 01      ld      (hl), 1
0000:06B1
0000:06B1      loc_0_6B1:
0000:06B1      daa                      ; CODE XREF: 0000:06AA|j
0000:06B2 27      ld      (bonus_timer), a
0000:06B5 C3 6A 06      jp      loc_0_66A
0000:06B8
0000:06B8      ;
0000:06B8      SUBROUTINE
0000:06B8
0000:06B8      display_lives_and_level:
0000:06B8      ld      c, a                ; CODE XREF: 0000:01DC|p
0000:06B8 4F      rst      8                  ; check_and_award_bonus+2C|j
0000:06B8      ; DATA XREF: ...
0000:06B8      ld      8                  ; store number of lives
0000:06B9 CF      ld      b, 6                ; return if attract mode
0000:06BA 06 06      ld      b, 6                ; max icons
0000:06BC 11 E0 FF      ld      de, 0FFE0h          ; column delta
0000:06BF 21 83 77      ld      hl, VRAM_start+383h
0000:06C2
0000:06C2      loc_0_6C2:
0000:06C2      ld      (hl), 10h           ; CODE XREF: display_lives_and_level+D|j
0000:06C4 19      add      hl, de              ; <space>
0000:06C4      ; next column

```

```

0000:06C5 10 FB      djnz    loc_0_6C2                ; wipe 6 icons
0000:06C7 3A 28 62      ld      a, (lives_left)
0000:06CA 91            sub      c                ; HUH???
0000:06CB CA D7 06      jp      z, loc_0_6D7
0000:06CE 47            ld      b, a                ; number of lives
0000:06CF 21 83 77      ld      hl, VRAM_start+383h ; screen location
0000:06D2
0000:06D2      loc_0_6D2:                ; CODE XREF: display_lives_and_level+1D|j
0000:06D2 36 FF      ld      (hl), 0FFh          ; mario icon
0000:06D4 19      add      hl, de            ; next screen location
0000:06D5 10 FB      djnz    loc_0_6D2          ; loop for no. of lives
0000:06D7      loc_0_6D7:                ; CODE XREF: display_lives_and_level+13|j
0000:06D7 21 03 75      ld      hl, VRAM_start+103h
0000:06DA 36 1C      ld      (hl), 1Ch          ; 'L'
0000:06DC 21 E3 74      ld      hl, VRAM_start+0E3h
0000:06DF 36 34      ld      (hl), 34h ; '4'
0000:06E1 3A 29 62      ld      a, (level)
0000:06E4 FE 64      cp      100                ; too high?
0000:06E6 38 05      jr      c, loc_0_6ED        ; no, skip
0000:06E8 3E 63      ld      a, 99              ; max out at 99
0000:06EA 32 29 62      ld      (level), a          ; adjust
0000:06ED
0000:06ED      loc_0_6ED:                ; CODE XREF: display_lives_and_level+2E|j
0000:06ED 01 0A FF      ld      bc, 0FF0Ah
0000:06F0      loc_0_6F0:                ; CODE XREF: display_lives_and_level+3A|j
0000:06F0 04      inc      b
0000:06F1 91      sub      c
0000:06F2 D2 F0 06      jp      nc, loc_0_6F0
0000:06F5 81      add      a, c                ; level tens digit
0000:06F6 32 A3 74      ld      (VRAM_start+0A3h), a
0000:06F9 78      ld      a, b                ; level units digit
0000:06FA 32 C3 74      ld      (VRAM_start+0C3h), a
0000:06FD C9      ret
0000:06FD      ; End of function display_lives_and_level
0000:06FD
0000:06FE
0000:06FE
0000:06FE      vector_on_ingame_sequencer: ; DATA XREF: 0000:00D0|o
0000:06FE 3A 0A 60      ld      a, (main_sequencer)
0000:0701 EF      rst      28h                ; go!
0000:0701
; -----
0000:0702 86 09      .dw cls_and_set_screen_flip ; Jump table
0000:0704 AB 09      .dw init_ingame_data_p1
0000:0706 D6 09      .dw loc_0_9D6
0000:0708 FE 09      .dw init_ingame_data_p2
0000:070A 1B 0A      .dw display_p2_screen?
0000:070C 37 0A      .dw display_p1_screen?
0000:070E 63 0A      .dw wait_cls_____
0000:0710 76 0A      .dw do_intro_sequence
0000:0712 DA 0B      .dw draw_how_high_can_you_get
0000:0714 00 00      .dw 0
0000:0716 91 0C      .dw wait_init_and_draw_level
0000:0718 3C 12      .dw init_mario
0000:071A 7A 19      .dw gameplay
0000:071C 7C 12      .dw died_in_gameplay
0000:071E F2 12      .dw copy_current_lvl_data_to_P1
0000:0720 44 13      .dw copy_current_lvl_data_to_P2
0000:0722 8F 13      .dw p1_game_over
0000:0724 A1 13      .dw p2_game_over
0000:0726 AA 13      .dw set_flip_and_current_P2
0000:0728 BB 13      .dw set_flip_and_current_P1
0000:072A 1E 14      .dw draw_name_registered
0000:072C 86 14      .dw do_initials_entry
0000:072E 15 16      .dw mario_pauline_reunion
0000:0730 6B 19      .dw loc_0_196B
0000:0732 00 00      .dw 0
0000:0734 00 00      .dw 0
0000:0736 00 00      .dw 0
0000:0738 00 00      .dw 0
0000:073A 00 00      .dw 0
0000:073C
; -----
0000:073C      chk_credits_and_vector_on_attrac: ; DATA XREF: 0000:00CC|o
0000:073C      ld      hl, main_sequencer
0000:073F 3A 01 60      ld      a, (no_of_credits)
0000:0742 A7      and      a                ; any credits?
0000:0743 C2 5C 07      jp      nz, inc_nmi_sequencer ; yes, skip
0000:0746 7E      ld      a, (hl)
0000:0747 EF      rst      28h                ; go!
0000:0747
; -----
0000:0748 79 07      .dw insert_coin_screen      ; Jump Table (attract sequencer)
0000:074A 63 07      .dw init_attract_mode_and_draw_level
0000:074C 3C 12      .dw init_mario
0000:074E 77 19      .dw attract_mode_gameplay
0000:0750 7C 12      .dw died_in_gameplay
0000:0752 C3 07      .dw cls_and_next_sequence
0000:0754 CB 07      .dw title_screen_flash
0000:0756 4B 08      .dw title_screen_no_flash
0000:0758 00 00      .dw 0
0000:075A 00 00      .dw 0
0000:075C
; -----
0000:075C      inc_nmi_sequencer:         ; CODE XREF: 0000:0743|j
0000:075C      ld      (hl), 0            ; reset game sequencer
0000:075E 21 05 60      ld      hl, nmi_sequencer
0000:0761 34      inc      (hl)              ; inc nmi_sequencer
0000:0762 C9      ret
0000:0763
; -----
0000:0763      init_attract_mode_and_draw_level: ; DATA XREF: 0000:074A|o
0000:0763      rst      20h                ; wait for 16-bit countdown
0000:0764 AF      xor      a
0000:0765 32 92 63      ld      (unk_0_6392), a
0000:0768 32 A0 63      ld      (unk_0_63A0), a
0000:076B 3E 01      ld      a, 1
0000:076D 32 27 62      ld      (level_type), a
0000:0770 32 29 62      ld      (level), a
0000:0773 32 28 62      ld      (lives_left), a
0000:0776 C3 92 0C      jp      init_and_draw_level
0000:0779
; -----
0000:0779      insert_coin_screen:        ; DATA XREF: 0000:0748|o
0000:0779      ld      hl, palette_bank
0000:077C 36 00      ld      (hl), 0

```

```

0000:077E 23          inc    hl
0000:077F 36 00          ld     (hl), 0
0000:0781 11 1B 03          ld     de, 31Bh
0000:0784 CD 9F 30          call  sub_0_309F
0000:0787 1C          inc    e
0000:0788 CD 9F 30          call  sub_0_309F
0000:078B CD 65 09          call  sub_0_965
0000:078E 21 09 60          ld     hl, eight_bit_countdown
0000:0791 36 02          ld     (hl), 2
0000:0793 23          inc    hl
0000:0794 34          inc    (hl)
0000:0795 CD 74 08          call  clear_visible_area_and_sprites
0000:0798 CD 53 0A          call  display_1UP
0000:079B 3A 0F 60          ld     a, (two_players)
0000:079E FE 01          cp     1
0000:07A0 CC EE 09          call  z, display_2UP
0000:07A3 ED 5B 22 60          ld     de, (coinage)
0000:07A7 21 6C 75          ld     hl, VRAM_start+16Ch
0000:07AA CD AD 07          call  display_coinage
0000:07AD          display_coinage:
0000:07AD 73          ld     (hl), e
0000:07AE 23          inc    hl
0000:07AF 23          inc    hl
0000:07B0 72          ld     (hl), d
0000:07B1 7A          ld     a, d
0000:07B2 D6 0A          sub    0Ah
0000:07B4 C2 BC 07          jp     nz, loc_0_7BC
0000:07B7 77          ld     (hl), a
0000:07B8 3C          inc    a
0000:07B9 32 8E 75          ld     (VRAM_start+18Eh), a
0000:07BC          loc_0_7BC:
0000:07BC 11 01 02          ld     de, 201h
0000:07BF 21 8C 76          ld     hl, VRAM_start+28Ch
0000:07C2 C9          ret
0000:07C3          ;
0000:07C3          cls_and_next_sequence:
0000:07C3 CD 74 08          call  clear_visible_area_and_sprites
0000:07C6 21 0A 60          ld     hl, main_sequencer
0000:07C9 34          inc    (hl)
0000:07CA C9          ret
0000:07CB          ;
0000:07CB          title_screen_flash:
0000:07CB 3A 8A 63          ld     a, (unk_0_638A)
0000:07CE FE 00          cp     0
0000:07D0 C2 2D 08          jp     nz, loc_0_82D
0000:07D3 3E 60          ld     a, 60h
0000:07D5 32 8A 63          ld     (unk_0_638A), a
0000:07D8 0E 5F          ld     c, 5Fh
0000:07DA          loc_0_7DA:
0000:07DA FE 00          cp     0
0000:07DC CA 3B 08          jp     z, loc_0_83B
0000:07DF 21 86 7D          ld     hl, palette_bank
0000:07E2 36 00          ld     (hl), 0
0000:07E4 79          ld     a, c
0000:07E5 CB 07          rlc    a
0000:07E7 30 02          jr     nc, loc_0_7EB
0000:07E9 36 01          ld     (hl), 1
0000:07EB          loc_0_7EB:
0000:07EB 23          inc    hl
0000:07EC 36 00          ld     (hl), 0
0000:07EE CB 07          rlc    a
0000:07F0 30 02          jr     nc, loc_0_7F4
0000:07F2 36 01          ld     (hl), 1
0000:07F4          loc_0_7F4:
0000:07F4 32 8B 63          ld     (unk_0_638B), a
0000:07F7 21 08 3D          ld     hl, title_screen
0000:07FA          display_donkey_kong_title:
0000:07FA 3E B0          ld     a, 0B0h
0000:07FC 46          ld     b, (hl)
0000:07FD 23          inc    hl
0000:07FE 5E          ld     e, (hl)
0000:07FF 23          inc    hl
0000:0800 56          ld     d, (hl)
0000:0801          loc_0_801:
0000:0801 12          ld     (de), a
0000:0802 13          inc    de
0000:0803 10 FC          djnz  loc_0_801
0000:0805 23          inc    hl
0000:0806 7E          ld     a, (hl)
0000:0807 FE 00          cp     0
0000:0809 C2 FA 07          jp     nz, display_donkey_kong_title
0000:080C 11 1E 03          ld     de, 31Eh
0000:080F CD 9F 30          call  sub_0_309F
0000:0812 13          inc    de
0000:0813 CD 9F 30          call  sub_0_309F
0000:0816 21 CF 39          ld     hl, dk_thrash_right_spr
0000:0819 CD 4E 00          call  copy_sprites_2_11_data
0000:081C CD 34 3F          call  sub_0_3F24
0000:081F 00          nop
0000:0820 21 08 69          ld     hl, soft_sprite_ram+8
0000:0823 0E 44          ld     c, 68
0000:0825 FF          rst    38h
0000:0826 21 0B 69          ld     hl, soft_sprite_ram+0Bh
0000:0829 0E 78          ld     c, 120
0000:082B FF          rst    38h
0000:082C C9          ret
0000:082D          ;
0000:082D          loc_0_82D:
0000:082D 3A 8B 63          ld     a, (unk_0_638B)
0000:0830 4F          ld     c, a
0000:0831 3A 8A 63          ld     a, (unk_0_638A)
0000:0834 3D          dec    a
0000:0835 32 8A 63          ld     (unk_0_638A), a
0000:0838 C3 DA 07          jp     loc_0_7DA
0000:083B          ;
0000:083B

```

```

0000:083B      loc_0_83B:      ld      hl, eight_bit_countdown      ; CODE XREF: 0000:07DC↑j
0000:083B 21 09 60      ld      (hl), 2
0000:083E 36 02      inc     hl                          ; game_sequencer
0000:0840 23      inc     (hl)
0000:0841 34      ld      hl, unk_0_638A
0000:0842 21 8A 63      ld      (hl), 0
0000:0845 36 00      inc     hl
0000:0847 23      ld      (hl), 0
0000:0848 36 00      ret
0000:084A C9      ;
0000:084B
0000:084B
0000:084B      title_screen_no_flash:      ; DATA XREF: 0000:0756↑o
0000:084B E7      rst      20h      ; wait for 16-bit countdown
0000:084C 21 0A 60      ld      hl, main_sequencer
0000:084F 36 00      ld      (hl), 0      ; reset game sequencer
0000:0851 C9      ret
0000:0852
0000:0852      ; ██████████ S U B R O U T I N E ██████████
0000:0852
0000:0852      clear_tiles_and_sprites:      ; CODE XREF: 0000:0986↑p
0000:0852 21 00 74      ; 0000:196B↑p
0000:0852      ld      hl, VRAM_start
0000:0855 0E 04      ld      c, 4      ; 4x256 bytes to clear
0000:0857
0000:0857      loc_0_857:      ; CODE XREF: clear_tiles_and_sprites+E↑j
0000:0857 06 00      ld      b, 0      ; 256 bytes to clear
0000:0859 3E 10      ld      a, 10h      ; space character
0000:085B
0000:085B      loc_0_85B:      ; CODE XREF: clear_tiles_and_sprites+B↑j
0000:085B 77      ; display space
0000:085C 23      ld      (hl), a
0000:085D 10 FC      djnz    loc_0_85B      ; clear 256 bytes
0000:085F 0D      dec     c
0000:0860 C2 57 08      jp      nz, loc_0_857      ; do 1024 bytes
0000:0863 21 00 69      ld      hl, soft_sprite_ram
0000:0866 0E 02      ld      c, 2      ; 2x192 bytes to clear
0000:0868
0000:0868      loc_0_868:      ; CODE XREF: clear_tiles_and_sprites+1E↑j
0000:0868 06 C0      ld      b, 192      ; 192 bytes to clear
0000:086A AF      xor     a
0000:086B
0000:086B      loc_0_86B:      ; CODE XREF: clear_tiles_and_sprites+1B↑j
0000:086B 77      ; clear soft sprite ram byte
0000:086C 23      ld      (hl), a      ; next address
0000:086D 10 FC      djnz    loc_0_86B      ; clear 192 bytes
0000:086F 0D      dec     c
0000:0870 C2 68 08      jp      nz, loc_0_868      ; clear 384 bytes
0000:0873 C9      ret
0000:0873      ; End of function clear_tiles_and_sprites
0000:0873
0000:0873      ; ██████████ S U B R O U T I N E ██████████
0000:0873
0000:0873      clear_visible_area_and_sprites:      ; CODE XREF: 0000:01C3↑p
0000:0873 21 04 74      ; 0000:0795↑p ...
0000:0873      ld      hl, VRAM_start+4
0000:0877 0E 20      ld      c, 32      ; 32 columns
0000:0879
0000:0879      loc_0_879:      ; CODE XREF: clear_visible_area_and_sprites+12↑j
0000:0879 06 1C      ld      b, 28      ; 28 rows
0000:087B 3E 10      ld      a, 10h      ; <space>
0000:087D 11 04 00      ld      de, 4      ; bottm-to-top next column increment
0000:0880
0000:0880      loc_0_880:      ; CODE XREF: clear_visible_area_and_sprites+E↑j
0000:0880 77      ; display space character
0000:0881 23      ld      (hl), a      ; next line
0000:0882 10 FC      djnz    loc_0_880      ; loop screen height
0000:0884 19      add     hl, de      ; next column
0000:0885 0D      dec     c      ; done all columns?
0000:0886 C2 79 08      jp      nz, loc_0_879      ; no, loop
0000:0889 21 22 75      ld      hl, VRAM_start+122h
0000:088C 11 20 00      ld      de, 32
0000:088F 0E 02      ld      c, 2
0000:0891 3E 10      ld      a, 10h      ; <space>
0000:0893
0000:0893      loc_0_893:      ; CODE XREF: clear_visible_area_and_sprites+29↑j
0000:0893 06 0E      ld      b, 14      ; 14 columns
0000:0895
0000:0895      loc_0_895:      ; CODE XREF: clear_visible_area_and_sprites+23↑j
0000:0895 77      ; display space character
0000:0896 19      add     hl, de      ; next column
0000:0897 10 FC      djnz    loc_0_895      ; loop for 14 columns
0000:0899 21 23 75      ld      hl, VRAM_start+123h
0000:089C 0D      dec     c
0000:089D C2 93 08      jp      nz, loc_0_893      ; repeat at new location
0000:08A0 21 00 69      ld      hl, soft_sprite_ram
0000:08A3 06 00      ld      b, 0      ; 256 bytes to clear
0000:08A5 3E 00      ld      a, 0      ; clear to 0x00
0000:08A7
0000:08A7      loc_0_8A7:      ; CODE XREF: clear_visible_area_and_sprites+35↑j
0000:08A7 77      ; clear soft sprite ram byte
0000:08A8 23      ld      (hl), a      ; next location
0000:08A9 10 FC      djnz    loc_0_8A7      ; do 256 bytes
0000:08AB 06 80      ld      b, 128      ; 128 bytes to clear
0000:08AD
0000:08AD      loc_0_8AD:      ; CODE XREF: clear_visible_area_and_sprites+3B↑j
0000:08AD 77      ; clear soft sprite ram byte
0000:08AE 23      ld      (hl), a      ; next location
0000:08AF 10 FC      djnz    loc_0_8AD      ; clear 128 bytes
0000:08B1 C9      ret
0000:08B1      ; End of function clear_visible_area_and_sprites
0000:08B1
0000:08B1      ;
0000:08B1
0000:08B1      vector_on_credit_sequencer:      ; DATA XREF: 0000:00CE↑o
0000:08B2 3A 0A 60      ld      a, (main_sequencer)
0000:08B5 EF      rst      28h      ; go!
0000:08B5
0000:08B5      ;
0000:08B5      .dw loc_0_8BA
0000:08B6 BA 08      ; jump table
0000:08B8 F8 08      ;
0000:08BA
0000:08BA      loc_0_8BA:      ; DATA XREF: 0000:08B6↑o

```

```

0000:08BA CD 74 08      call    clear_visible_area_and_sprites
0000:08BD AF           xor      a
0000:08BE 32 07 60      ld      (attract_mode_flag), a          ; clear attract mode flag
0000:08C1 11 0C 03      ld      de, 30Ch
0000:08C4 CD 9F 30      call    sub_0_309F
0000:08C7 21 0A 60      ld      hl, main_sequencer
0000:08CA 34           inc      (hl)
0000:08CB CD 65 09      call    sub_0_965
0000:08CE AF           xor      a
0000:08CF 21 86 7D      ld      hl, palette_bank
0000:08D2 77           ld      (hl), a
0000:08D3 2C           inc      l
0000:08D4 77           ld      (hl), a          ; palette bank 0
0000:08D5           ; ██████████ SUBROUTINE ██████████
0000:08D5
0000:08D5
0000:08D5
0000:08D5 display_start_12P_get_selection:          ; CODE XREF: 0000:08F8|p
0000:08D5      ld      b, 4
0000:08D7 1E 09      ld      e, return_if_attract_mode+1    ; "ONLY 1 PLAYER BUTTON"
0000:08D9 3A 01 60      ld      a, (no_of_credits)
0000:08DC FE 01      cp      1
0000:08DE CA E4 08      jp      z, loc_0_8E4
0000:08E1 06 0C      ld      b, 0Ch          ; mask for START1/START2
0000:08E3 1C           inc      e          ; "1 or 2 PLAYERS"
0000:08E4
0000:08E4 loc_0_8E4:          ; CODE XREF: display_start_12P_get_selection+9|j
0000:08E4      ld      a, (gen_purpose_timer)
0000:08E7 E6 07      and     7
0000:08E9 C2 F3 08      jp      nz, loc_0_8F3
0000:08EC 7B           ld      a, e          ; message 9/10
0000:08ED CD E9 05      call    print_message_A          ; display
0000:08F0 CD 16 06      call    display_credits
0000:08F3
0000:08F3 loc_0_8F3:          ; CODE XREF: display_start_12P_get_selection+14|j
0000:08F3      ld      a, (in2_snd_latch)
0000:08F6 A0           and     b
0000:08F7 C9           ret          ; read IN2
0000:08F7           ; End of function display_start_12P_get_selection
0000:08F7
0000:08F7
0000:08F8
0000:08F8
0000:08F8 display_start_12P_init_???:          ; DATA XREF: 0000:08B8|o
0000:08F8      call    display_start_12P_get_selection
0000:08FB FE 04      cp      4          ; START1?
0000:08FD CA 06 09      jp      z, start_1_selected        ; yes, skip
0000:0900 FE 08      cp      8          ; START2?
0000:0902 CA 19 09      jp      z, start_2_selected        ; yes, skip
0000:0905 C9           ret
0000:0906
0000:0906
0000:0906
0000:0906 start_1_selected:          ; CODE XREF: 0000:08FD|j
0000:0906      call    dec_credits_and_???
0000:0909 21 48 60      ld      hl, p2_level_data
0000:090C 06 08      ld      b, 8
0000:090E AF           xor      a
0000:090F
0000:090F loc_0_90F:          ; CODE XREF: 0000:0911|j
0000:090F      ld      (hl), a
0000:0910 2C           inc      l
0000:0911 10 FC      djnz    loc_0_90F
0000:0913 21 00 00      ld      hl, 0
0000:0916 C3 38 09      jp      loc_0_938
0000:0919
0000:0919
0000:0919
0000:0919 start_2_selected:          ; CODE XREF: 0000:0902|j
0000:0919      call    dec_credits_and_???
0000:091C CD 77 09      call    dec_credits_and_???
0000:091F 11 48 60      ld      de, p2_level_data
0000:0922 3A 20 60      ld      a, (lives_per_game)
0000:0925 12           ld      (de), a
0000:0926 1C           inc      e
0000:0927 21 5E 09      ld      hl, 95Eh
0000:092A 01 07 00      ld      bc, 7
0000:092D ED B0      ldir
0000:092F 11 01 01      ld      de, 101h
0000:0932 CD 9F 30      call    sub_0_309F
0000:0935 21 00 01      ld      hl, 100h          ; players=2, current_player=1
0000:0938
0000:0938 loc_0_938:          ; CODE XREF: 0000:0916|j
0000:0938      ld      (current_player_E), hl    ; players and current player
0000:093B CD 74 08      call    clear_visible_area_and_sprites
0000:093E 11 40 60      ld      de, pl_level_data
0000:0941 3A 20 60      ld      a, (lives_per_game)
0000:0944 12           ld      (de), a
0000:0945 1C           inc      e
0000:0946 21 5E 09      ld      hl, game_init_data
0000:0949 01 07 00      ld      bc, 7          ; 7 bytes
0000:094C ED B0      ldir
0000:094E 11 00 01      ld      de, 100h
0000:0951 CD 9F 30      call    sub_0_309F
0000:0954 AF           xor      a
0000:0955 32 0A 60      ld      (main_sequencer), a
0000:0958 3E 03      ld      a, 3
0000:095A 32 05 60      ld      (nmi_sequencer), a
0000:095D C9           ret
0000:095D
0000:095D
0000:095D game_init_data: .db 1          ; DATA XREF: 0000:0946|o
0000:095E           ; Start of game level init data
0000:095F 65 3A      .dw level_seq_1
0000:0961 01 00 00 00 .db 1, 0, 0, 0
0000:0965
0000:0965
0000:0965
0000:0965
0000:0965
0000:0965
0000:0965
0000:0965
0000:0965 sub_0_965:          ; CODE XREF: 0000:078B|p
0000:0965      ; 0000:08CB|p
0000:0965      ld      de, 400h
0000:0968 CD 9F 30      call    sub_0_309F
0000:096B 11 14 03      ld      de, 314h
0000:096E 06 06      ld      b, 6
0000:0970
0000:0970 loc_0_970:          ; CODE XREF: sub_0_965+F|j
0000:0970      call    sub_0_309F
0000:0973 1C           inc      e
0000:0974 10 FA      djnz    loc_0_970

```



```

0000:0976 C9          ret
0000:0976          ; End of function sub_0_965
0000:0976
0000:0977
0000:0977          ; SUBROUTINE
0000:0977
0000:0977          dec_credits_and_???          ; CODE XREF: 0000:0906|p
0000:0977 21 01 60          ; 0000:0919|p ...
0000:0977          ld      hl, no_of_credits
0000:097A 3E 99          ld      a, 99h ; 'Ö'
0000:097C 86          add     a, (hl)
0000:097D 27          daa     ; decrement credits
0000:097E 77          ld      (hl), a
0000:097F 11 00 04        ld      de, 400h
0000:0982 CD 9F 30        call    sub_0_309F
0000:0985 C9          ret
0000:0985          ; End of function dec_credits_and_???
0000:0985
0000:0986          ;
0000:0986
0000:0986          cls_and_set_screen_flip:          ; DATA XREF: 0000:0702|o
0000:0986 CD 52 08          call    clear_tiles_and_sprites
0000:0989 CD 1C 01          call    stop_sound
0000:098C 11 82 7D          ld      de, flipscreen
0000:098F 3E 01          ld      a, 1
0000:0991 12          ld      (de), a
0000:0992 21 0A 60          ld      hl, main_sequencer
0000:0995 3A 0E 60          ld      a, (current_player_E)
0000:0998 A7          and     a
0000:0999 C2 9F 09          jp      nz, loc_0_99F
0000:099C 36 01          ld      (hl), 1
0000:099E C9          ret
0000:099F          ;
0000:099F
0000:099F          loc_0_99F:          ; CODE XREF: 0000:0999|j
0000:099F 3A 26 60          ld      a, (upright)
0000:09A2 3D          dec     a
0000:09A3 CA A8 09          jp      z, loc_0_9A8
0000:09A6 AF          xor     a
0000:09A7 12          ld      (de), a
0000:09A8          ;
0000:09A8          loc_0_9A8:          ; CODE XREF: 0000:09A3|j
0000:09A8 36 03          ld      (hl), 3
0000:09AA C9          ret
0000:09AB          ;
0000:09AB
0000:09AB          init_ingame_data_p1:          ; DATA XREF: 0000:0704|o
0000:09AB 21 40 60          ld      hl, p1_level_data
0000:09AE 11 28 62          ld      de, lives_left
0000:09B1 01 08 00          ld      bc, 8
0000:09B4 ED B0          ldir
0000:09B6 2A 2A 62          ld      hl, (seq_data)
0000:09B9 7E          ld      a, (hl)
0000:09BA 32 27 62          ld      (level_type), a
0000:09BD 3A 0F 60          ld      a, (two_players)
0000:09C0 A7          and     a
0000:09C1 21 09 60          ld      hl, eight_bit_countdown
0000:09C4 11 0A 60          ld      de, main_sequencer
0000:09C7 CA D0 09          jp      z, loc_0_9D0
0000:09CA 36 78          ld      (hl), 78h ; 'x'
0000:09CC EB          ex      de, hl
0000:09CD 36 02          ld      (hl), 2
0000:09CF C9          ret
0000:09D0          ;
0000:09D0
0000:09D0          loc_0_9D0:          ; CODE XREF: 0000:09C7|j
0000:09D0 36 01          ld      (hl), 1
0000:09D2 EB          ex      de, hl
0000:09D3 36 05          ld      (hl), 5
0000:09D5 C9          ret
0000:09D6          ;
0000:09D6
0000:09D6          loc_0_9D6:          ; DATA XREF: 0000:0706|o
0000:09D6 AF          xor     a
0000:09D7 32 86 7D          ld      (palette_bank), a
0000:09DA 32 87 7D          ld      (palette_bank+1), a
0000:09DD 11 02 03          ld      de, 302h
0000:09E0 CD 9F 30          call    sub_0_309F
0000:09E3 11 01 02          ld      de, 201h
0000:09E6 CD 9F 30          call    sub_0_309F
0000:09E9 3E 05          ld      a, 5
0000:09EB 32 0A 60          ld      (main_sequencer), a
0000:09EE          ; SUBROUTINE
0000:09EE
0000:09EE          display_2UP:          ; CODE XREF: 0000:07A0|p
0000:09EE 3E 02          ld      a, 2
0000:09F0 32 E0 74          ld      (VRAM_start+0E0h), a
0000:09F3 3E 25          ld      a, 25h ; '%'
0000:09F5 32 C0 74          ld      (VRAM_start+0C0h), a
0000:09F8 3E 20          ld      a, 20h ; ' '
0000:09FA 32 A0 74          ld      (VRAM_start+0A0h), a
0000:09FD C9          ret
0000:09FD          ; End of function display_2UP
0000:09FD
0000:09FE          ;
0000:09FE
0000:09FE          init_ingame_data_p2:          ; DATA XREF: 0000:0708|o
0000:09FE 21 48 60          ld      hl, p2_level_data
0000:0A01 11 28 62          ld      de, lives_left
0000:0A04 01 08 00          ld      bc, 8
0000:0A07 ED B0          ldir
0000:0A09 2A 2A 62          ld      hl, (seq_data)
0000:0A0C 7E          ld      a, (hl)
0000:0A0D 32 27 62          ld      (level_type), a
0000:0A10 3E 78          ld      a, 78h ; 'x'
0000:0A12 32 09 60          ld      (eight_bit_countdown), a
0000:0A15 3E 04          ld      a, 4
0000:0A17 32 0A 60          ld      (main_sequencer), a
0000:0A1A C9          ret
0000:0A1B          ;
0000:0A1B
0000:0A1B          display_p2_screen?:          ; DATA XREF: 0000:070A|o

```

```

0000:0A1B AF      xor      a
0000:0A1C 32 86 7D     ld      (palette_bank), a
0000:0A1F 32 87 7D     ld      (palette_bank+1), a          ; palette bank 0
0000:0A22 11 03 03     ld      de, 303h
0000:0A25 CD 9F 30     call   sub_0_309F
0000:0A28 11 01 02     ld      de, 201h
0000:0A2B CD 9F 30     call   sub_0_309F
0000:0A2E CD EE 09     call   display_2UP
0000:0A31 3E 05      ld      a, 5
0000:0A33 32 0A 60     ld      (main_sequencer), a
0000:0A36 C9          ret
0000:0A37          ;
0000:0A37          ;
0000:0A37          display_pl_screen?:          ; DATA XREF: 0000:070C|o
0000:0A37 11 04 03     ld      de, 304h
0000:0A3A CD 9F 30     call   sub_0_309F
0000:0A3D 11 02 02     ld      de, 202h
0000:0A40 CD 9F 30     call   sub_0_309F
0000:0A43 11 00 02     ld      de, 200h
0000:0A46 CD 9F 30     call   sub_0_309F
0000:0A49 11 00 06     ld      de, 600h
0000:0A4C CD 9F 30     call   sub_0_309F
0000:0A4F 21 0A 60     ld      hl, main_sequencer
0000:0A52 34          inc     (hl)
0000:0A53          ;
0000:0A53          ; SUBROUTINE
0000:0A53          ;
0000:0A53          display_1UP:          ; CODE XREF: 0000:01F1|p
0000:0A53 3E 01          ; 0000:0798|p
0000:0A53          ld      a, 1          ; '1'
0000:0A55 32 40 77     ld      (VRAM_start+340h), a
0000:0A58 3E 25      ld      a, 25h ; '%'          ; 'U'
0000:0A5A 32 20 77     ld      (VRAM_start+320h), a
0000:0A5D 3E 20      ld      a, 20h ; ' '          ; 'P'
0000:0A5F 32 00 77     ld      (VRAM_start+300h), a
0000:0A62 C9          ret
0000:0A62          ; End of function display_1UP
0000:0A63          ;
0000:0A63          ;
0000:0A63          wait_cls____:          ; DATA XREF: 0000:070E|o
0000:0A63 DF          rst      18h          ; wait for 8-bit countdown
0000:0A64 CD 74 08     call   clear_visible_area_and_sprites
0000:0A67 21 09 60     ld      hl, eight_bit_countdown
0000:0A6A 36 01      ld      (hl), 1
0000:0A6C 2C          inc     l          ; game_sequencer
0000:0A6D 34          inc     (hl)          ; inc
0000:0A6E 11 2C 62     ld      de, unk_0_622C
0000:0A71 1A          ld      a, (de)
0000:0A72 A7          and     a          ; already seen intro?
0000:0A73 C0          ret      nz          ; no, return
0000:0A74 34          inc     (hl)          ; skip intro sequence
0000:0A75 C9          ret
0000:0A76          ;
0000:0A76          ;
0000:0A76          do_intro_sequence:          ; DATA XREF: 0000:0710|o
0000:0A76          ld      a, (intro_sequencer)
0000:0A79 EF          rst      28h          ; go!
0000:0A79          ;
0000:0A79          .dw draw_climb_screen          ; Jump table
0000:0A7C BF 0A          .dw loc_0_ABF
0000:0A7E E8 0A          .dw loc_0_AE8
0000:0A80 69 30          .dw loc_0_3069
0000:0A82 06 0B          .dw draw_1st_girder_deformation
0000:0A84 69 30          .dw loc_0_3069
0000:0A86 68 0B          .dw draw_rest_of_deformations
0000:0A88 B3 0B          .dw growl
0000:0A8A          ;
0000:0A8A          ;
0000:0A8A          draw_climb_screen:          ; DATA XREF: 0000:0A7A|o
0000:0A8A AF      xor      a
0000:0A8B 32 86 7D     ld      (palette_bank), a
0000:0A8E 3C          inc     a
0000:0A8F 32 87 7D     ld      (palette_bank+1), a          ; palette bank 2
0000:0A92 11 0D 38     ld      de, draw_data_climb
0000:0A95 CD A7 0D     call   sub_0_DA7
0000:0A98 3E 10      ld      a, 10h
0000:0A9A 32 A3 76     ld      (VRAM_start+2A3h), a
0000:0A9D 32 63 76     ld      (VRAM_start+263h), a
0000:0AA0 3E D4      ld      a, 0D4h ; 'E'          ; half ladder, half girder
0000:0AA2 32 AA 75     ld      (VRAM_start+1AAh), a
0000:0AA5 AF      xor      a
0000:0AA6 32 AF 62     ld      (unk_0_62AF), a
0000:0AA9 21 B4 38     ld      hl, dk_intro_jump_up_data
0000:0AAC 22 C2 63     ld      (unk_0_63C2), hl
0000:0AAF 21 CB 38     ld      hl, dk_intro_jump_across_data
0000:0AB2 22 C4 63     ld      (unk_0_63C4), hl
0000:0AB5 3E 40      ld      a, 40h ; '@'
0000:0AB7 32 09 60     ld      (eight_bit_countdown), a
0000:0ABA 21 85 63     ld      hl, intro_sequencer
0000:0ABD 34          inc     (hl)
0000:0ABE C9          ret
0000:0ABF          ;
0000:0ABF          ;
0000:0ABF          loc_0_ABF:          ; DATA XREF: 0000:0A7C|o
0000:0ABF DF          rst      18h          ; wait for 8-bit countdown
0000:0AC0 21 8C 38     ld      hl, kong_climbing_spr
0000:0AC3 CD 4E 00     call   copy_sprites_2_11_data
0000:0AC6 21 08 69     ld      hl, soft_sprite_ram+8
0000:0AC9 0E 30      ld      c, 48
0000:0ACB FF          rst      38h          ; add 48 to y coord for 10 sprites
0000:0ACC 21 0B 69     ld      hl, soft_sprite_ram+0Bh
0000:0ACF 0E 99      ld      c, 153          ; sprite #2, x coord
0000:0AD1 FF          rst      38h          ; add 153 to x coord for 10 sprites
0000:0AD2 3E 1F      ld      a, 1Fh
0000:0AD4 32 8E 63     ld      (unk_0_638E), a
0000:0AD7 AF      xor      a
0000:0AD8 32 0C 69     ld      (soft_sprite_ram+0Ch), a          ; sprite #3, y coord
0000:0ADB 21 8A 60     ld      hl, unk_0_608A
0000:0ADE 36 01      ld      (hl), 1
0000:0AE0          inc     hl
0000:0AE1 36 03      ld      (hl), 3
0000:0AE3 21 85 63     ld      hl, intro_sequencer
0000:0AE6 34          inc     (hl)
0000:0AE7 C9          ret

```

```

0000:0AE8      ; -----
0000:0AE8
0000:0AE8      loc_0_AE8:                                ; DATA XREF: 0000:0A7E|o
0000:0AE8 CD 6F 30      call      sub_0_306F
0000:0AEB 3A AF 62      ld        a, (unk_0_62AF)
0000:0AEE E6 0F      and      0Fh
0000:0AF0 CC 4A 30      call      z, sub_0_304A
0000:0AF3 3A 0B 69      ld        a, (soft_sprite_ram+0Bh)      ; sprite #2, x coord
0000:0AF6 FE 5D      cp        5Dh ; ']'
0000:0AF8 D0      ret      nc
0000:0AF9 3E 20      ld        a, 20h ; ' '
0000:0AFB 32 09 60      ld        (eight_bit_countdown), a
0000:0AFE 21 85 63      ld        hl, 6385h
0000:0B01 34      inc      (hl)
0000:0B02 22 C0 63      ld        (unk_0_63C0), hl
0000:0B05 C9      ret
0000:0B06      ; -----
0000:0B06
0000:0B06      draw_1st_girder_dformation:                    ; DATA XREF: 0000:0A82|o
0000:0B06 3A 1A 60      ld        a, (gen_purpose_timer)
0000:0B09 0F      rrca
0000:0B0A D8      ret      c
0000:0B0B 2A C2 63      ld        hl, (unk_0_63C2)
0000:0B0E 7E      ld        a, (hl)
0000:0B0F FE 7F      cp        7Fh ; ' '
0000:0B11 CA 1E 0B      jp        z, loc_0_B1E
0000:0B14 23      inc      hl
0000:0B15 22 C2 63      ld        (unk_0_63C2), hl
0000:0B18 4F      ld        c, a
0000:0B19 21 0B 69      ld        hl, 690Bh
0000:0B1C FF      rst      38h
0000:0B1D C9      ret
0000:0B1E      ; -----
0000:0B1E
0000:0B1E      loc_0_B1E:                                ; CODE XREF: 0000:0B11|j
0000:0B1E 21 5C 38      ld        hl, kong_normal_spr
0000:0B21 CD 4E 00      call      copy_sprites_2_1l_data
0000:0B24 11 00 69      ld        de, soft_sprite_ram
0000:0B27 01 08 00      ld        bc, 8
0000:0B2A ED B0      ldir
0000:0B2C 21 08 69      ld        hl, soft_sprite_ram+8      ; sprite #2, y coord
0000:0B2F 0E 50      ld        c, 50h ; 'P'
0000:0B31 FF      rst      38h
0000:0B32 21 0B 69      ld        hl, soft_sprite_ram+0Bh      ; sprite #2, x coord
0000:0B35 0E FC      ld        c, 0FCh ; '3'
0000:0B37 FF      rst      38h
0000:0B38      ; -----
0000:0B38      loc_0_B38:                                ; CODE XREF: 0000:0B40|j
0000:0B38 CD 4A 30      call      sub_0_304A
0000:0B3B 3A 8E 63      ld        a, (unk_0_638E)
0000:0B3E FE 0A      cp        0Ah
0000:0B40 C2 38 0B      jp        nz, loc_0_B38
0000:0B43 3E 03      ld        a, 3      ; tmr=3
0000:0B45 32 82 60      ld        (digital_snd_tmr_thump), a
0000:0B48 11 2C 39      ld        de, draw_data_bend_girders_1
0000:0B4B CD A7 0D      call      sub_0_DA7
0000:0B4E 3E 10      ld        a, 10h
0000:0B50 32 AA 74      ld        (VRAM_start+0AAh), a
0000:0B53 32 8A 74      ld        (VRAM_start+8Ah), a
0000:0B56 3E 05      ld        a, 5
0000:0B58 32 8D 63      ld        (unk_0_638D), a
0000:0B5B 3E 20      ld        a, 20h ; ' '
0000:0B5D 32 09 60      ld        (eight_bit_countdown), a
0000:0B60 21 85 63      ld        hl, 6385h
0000:0B63 34      inc      (hl)
0000:0B64 22 C0 63      ld        (unk_0_63C0), hl
0000:0B67 C9      ret
0000:0B68      ; -----
0000:0B68
0000:0B68      draw_rest_of_deformations:                    ; DATA XREF: 0000:0A86|o
0000:0B68 3A 1A 60      ld        a, (gen_purpose_timer)
0000:0B6B 0F      rrca
0000:0B6C D8      ret      c
0000:0B6D 2A C4 63      ld        hl, (unk_0_63C4)
0000:0B70 7E      ld        a, (hl)
0000:0B71 FE 7F      cp        7Fh ; ' '
0000:0B73 CA 86 0B      jp        z, loc_0_B86
0000:0B76 23      inc      hl
0000:0B77 22 C4 63      ld        (unk_0_63C4), hl
0000:0B7A 21 0B 69      ld        hl, soft_sprite_ram+0Bh      ; sprite #2, x coord
0000:0B7D 4F      ld        c, a
0000:0B7E FF      rst      38h
0000:0B7F 21 08 69      ld        hl, soft_sprite_ram+8      ; sprite #2, y coord
0000:0B82 0E FF      ld        c, 0FFh
0000:0B84 FF      rst      38h      ; subtract 1 from y coord for 10 sprites
0000:0B85 C9      ret
0000:0B86      ; -----
0000:0B86
0000:0B86      loc_0_B86:                                ; CODE XREF: 0000:0B73|j
0000:0B86 21 CB 38      ld        hl, dk_intro_jump_across_data
0000:0B89 22 C4 63      ld        (unk_0_63C4), hl
0000:0B8C 3E 03      ld        a, 3      ; tmr=3
0000:0B8E 32 82 60      ld        (digital_snd_tmr_thump), a
0000:0B91 21 DC 38      ld        hl, draw_data_bend_girders_2
0000:0B94 3A 8D 63      ld        a, (unk_0_638D)
0000:0B97 3D      dec      a
0000:0B98 07      rlca
0000:0B99 07      rlca
0000:0B9A 07      rlca
0000:0B9B 07      rlca
0000:0B9C 5F      ld        e, a
0000:0B9D 16 00      ld        d, 0
0000:0B9F 19      add      hl, de
0000:0BA0 EB      ex       de, hl
0000:0BA1 CD A7 0D      call      sub_0_DA7
0000:0BA4 21 8D 63      ld        hl, unk_0_638D
0000:0BA7 35      dec      (hl)
0000:0BA8 C0      ret      nz
0000:0BA9 3E B0      ld        a, 0B0h ; '0'
0000:0BAB 32 09 60      ld        (eight_bit_countdown), a
0000:0BAE 21 85 63      ld        hl, intro_sequencer
0000:0BB1 34      inc      (hl)
0000:0BB2 C9      ret
0000:0BB3      ; -----
0000:0BB3
0000:0BB3      growl:                                ; DATA XREF: 0000:0A88|o
0000:0BB3

```

```

0000:0BB3 21 8A 60      ld      hl, unk_0_608A
0000:0BB6 3A 09 60      ld      a, (eight_bit_countdown)
0000:0BB9 FE 90      cp      90h ; 'E'
0000:0BBB 20 0B      jr      nz, loc_0_BC8
0000:0BBD 36 0F      ld      (hl), 0Fh
0000:0BBF 23      inc     hl
0000:0BC0 36 03      ld      (hl), 3
0000:0BC2 21 19 69    ld      hl, soft_sprite_ram+19h      ; sprite #6, flipy & code
0000:0BC5 34      inc     (hl)
0000:0BC6 18 09      jr      loc_0_BD1
;
0000:0BC8
0000:0BC8      loc_0_BC8:                                ; CODE XREF: 0000:0BBB|j
0000:0BC8 FE 18      cp      18h
0000:0BCA 20 05      jr      nz, loc_0_BD1
0000:0BCC 21 19 69    ld      hl, soft_sprite_ram+19h      ; sprite #6, flipy & code
0000:0BCF 35      dec     (hl)
0000:0BD0 00      nop
0000:0BD1
0000:0BD1      loc_0_BD1:                                ; CODE XREF: 0000:0BC6|j
0000:0BD1 DF      rst      18h                          ; 0000:0BCA|j
0000:0BD1      ; wait for 8-bit countdown
0000:0BD2 AF      xor     a
0000:0BD3 32 85 63    ld      (intro_sequencer), a
0000:0BD6 34      inc     (hl)
0000:0BD7 23      inc     hl
0000:0BD8 34      inc     (hl)
0000:0BD9 C9      ret
;
0000:0BDA
0000:0BDA      draw_how_high_can_you_get:                ; DATA XREF: 0000:0712|o
0000:0BDA CD 1C 01      call    stop_sound
0000:0BDD DF      rst      18h                          ; wait for 8-bit countdown
0000:0BDE CD 74 08    call    clear_visible_area_and_sprites
0000:0BE1 16 06      ld      d, 6
0000:0BE3 3A 00 62    ld      a, (mario_alive_flag)
0000:0BE6 5F      ld      e, a
0000:0BE7 CD 9F 30    call    sub_0_309F
0000:0BEA 21 86 7D    ld      hl, palette_bank
0000:0BED 36 01      ld      (hl), 1
0000:0BEF 23      inc     hl
0000:0BF0 36 00      ld      (hl), 0
0000:0BF2 21 8A 60    ld      hl, unk_0_608A
0000:0BF5 36 02      ld      (hl), 2
0000:0BF7 23      inc     hl
0000:0BF8 36 03      ld      (hl), 3
0000:0BFA 21 A7 63    ld      hl, unk_0_63A7
0000:0BFD 36 00      ld      (hl), 0
0000:0BFF 21 DC 76    ld      hl, VRAM_start+2DCh
0000:0C02 22 A8 63    ld      (unk_0_63A8), hl
0000:0C05 3A 2E 62    ld      a, (unk_0_622E)
0000:0C08 FE 06      cp      6
0000:0C0A 38 05      jr      c, loc_0_C11
0000:0C0C 3E 05      ld      a, 5
0000:0C0E 32 2E 62    ld      (unk_0_622E), a
0000:0C11
0000:0C11      loc_0_C11:                                ; CODE XREF: 0000:0C0A|j
0000:0C11 3A 2F 62      ld      a, (unk_0_622F)
0000:0C14 47      ld      b, a
0000:0C15 3A 2A 62    ld      a, (seq_data)
0000:0C18 B8      cp      b
0000:0C19 28 04      jr      z, loc_0_C1F
0000:0C1B 21 2E 62    ld      hl, unk_0_622E
0000:0C1E 34      inc     (hl)
0000:0C1F
0000:0C1F      loc_0_C1F:                                ; CODE XREF: 0000:0C19|j
0000:0C1F 32 2F 62      ld      (unk_0_622F), a
0000:0C22 3A 2E 62    ld      a, (unk_0_622E)
0000:0C25 47      ld      b, a
0000:0C26 21 BC 75    ld      hl, VRAM_start+1BCh
0000:0C29
0000:0C29      loc_0_C29:                                ; CODE XREF: 0000:0C27F|j
0000:0C29 0E 50      ld      c, 50h ; 'P'
0000:0C2B
0000:0C2B      loc_0_C2B:                                ; CODE XREF: 0000:0C40|j
0000:0C2B      ld      (hl), c
0000:0C2C 0C      inc     c
0000:0C2D 2B      dec     hl
0000:0C2E 71      ld      (hl), c
0000:0C2F 0C      inc     c
0000:0C30 2B      dec     hl
0000:0C31 71      ld      (hl), c
0000:0C32 0C      inc     c
0000:0C33 2B      dec     hl
0000:0C34 71      ld      (hl), c
0000:0C35 79      ld      a, c
0000:0C36 FE 67      cp      67h ; 'g'
0000:0C38 CA 43 0C      jp      z, loc_0_C43
0000:0C3B 0C      inc     c
0000:0C3C 11 23 00      ld      de, 23h ; '#'
0000:0C3F 19      add     hl, de
0000:0C40 C3 2B 0C      jp      loc_0_C2B
;
0000:0C43
0000:0C43      loc_0_C43:                                ; CODE XREF: 0000:0C38|j
0000:0C43 3A A7 63      ld      a, (unk_0_63A7)
0000:0C46 3C      inc     a
0000:0C47 32 A7 63      ld      (unk_0_63A7), a
0000:0C4A 3D      dec     a
0000:0C4B CB 27      sla     a
0000:0C4D CB 27      sla     a
0000:0C4F E5      push    hl
0000:0C50 21 F0 3C      ld      hl, draw_data_how_high
0000:0C53 C5      push    bc
0000:0C54 DD 2A A8 63    ld      ix, (unk_0_63A8)
0000:0C58 4F      ld      c, a
0000:0C59 06 00      ld      b, 0
0000:0C5B 09      add     hl, bc
0000:0C5C 7E      ld      a, (hl)
0000:0C5D DD 77 60      ld      (ix+60h), a
0000:0C60 23      inc     hl
0000:0C61 7E      ld      a, (hl)
0000:0C62 DD 77 40      ld      (ix+40h), a
0000:0C65 23      inc     hl
0000:0C66 7E      ld      a, (hl)
0000:0C67 DD 77 20      ld      (ix+20h), a

```

```

0000:0C6A DD 36 E0 8B      ld      (ix-20h), 8Bh ; 'i'
0000:0C6E C1              pop     bc
0000:0C6F DD E5              push    ix
0000:0C71 E1              pop     hl
0000:0C72 11 FC FF      ld      de, 0FFFCh
0000:0C75 19              add     hl, de
0000:0C76 22 A8 63      ld      (unk_0_63A8), hl
0000:0C79 E1              pop     hl
0000:0C7A 11 5F FF      ld      de, 0FF5Fh
0000:0C7D 19              add     hl, de
0000:0C7E 05              dec     b
0000:0C7F C2 29 0C      jp      nz, loc_0_C29
0000:0C82 11 07 03      ld      de, 307h
0000:0C85 CD 9F 30      call    sub_0_309F
0000:0C88 21 09 60      ld      hl, eight_bit_countdown
0000:0C8B 36 A0      ld      (hl), 0A0h ; 'á'
0000:0C8D 23              inc     hl
0000:0C8E 34              inc     (hl)
0000:0C8F 34              inc     (hl)
0000:0C90 C9              ret
0000:0C91
0000:0C91
0000:0C91
0000:0C91 DF      wait_init_and_draw_level:      ; DATA XREF: 0000:0716|o
                                rst      18h      ; wait for 8-bit countdown
0000:0C92
0000:0C92      init_and_draw_level:          ; CODE XREF: 0000:0776|j
0000:0C92 CD 74 08      call    clear_visible_area_and_sprites
0000:0C95 AF              xor     a
0000:0C96 32 8C 63      ld      (bonus_timer), a      ; init bonus timer
0000:0C99 11 01 05      ld      de, 501h
0000:0C9C CD 9F 30      call    sub_0_309F
0000:0C9F 21 86 7D      ld      hl, palette_bank
0000:0CA2 36 00      ld      (hl), 0
0000:0CA4 23              inc     hl
0000:0CA5 36 01      ld      (hl), 1      ; select palette bank 1
0000:0CA7 3A 27 62      ld      a, (level_type)
0000:0CAA 3D              dec     a      ; barrel level?
0000:0CAB CA D4 0C      jp      z, draw_barrel_level    ; yes, skip
0000:0CAE 3D              dec     a      ; cement pie level?
0000:0CAF CA DF 0C      jp      z, draw_cement_pie_level ; yes, skip
0000:0CB2 3D              dec     a      ; elevator level?
0000:0CB3 CA F2 0C      jp      z, draw_elevator_level  ; yes, skip
0000:0CB6 CD 43 0D      call    sub_0_D43
0000:0CB9 21 86 7D      ld      hl, palette_bank
0000:0CBC 36 01      ld      (hl), 1      ; select palette bank 3
0000:0CBE 3E 0B      ld      a, 0Bh
0000:0CC0 32 89 60      ld      (unk_0_6089), a
0000:0CC3 11 8B 3C      ld      de, rivet_level_tilemap_data
0000:0CC6
0000:0CC6      draw_level_tilemap:          ; CODE XREF: 0000:0CDC|j
                                ; 0000:0CEF|j ...
0000:0CC6 CD A7 0D      call    sub_0_DA7      ; draw screen
0000:0CC9 3A 27 62      ld      a, (level_type)
0000:0CCC FE 04      cp      4      ; rivets?
0000:0CCE CC 00 0D      call    z, draw_rivets      ; yes, call
0000:0CD1 C3 A0 3F      jp      loc_0_3FA0
0000:0CD4
0000:0CD4
0000:0CD4      draw_barrel_level:          ; CODE XREF: 0000:0CAB|j
0000:0CD4 11 E4 3A      ld      de, barrel_level_tilemap_data
0000:0CD7 3E 08      ld      a, 8
0000:0CD9 32 89 60      ld      (unk_0_6089), a
0000:0CDC C3 C6 0C      jp      draw_level_tilemap
0000:0CDF
0000:0CDF
0000:0CDF      draw_cement_pie_level:      ; CODE XREF: 0000:0CAF|j
0000:0CDF 11 5D 3B      ld      de, cement_pie_level_tilemap_data
0000:0CE2 21 86 7D      ld      hl, palette_bank
0000:0CE5 36 01      ld      (hl), 1
0000:0CE7 23              inc     hl
0000:0CE8 36 00      ld      (hl), 0
0000:0CEA 3E 09      ld      a, 9
0000:0CEC 32 89 60      ld      (unk_0_6089), a
0000:0CEF C3 C6 0C      jp      draw_level_tilemap
0000:0CF2
0000:0CF2
0000:0CF2      draw_elevator_level:        ; CODE XREF: 0000:0CB3|j
0000:0CF2 CD 27 0D      call    sub_0_D27
0000:0CF5 3E 0A      ld      a, 0Ah
0000:0CF7 32 89 60      ld      (unk_0_6089), a
0000:0CFA 11 E5 3B      ld      de, elevator_level_tilemap_data
0000:0CFD C3 C6 0C      jp      draw_level_tilemap
0000:0D00
0000:0D00
0000:0D00
0000:0D00      ; ██████████ SUBROUTINE ██████████
0000:0D00
0000:0D00      draw_rivets:                ; CODE XREF: 0000:0CCE|p
                                ; 8 rivets
0000:0D00 06 08      ld      b, 8
0000:0D02 21 17 0D      ld      hl, rivet_loc_tbl
0000:0D05
0000:0D05      loc_0_D05:                  ; CODE XREF: draw_rivets+14|j
                                ; top of rivet tile
                                ; 2 tiles/rivet (vertical)
0000:0D05 3E B8      ld      a, 0B8h ; '@'
0000:0D07 0E 02      ld      c, 2
0000:0D09 5E      ld      e, (hl)
0000:0D0A 23              inc     hl
0000:0D0B 56      ld      d, (hl)
0000:0D0C 23              inc     hl      ; get VRAM location
0000:0D0D
0000:0D0D      loc_0_D0D:                  ; CODE XREF: draw_rivets+11|j
                                ; draw rivet tile
                                ; next rivet tile
                                ; next VRAM location
0000:0D0D 12      ld      (de), a
0000:0D0E 3D      dec     a
0000:0D0F 13      inc     de
0000:0D10 0D      dec     c
0000:0D11 C2 0D 0D      jp      nz, loc_0_D0D
0000:0D14 10 EF      djnz    loc_0_D05      ; done a rivet?
                                ; no, loop
0000:0D16 C9              ret      ; loop through 8 rivets
0000:0D16
0000:0D16
0000:0D16      ; End of function draw_rivets
0000:0D16
0000:0D16
0000:0D17 CA 76      rivet_loc_tbl: .dw VRAM_start+2CAh      ; DATA XREF: draw_rivets+2|o
                                ; Rivets level, location of rivets
0000:0D17
0000:0D19 CF 76      .dw VRAM_start+2CFh
0000:0D1B D4 76      .dw VRAM_start+2D4h
0000:0D1D D9 76      .dw VRAM_start+2D9h
0000:0D1F 2A 75      .dw VRAM_start+12Ah
0000:0D21 2F 75      .dw VRAM_start+12Fh

```

```

0000:0D23 34 75      .dw VRAM_start+134h
0000:0D25 39 75      .dw VRAM_start+139h
0000:0D27
0000:0D27      ; ██████████ S U B R O U T I N E ██████████
0000:0D27
0000:0D27      sub_0_D27:                                ; CODE XREF: 0000:0CF2|p
0000:0D27      ld      hl, VRAM_start+30Dh
0000:0D2A CD 30 0D      call    sub_0_D30
0000:0D2D 21 0D 76      ld      hl, VRAM_start+20Dh
0000:0D2D      ; End of function sub_0_D27
0000:0D2D
0000:0D30
0000:0D30      ; ██████████ S U B R O U T I N E ██████████
0000:0D30
0000:0D30      sub_0_D30:                                ; CODE XREF: sub_0_D27+3|p
0000:0D30 06 11      ld      b, 11h
0000:0D32
0000:0D32      loc_0_D32:                                ; CODE XREF: sub_0_D30+5|j
0000:0D32 36 FD      ld      (hl), 0FDh ; '2'
0000:0D34 23      inc     hl
0000:0D35 10 FB      djnz    loc_0_D32
0000:0D37 11 0F 00      ld      de, 0Fh
0000:0D3A 19      add     hl, de
0000:0D3B 06 11      ld      b, 11h
0000:0D3D
0000:0D3D      loc_0_D3D:                                ; CODE XREF: sub_0_D30+10|j
0000:0D3D 36 FC      ld      (hl), 0FCh ; '3'
0000:0D3F 23      inc     hl
0000:0D40 10 FB      djnz    loc_0_D3D
0000:0D42 C9      ret
0000:0D42      ; End of function sub_0_D30
0000:0D43
0000:0D43      ; ██████████ S U B R O U T I N E ██████████
0000:0D43
0000:0D43      sub_0_D43:                                ; CODE XREF: 0000:0CB6|p
0000:0D43      ld      hl, VRAM_start+287h
0000:0D46 CD 4C 0D      call    sub_0_D4C
0000:0D49 21 47 75      ld      hl, VRAM_start+147h
0000:0D49      ; End of function sub_0_D43
0000:0D4C
0000:0D4C      ; ██████████ S U B R O U T I N E ██████████
0000:0D4C
0000:0D4C      sub_0_D4C:                                ; CODE XREF: sub_0_D43+3|p
0000:0D4C 06 04      ld      b, 4
0000:0D4E
0000:0D4E      loc_0_D4E:                                ; CODE XREF: sub_0_D4C+5|j
0000:0D4E 36 FD      ld      (hl), 0FDh ; '2'
0000:0D50 23      inc     hl
0000:0D51 10 FB      djnz    loc_0_D4E
0000:0D53 11 1C 00      ld      de, 1Ch
0000:0D56 19      add     hl, de
0000:0D57 06 04      ld      b, 4
0000:0D59
0000:0D59      loc_0_D59:                                ; CODE XREF: sub_0_D4C+10|j
0000:0D59 36 FC      ld      (hl), 0FCh ; '3'
0000:0D5B 23      inc     hl
0000:0D5C 10 FB      djnz    loc_0_D59
0000:0D5E C9      ret
0000:0D5E      ; End of function sub_0_D4C
0000:0D5E
0000:0D5F      ; -----
0000:0D5F      loc_0_D5F:                                ; CODE XREF: 0000:3FA3|j
0000:0D5F CD 56 0F      call    loc_0_F56
0000:0D62 CD 41 24      call    sub_0_2441
0000:0D65 21 09 60      ld      hl, eight_bit_countdown
0000:0D68 36 40      ld      (hl), 40h ; '@'
0000:0D6A 23      inc     hl
0000:0D6B 34      inc     (hl)
0000:0D6C 21 5C 38      ld      hl, kong_normal_spr
0000:0D6F CD 4E 00      call    copy_sprites_2_11_data
0000:0D72 11 00 69      ld      de, soft_sprite_ram
0000:0D75 01 08 00      ld      bc, 8
0000:0D78 ED B0      ldir
0000:0D7A 3A 27 62      ld      a, (level_type)
0000:0D7D FE 04      cp      4
0000:0D7F 28 0A      jr      z, loc_0_D8B
0000:0D81 0F      rrca
0000:0D82 0F      rrca
0000:0D83 D8      ret
0000:0D84 21 0B 69      ld      hl, soft_sprite_ram+0Bh
0000:0D87 0E FC      ld      c, 0FCh ; '3'
0000:0D89 FF      rst
0000:0D8A C9      ret
0000:0D8A      ; subtract 4 from x coord for 10 sprites
0000:0D8B
0000:0D8B      ; -----
0000:0D8B      loc_0_D8B:                                ; CODE XREF: 0000:0D7F|j
0000:0D8B      ld      hl, soft_sprite_ram+8
0000:0D8E 0E 44      ld      c, 68
0000:0D90 FF      rst
0000:0D91 11 04 00      ld      de, 4
0000:0D94 01 10 02      ld      bc, 210h
0000:0D97 21 00 69      ld      hl, soft_sprite_ram
0000:0D9A CD 3D 00      call    loc_0_3D
0000:0D9D 01 F8 02      ld      bc, 2F8h
0000:0DA0 21 03 69      ld      hl, soft_sprite_ram+3
0000:0DA3 CD 3D 00      call    loc_0_3D
0000:0DA6 C9      ret
0000:0DA7
0000:0DA7      ; ██████████ S U B R O U T I N E ██████████
0000:0DA7
0000:0DA7      sub_0_DA7:                                ; CODE XREF: 0000:0A95|p
0000:0DA7      ld      a, (de)
0000:0DA7 1A      ld      (unk_0_63B3), a
0000:0DA8 32 B3 63      cp      0AAh ; '-'
0000:0DAB FE AA      ret
0000:0DAD C8      ret
0000:0DAE 13      inc     de
0000:0DAF 1A      ld      a, (de)
0000:0DAF      ; get flag/tile
0000:0DAF      ; store for later
0000:0DAF      ; done?
0000:0DAF      ; yes, return
0000:0DAF      ; next table address
0000:0DAF      ; get byte

```

```

0000:0DB0 67      ld      h, a      ; H=byte1
0000:0DB1 44      ld      b, h      ; B=byte1
0000:0DB2 13      inc     de        ; next table address
0000:0DB3 1A      ld      a, (de)   ; get byte
0000:0DB4 6F      ld      l, a      ; L=byte2
0000:0DB5 4D      ld      c, l      ; C=byte2
0000:0DB6 D5      push   de
0000:0DB7 CD F0 2F call    sub_0_2FF0
0000:0DBA D1      pop     de
0000:0DBB 22 AB 63 ld      (unk_0_63AB), hl
0000:0DBE 78      ld      a, b
0000:0DBF E6 07      and     7
0000:0DC1 32 B4 63 ld      (unk_0_63B4), a
0000:0DC4 79      ld      a, c
0000:0DC5 E6 07      and     7
0000:0DC7 32 AF 63 ld      (unk_0_63AF), a
0000:0DCA 13      inc     de
0000:0DCB 1A      ld      a, (de)
0000:0DCC 67      ld      h, a
0000:0DCD 90      sub     b
0000:0DCE D2 D3 0D jp      nc, loc_0_DD3
0000:0DD1 ED 44      neg
0000:0DD3      loc_0_DD3:      ; CODE XREF: sub_0_DA7+27↑j
0000:0DD3 32 B1 63      ld      (unk_0_63B1), a
0000:0DD6 13      inc     de
0000:0DD7 1A      ld      a, (de)
0000:0DD8 6F      ld      l, a
0000:0DD9 91      sub     c
0000:0DDA 32 B2 63 ld      (unk_0_63B2), a
0000:0DDD 1A      ld      a, (de)
0000:0DDE E6 07      and     7
0000:0DE0 32 B0 63 ld      (unk_0_63B0), a
0000:0DE3 D5      push   de
0000:0DE4 CD F0 2F call    sub_0_2FF0
0000:0DE7 D1      pop     de
0000:0DE8 22 AD 63 ld      (unk_0_63AD), hl
0000:0DEB 3A B3 63 ld      a, (unk_0_63B3)
0000:0DEE FE 02      cp      2
0000:0DF0 F2 4F 0E jp      p, loc_0_E4F
0000:0DF3 3A B2 63 ld      a, (unk_0_63B2)
0000:0DF6 D6 10      sub     10h
0000:0DF8 47      ld      b, a
0000:0DF9 3A AF 63 ld      a, (unk_0_63AF)
0000:0DFC 80      add     a, b
0000:0DFD 32 B2 63 ld      (unk_0_63B2), a
0000:0E00 3A AF 63 ld      a, (unk_0_63AF)
0000:0E03 C6 F0      add     a, 0F0h ; '-'
0000:0E05 2A AB 63 ld      hl, (unk_0_63AB)
0000:0E08 77      ld      (hl), a
0000:0E09 2C      inc     l
0000:0E0A D6 30      sub     30h ; '0'
0000:0E0C 77      ld      (hl), a
0000:0E0D 3A B3 63 ld      a, (unk_0_63B3)
0000:0E10 FE 01      cp      1
0000:0E12 C2 19 0E jp      nz, loc_0_E19
0000:0E15 AF      xor     a
0000:0E16 32 B2 63 ld      (unk_0_63B2), a
0000:0E19      loc_0_E19:      ; CODE XREF: sub_0_DA7+6B↑j
0000:0E19      ; sub_0_DA7+80↑j
0000:0E19 3A B2 63      ld      a, (unk_0_63B2)
0000:0E19      ld      sub     8
0000:0E1C D6 08      ld      (unk_0_63B2), a
0000:0E1E 32 B2 63 ld      c, loc_0_E2A
0000:0E21 DA 2A 0E jp      l
0000:0E24 2C      inc     l
0000:0E25 36 C0      ld      (hl), 0C0h ; 'L'
0000:0E27 C3 19 0E jp      loc_0_E19
0000:0E2A      ; _____
0000:0E2A      loc_0_E2A:      ; CODE XREF: sub_0_DA7+7A↑j
0000:0E2A 3A B0 63      ld      a, (unk_0_63B0)
0000:0E2D C6 D0      add     a, 0D0h ; 'ð'
0000:0E2F 2A AD 63 ld      hl, (unk_0_63AD)
0000:0E32 77      ld      (hl), a
0000:0E33 3A B3 63 ld      a, (unk_0_63B3)
0000:0E36 FE 01      cp      1
0000:0E38 C2 3F 0E jp      nz, loc_0_E3F
0000:0E3B 2D      dec     l
0000:0E3C 36 C0      ld      (hl), 0C0h ; 'L'
0000:0E3E 2C      inc     l
0000:0E3F      loc_0_E3F:      ; CODE XREF: sub_0_DA7+91↑j
0000:0E3F 3A B0 63      ld      a, (unk_0_63B0)
0000:0E42 FE 00      cp      0
0000:0E44 CA 4B 0E jp      z, loc_0_E4B
0000:0E47 C6 E0      add     a, 0E0h ; 'Ö'
0000:0E49 2C      inc     l
0000:0E4A 77      ld      (hl), a
0000:0E4B      loc_0_E4B:      ; CODE XREF: sub_0_DA7+9D↑j
0000:0E4B 13      inc     de
0000:0E4C C3 A7 0D jp      sub_0_DA7
0000:0E4F      ; _____
0000:0E4F      loc_0_E4F:      ; CODE XREF: sub_0_DA7+49↑j
0000:0E4F 3A B3 63      ld      a, (unk_0_63B3)
0000:0E52 FE 02      cp      2
0000:0E54 C2 E8 0E jp      nz, loc_0_EE8
0000:0E57 3A AF 63 ld      a, (unk_0_63AF)
0000:0E5A C6 F0      add     a, 0F0h ; '-'
0000:0E5C 32 B5 63 ld      (unk_0_63B5), a
0000:0E5F 2A AB 63 ld      hl, (unk_0_63AB)
0000:0E62      loc_0_E62:      ; CODE XREF: sub_0_DA7+E5↑j
0000:0E62      ; sub_0_DA7+125↑j ...
0000:0E62 3A B5 63      ld      a, (unk_0_63B5)
0000:0E65 77      ld      (hl), a
0000:0E66 23      inc     hl
0000:0E67 7D      ld      a, l
0000:0E68 E6 1F      and     1Fh
0000:0E6A CA 78 0E jp      z, loc_0_E78
0000:0E6D 3A B5 63 ld      a, (unk_0_63B5)
0000:0E70 FE F0      cp      0F0h ; '-'
0000:0E72 CA 78 0E jp      z, loc_0_E78
0000:0E75 D6 10      sub     10h

```

```

0000:0E77 77          ld      (hl), a
0000:0E78          ld      bc, 1Fh
0000:0E78          loc_0_E78:          ; CODE XREF: sub_0_DA7+C3↑j
0000:0E78 01 1F 00          ; sub_0_DA7+CB↑j
0000:0E78          ld      bc, 1Fh
0000:0E7B 09          add     hl, bc
0000:0E7C 3A B1 63          ld      a, (unk_0_63B1)
0000:0E7F D6 08          sub     8
0000:0E81 DA CF 0E          jp      c, loc_0_ECF
0000:0E84 32 B1 63          ld      (unk_0_63B1), a
0000:0E87 3A B2 63          ld      a, (unk_0_63B2)
0000:0E8A FE 00          cp      0
0000:0E8C CA 62 0E          jp      z, loc_0_E62
0000:0E8F 3A B5 63          ld      a, (unk_0_63B5)
0000:0E92 77          ld      (hl), a
0000:0E93 23          inc     hl
0000:0E94 7D          ld      a, 1
0000:0E95 E6 1F          and     1Fh
0000:0E97 CA A0 0E          jp      z, loc_0_EA0
0000:0E9A 3A B5 63          ld      a, (unk_0_63B5)
0000:0E9D D6 10          sub     10h
0000:0E9F 77          ld      (hl), a
0000:0EA0          loc_0_EA0:          ; CODE XREF: sub_0_DA7+F0↑j
0000:0EA0 01 1F 00          ld      bc, 1Fh
0000:0EA3 09          add     hl, bc
0000:0EA4 3A B1 63          ld      a, (unk_0_63B1)
0000:0EA7 D6 08          sub     8
0000:0EA9 DA CF 0E          jp      c, loc_0_ECF
0000:0EAC 32 B1 63          ld      (unk_0_63B1), a
0000:0EAF 3A B2 63          ld      a, (unk_0_63B2)
0000:0EB2 CB 7F          bit     7, a
0000:0EB4 C2 D3 0E          jp      nz, loc_0_ED3
0000:0EB7 3A B5 63          ld      a, (unk_0_63B5)
0000:0EBA 3C          inc     a
0000:0EBB 32 B5 63          ld      (unk_0_63B5), a
0000:0EBE FE F8          cp      0F8h ; '0'
0000:0EC0 C2 C9 0E          jp      nz, loc_0_EC9
0000:0EC3 23          inc     hl
0000:0EC4 3E F0          ld      a, 0F0h ; '-'
0000:0EC6 32 B5 63          ld      (unk_0_63B5), a
0000:0EC9          loc_0_EC9:          ; CODE XREF: sub_0_DA7+119↑j
0000:0EC9 7D          ld      a, 1
0000:0ECA E6 1F          and     1Fh
0000:0ECC C2 62 0E          jp      nz, loc_0_E62
0000:0ECF          loc_0_ECF:          ; CODE XREF: sub_0_DA7+DA↑j
0000:0ECF 13          ; sub_0_DA7+102↑j ...
0000:0ECF          inc     de
0000:0ED0 C3 A7 0D          jp      sub_0_DA7
0000:0ED3          ;
0000:0ED3          loc_0_ED3:          ; CODE XREF: sub_0_DA7+10D↑j
0000:0ED3 3A B5 63          ld      a, (unk_0_63B5)
0000:0ED6 3D          dec     a
0000:0ED7 32 B5 63          ld      (unk_0_63B5), a
0000:0EDA FE F0          cp      0F0h ; '-'
0000:0EDC F2 E5 0E          jp      p, loc_0_EE5
0000:0EDF 2B          dec     hl
0000:0EE0 3E F7          ld      a, 0F7h ; ','
0000:0EE2 32 B5 63          ld      (unk_0_63B5), a
0000:0EE5          loc_0_EE5:          ; CODE XREF: sub_0_DA7+135↑j
0000:0EE5          ;
0000:0EE5 C3 62 0E          jp      loc_0_E62
0000:0EE8          loc_0_EE8:          ; CODE XREF: sub_0_DA7+AD↑j
0000:0EE8 3A B3 63          ld      a, (unk_0_63B3)
0000:0EEB FE 03          cp      3
0000:0EED C2 1B 0F          jp      nz, loc_0_F1B
0000:0EEF 2A AB 63          ld      hl, (unk_0_63AB)
0000:0EF3 3E B3          ld      a, 0B3h ; '3'
0000:0EF5 77          ld      (hl), a
0000:0EF6 01 20 00          ld      bc, 20h ; ' '
0000:0EF9 09          add     hl, bc
0000:0EFA 3A B1 63          ld      a, (unk_0_63B1)
0000:0EFD D6 10          sub     10h
0000:0EFF          loc_0_EFF:          ; CODE XREF: sub_0_DA7+16A↑j
0000:0EFF          ;
0000:0EFF DA 14 0F          jp      c, loc_0_F14
0000:0F02 32 B1 63          ld      (unk_0_63B1), a
0000:0F05 3E B1          ld      a, 0B1h ; '1'
0000:0F07 77          ld      (hl), a
0000:0F08 01 20 00          ld      bc, 20h ; ' '
0000:0F0B 09          add     hl, bc
0000:0F0C 3A B1 63          ld      a, (unk_0_63B1)
0000:0F0F D6 08          sub     8
0000:0F11 C3 FF 0E          jp      loc_0_EFF
0000:0F14          ;
0000:0F14          loc_0_F14:          ; CODE XREF: sub_0_DA7+158↑j
0000:0F14 3E B2          ld      a, 0B2h ; '2'
0000:0F16 77          ld      (hl), a
0000:0F17 13          inc     de
0000:0F18 C3 A7 0D          jp      sub_0_DA7
0000:0F1B          ;
0000:0F1B          loc_0_F1B:          ; CODE XREF: sub_0_DA7+146↑j
0000:0F1B 3A B3 63          ld      a, (unk_0_63B3)
0000:0F1E FE 07          cp      7
0000:0F20 F2 CF 0E          jp      p, loc_0_ECF
0000:0F23 FE 04          cp      4
0000:0F25 CA 4C 0F          jp      z, erase_straight_girders
0000:0F28 FE 05          cp      5
0000:0F2A CA 51 0F          jp      z, loc_0_F51
0000:0F2D 3E FE          ld      a, 0FEh ; '■'
0000:0F2F          loc_0_F2F:          ; CODE XREF: sub_0_DA7+1A7↑j
0000:0F2F 32 B5 63          ; sub_0_DA7+1AC↑j
0000:0F2F          ld      (unk_0_63B5), a
0000:0F32 2A AB 63          ld      hl, (unk_0_63AB)
0000:0F35          loc_0_F35:          ; CODE XREF: sub_0_DA7+19E↑j
0000:0F35 3A B5 63          ld      a, (unk_0_63B5)
0000:0F38 77          ld      (hl), a

```



```

0000:0F39 01 20 00      ld      bc, 20h ; ' '
0000:0F3C 09              add     hl, bc
0000:0F3D 3A B1 63      ld      a, (unk_0_63B1)
0000:0F40 D6 08              sub     8
0000:0F42 32 B1 63      ld      (unk_0_63B1), a
0000:0F45 D2 35 0F      jp      nc, loc_0_F35
0000:0F48 13              inc     de
0000:0F49 C3 A7 0D      jp      sub_0_DA7
0000:0F4C              ;
0000:0F4C
0000:0F4C
0000:0F4C 3E E0      erase_straight_girders:      ; CODE XREF: sub_0_DA7+17E|j
                                ld      a, 0E0h ; 'Ó'
0000:0F4E C3 2F 0F      jp      loc_0_F2F
0000:0F51              ;
0000:0F51
0000:0F51
0000:0F51 3E B0      loc_0_F51:      ld      a, 0B0h ; ''
                                jp      loc_0_F2F
0000:0F53 C3 2F 0F      ; End of function sub_0_DA7
0000:0F53
0000:0F53
0000:0F56
0000:0F56
0000:0F56
0000:0F56 06 27      loc_0_F56:      ld      b, 27h ; '7'
                                ld      hl, mario_alive_flag
0000:0F58 21 00 62      xor     a
0000:0F5B AF
0000:0F5C
0000:0F5C      loc_0_F5C:      ; CODE XREF: 0000:0F5E|j
                                ld      (hl), a
0000:0F5C 77              inc     l
0000:0F5D 2C              djnz    loc_0_F5C
0000:0F5E 10 FC      ld      c, 11h
0000:0F60 0E 11      ld      d, 80h ; 'Ç'
0000:0F62 16 80      ld      hl, unk_0_6280
0000:0F64 21 80 62
0000:0F67
0000:0F67      loc_0_F67:      ; CODE XREF: 0000:0F6D|j
                                ld      b, d
0000:0F67 42
0000:0F68
0000:0F68      loc_0_F68:      ; CODE XREF: 0000:0F6A|j
                                ld      (hl), a
0000:0F68 77              inc     hl
0000:0F69 23              djnz    loc_0_F68
0000:0F6A 10 FC      dec     c
0000:0F6C 0D              jr      nz, loc_0_F67
0000:0F6D 20 F8      ld      hl, level_init_data
0000:0F6F 21 9C 3D      ld      de, unk_0_6280
0000:0F72 11 80 62      ld      bc, 40h ; '@'
0000:0F75 01 40 00      ldir
0000:0F78 ED B0      ld      a, (level)
0000:0F7A 3A 29 62      ld      b, a
0000:0F7D 47              and     a
0000:0F7E A7              rla
0000:0F7F 17              and     a
0000:0F80 A7              rla
0000:0F81 17              and     a
0000:0F82 A7              rla
0000:0F83 17              add     a, b
0000:0F84 80              add     a, b
0000:0F85 80              add     a, 28h ; '('
0000:0F86 C6 28      cp      51h ; 'Q'
0000:0F88 FE 51      jr      c, loc_0_F8E
0000:0F8A 38 02      ld      a, 50h ; 'P'
0000:0F8C 3E 50
0000:0F8E
0000:0F8E      loc_0_F8E:      ; CODE XREF: 0000:0F8A|j
                                ld      hl, unk_0_62B0
0000:0F91 06 03      ld      b, 3
0000:0F93
0000:0F93      loc_0_F93:      ; CODE XREF: 0000:0F95|j
                                ld      (hl), a
0000:0F93 77              inc     l
0000:0F94 2C              djnz    loc_0_F93
0000:0F95 10 FC      add     a, a
0000:0F97 87      ld      b, a
0000:0F98 47      ld      a, 0DCh ; ''
0000:0F99 3E DC      sub     b
0000:0F9B 90      cp      28h ; '('
0000:0F9C FE 28      jr      nc, loc_0_FA2
0000:0F9E 30 02      ld      a, 28h ; '('
0000:0FA0 3E 28
0000:0FA2
0000:0FA2      loc_0_FA2:      ; CODE XREF: 0000:0F9E|j
                                ld      (hl), a
0000:0FA2 77              inc     l
0000:0FA3 2C      ld      (hl), a
0000:0FA4 77      ld      hl, unk_0_6209
0000:0FA5 21 09 62      ld      (hl), 4
0000:0FA8 36 04      inc     l
0000:0FAA 2C      ld      (hl), 8
0000:0FAB 36 08      ld      a, (level_type)
0000:0FAD 3A 27 62      ld      c, a
0000:0FB0 4F      bit     2, a
0000:0FB1 CB 57      jr      nz, loc_0_FCB
0000:0FB3 20 16      ld      hl, soft_sprite_ram+100h
0000:0FB5 21 00 6A      ld      a, 4Fh ; 'O'
0000:0FB8 3E 4F      ld      b, 3
0000:0FBA 06 03
0000:0FBC
0000:0FBC      loc_0_FBC:      ; CODE XREF: 0000:0FC9|j
                                ld      (hl), a
0000:0FBC 77              inc     l
0000:0FBD 2C      ld      (hl), 3Ah ; ':'
0000:0FBE 36 3A      inc     l
0000:0FC0 2C      ld      (hl), 0Fh
0000:0FC1 36 0F      inc     l
0000:0FC3 2C      ld      (hl), 18h
0000:0FC4 36 18      inc     l
0000:0FC6 2C      add     a, 10h
0000:0FC7 C6 10      djnz    loc_0_FBC
0000:0FC9 10 F1
0000:0FCB
0000:0FCB      loc_0_FCB:      ; CODE XREF: 0000:0FB3|j
                                ld      a, c
0000:0FCB 79      rst     28h
0000:0FCC EF      ; go!
0000:0FCC
0000:0FCC
0000:0FCD 00 00      ;
                                .dw RESET
0000:0FCD 00 00      ; Jump table
0000:0FCD 00 00
0000:0FCD D7 0F      .dw init_l1_girder
0000:0FDD 1F 10      .dw init_l2_cement
0000:0FDD 1F 10      .dw init_l3_elevator
0000:0FDD 87 10      .dw init_l4_rivets
0000:0FDD 31 11

```

```

0000:0FD7 ; -----
0000:0FD7
0000:0FD7
0000:0FD7 21 DC 3D init_l1_girder: ; DATA XREF: 0000:0FCF|o
0000:0FDA 11 A8 69 ld hl, top_barrel_spr
0000:0FDD 01 10 00 ld de, soft_sprite_ram+0A8h
0000:0FE0 ED B0 ld bc, 10h
0000:0FE2 21 EC 3D ldir
0000:0FE5 11 07 64 ld hl, fireball_spr
0000:0FE8 0E 1C ld de, unk_0_6407
0000:0FEA 06 05 ld c, 1Ch
0000:0FEC CD 2A 12 ld b, 5
0000:0FEF 21 F4 3D call sub_0_122A
0000:0FF2 CD FA 11 ld hl, girders_extra_spr
0000:0FF5 21 00 3E call sub_0_11FA
0000:0FF8 11 FC 69 ld hl, girder_oil_barrel_spr
0000:0FFB 01 04 00 ld de, soft_sprite_ram+0FCh
0000:0FFE ED B0 ld bc, 4
0000:1000 21 0C 3E ldir
0000:1003 CD A6 11 ld hl, girder_hammer_locs
0000:1006 call sub_0_11A6
0000:1006
0000:1006 21 1B 10 loc_0_1006: ld hl, cement_unknown_spr
0000:1009 11 07 67 ld de, unk_0_6707
0000:100C 01 1C 08 ld bc, 81Ch
0000:100F CD 2A 12 call sub_0_122A
0000:1012 11 07 68 ld de, unk_0_6807
0000:1015 06 02 ld b, 2
0000:1017 CD 2A 12 call sub_0_122A
0000:101A C9 ret
0000:101A
0000:101B 00 00 02 02 cement_unknown_spr:.db 0, 0, 2, 2 ; DATA XREF: 0000:1006|o
0000:101F ; -----
0000:101F
0000:101F init_l2_cement: ; DATA XREF: 0000:0FD1|o
0000:101F 21 EC 3D ld hl, fireball_spr
0000:1022 11 07 64 ld de, unk_0_6407
0000:1025 01 1C 05 ld bc, 51Ch
0000:1028 CD 2A 12 call sub_0_122A
0000:102B CD 86 11 call sub_0_1186
0000:102E 21 18 3E ld hl, cement_pie_spr
0000:1031 11 A7 65 ld de, unk_0_65A7
0000:1034 01 0C 06 ld bc, 60Ch
0000:1037 CD 2A 12 call sub_0_122A
0000:103A DD 21 A0 65 ld ix, unk_0_65A0
0000:103E 21 B8 69 ld hl, soft_sprite_ram+0B8h
0000:1041 11 10 00 ld de, 10h
0000:1044 06 06 ld b, 6
0000:1046 CD D3 11 call set_B_sprites_data
0000:1049 21 FA 3D ld hl, cement_extra_spr
0000:104C CD FA 11 call sub_0_11FA
0000:104F 21 04 3E ld hl, cement_oil_barrel_spr
0000:1052 11 FC 69 ld de, soft_sprite_ram+0FCh
0000:1055 01 04 00 ld bc, 4
0000:1058 ED B0 ldir
0000:105A 21 1C 3E ld hl, cement_ladder_spr
0000:105D 11 44 69 ld de, soft_sprite_ram+44h
0000:1060 01 08 00 ld bc, 8
0000:1063 ED B0 ldir
0000:1065 21 24 3E ld hl, cement_conveyer_spr
0000:1068 11 E4 69 ld de, soft_sprite_ram+0E4h
0000:106B 01 18 00 ld bc, 18h
0000:106E ED B0 ldir
0000:1070 21 10 3E ld hl, cement_hammer_locs
0000:1073 CD A6 11 call sub_0_11A6
0000:1076 21 3C 3E ld hl, cement_obj_spr
0000:1079 11 0C 6A ld de, soft_sprite_ram+10Ch
0000:107C 01 0C 00 ld bc, 0Ch
0000:107F ED B0 ldir
0000:1081 3E 01 ld a, 1
0000:1083 32 B9 62 ld (unk_0_62B9), a
0000:1086 C9 ret
0000:1087 ; -----
0000:1087
0000:1087
0000:1087 init_l3_elevator: ; DATA XREF: 0000:0FD3|o
0000:1087 21 EC 3D ld hl, fireball_spr
0000:108A 11 07 64 ld de, unk_0_6407
0000:108D 01 1C 05 ld bc, 51Ch
0000:1090 CD 2A 12 call sub_0_122A
0000:1093 CD 86 11 call sub_0_1186
0000:1096 21 00 66 ld hl, unk_0_6600
0000:1099 11 10 00 ld de, 10h
0000:109C 3E 01 ld a, 1
0000:109E 06 06 ld b, 6
0000:10A0
0000:10A0
0000:10A0 77 loc_0_10A0: ; CODE XREF: 0000:10A2|j
0000:10A1 19 ld (hl), a
0000:10A2 10 FC add hl, de
0000:10A4 0E 02 djnz loc_0_10A0
0000:10A6 3E 08 ld c, 2
0000:10A8 ld a, 8
0000:10A8
0000:10A8 06 03 loc_0_10A8: ; CODE XREF: 0000:10B4|j
0000:10AA 21 0D 66 ld b, 3
0000:10AD ld hl, unk_0_660D
0000:10AD
0000:10AD
0000:10AD 77 loc_0_10AD: ; CODE XREF: 0000:10AF|j
0000:10AE 19 ld (hl), a
0000:10AF 10 FC add hl, de
0000:10B1 3E 08 djnz loc_0_10AD
0000:10B3 0D ld a, 8
0000:10B4 C2 A8 10 dec c
0000:10B7 21 64 3E jnp nz, loc_0_10A8
0000:10BA 11 03 66 ld hl, elevator_spr_xy
0000:10BD 01 0E 06 ld de, unk_0_6603
0000:10C0 CD EC 11 ld bc, 60Eh
0000:10C3 21 60 3E call sub_0_11EC
0000:10C6 11 07 66 ld hl, elevator_spr
0000:10C9 01 0C 06 ld de, unk_0_6607
0000:10CC CD 2A 12 ld bc, 60Ch
0000:10CF DD 21 00 66 call sub_0_122A
0000:10D3 21 58 69 ld ix, unk_0_6600
0000:10D6 06 06 ld hl, soft_sprite_ram+58h
0000:10D8 11 10 00 ld b, 6
0000:10DB CD D3 11 ld de, 10h
call set_B_sprites_data

```

```

0000:10DE 21 48 3E      ld      hl, elevator_obj_spr
0000:10E1 11 0C 6A      ld      de, soft_sprite_ram+10Ch
0000:10E4 01 0C 00      ld      bc, 0Ch
0000:10E7 ED B0      ldir
0000:10E9 DD 21 00 64      ld      ix, unk_0_6400 ; fireball character data
0000:10ED DD 36 00 01      ld      (ix+0), 1
0000:10F1 DD 36 03 58      ld      (ix+3), 58h ; 'X'
0000:10F5 DD 36 0E 58      ld      (ix+0Eh), 58h ; 'X'
0000:10F9 DD 36 05 80      ld      (ix+5), 80h ; 'Ç'
0000:10FD DD 36 0F 80      ld      (ix+0Fh), 80h ; 'Ç'
0000:1101 DD 36 20 01      ld      (ix+20h), 1
0000:1105 DD 36 23 EB      ld      (ix+23h), 0EBh ; 'Ü'
0000:1109 DD 36 2E EB      ld      (ix+2Eh), 0EBh ; 'Ü'
0000:110D DD 36 25 60      ld      (ix+25h), 60h ; ' '
0000:1111 DD 36 2F 60      ld      (ix+2Fh), 60h ; ' '
0000:1115 11 70 69      ld      de, soft_sprite_ram+70h
0000:1118 21 21 11      ld      hl, elevator_cap_spr
0000:111B 01 10 00      ld      bc, 10h
0000:111E ED B0      ldir
0000:1120 C9      ret
0000:1120 ;
0000:1121 37 45 0F 60+elevator_cap_spr:.db 37h, 45h, 0Fh, 60h, 37h, 45h, 8Fh, 0F7h, 77h, 45h
0000:1121 37 45 8F F7+ ; DATA XREF: 0000:1118|o
0000:1121 77 45 0F 60+ .db 0Fh, 60h, 77h, 45h, 8Fh, 0F7h
0000:1131 ;
0000:1131 init_l4_rivets: ; DATA XREF: 0000:0FD5|o
0000:1131 21 F0 3D      ld      hl, fireball_rivet_spr
0000:1134 11 07 64      ld      de, unk_0_6407
0000:1137 01 1C 05      ld      bc, 51Ch
0000:113A CD 2A 12      call    sub_0_122A
0000:113D 21 14 3E      ld      hl, rivet_hammer_locs
0000:1140 CD A6 11      call    sub_0_11A6
0000:1143 21 54 3E      ld      hl, rivit_obj_spr
0000:1146 11 0C 6A      ld      de, soft_sprite_ram+10Ch
0000:1149 01 0C 00      ld      bc, 0Ch
0000:114C ED B0      ldir
0000:114E 21 82 11      ld      hl, rivet_unknown_spr2
0000:1151 11 A3 64      ld      de, unk_0_64A3
0000:1154 01 1E 02      ld      bc, 21Eh
0000:1157 CD EC 11      call    sub_0_11EC
0000:115A 21 7E 11      ld      hl, rivet_unknown_spr1
0000:115D 11 A7 64      ld      de, unk_0_64A7
0000:1160 01 1C 02      ld      bc, 21Ch
0000:1163 CD 2A 12      call    sub_0_122A
0000:1166 DD 21 A0 64      ld      ix, unk_0_64A0
0000:116A DD 36 00 01      ld      (ix+0), 1
0000:116E DD 36 20 01      ld      (ix+20h), 1
0000:1172 21 50 69      ld      hl, soft_sprite_ram+50h
0000:1175 06 02      ld      b, 2
0000:1177 11 20 00      ld      de, 20h ; ' '
0000:117A CD D3 11      call    set_B_sprites_data
0000:117D C9      ret
0000:117D ;
0000:117E 3F 0C 08 08 rivet_unknown_spr1:.db 3Fh, 0Ch, 8, 8 ; DATA XREF: 0000:115A|o
0000:117E ; Rivets level? Unknown sprites?
0000:1182 73 50 8D 50 rivet_unknown_spr2:.db 73h, 50h, 8Dh, 50h ; DATA XREF: 0000:114E|o
0000:1186 ; SUBROUTINE
0000:1186 ;
0000:1186 sub_0_1186: ; CODE XREF: 0000:102B|p
0000:1186 21 A2 11 ; 0000:1093|p
0000:1186      ld      hl, elevator_bouncing_spr
0000:1189 11 07 65      ld      de, unk_0_6507
0000:118C 01 0C 0A      ld      bc, 0A0Ch
0000:118F CD 2A 12      call    sub_0_122A
0000:1192 DD 21 00 65      ld      ix, unk_0_6500
0000:1196 21 80 69      ld      hl, soft_sprite_ram+80h
0000:1199 06 0A      ld      b, 0Ah
0000:119B 11 10 00      ld      de, 10h
0000:119E CD D3 11      call    set_B_sprites_data
0000:11A1 C9      ret
0000:11A1 ; End of function sub_0_1186
0000:11A1 ;
0000:11A1 ;
0000:11A2 3B 00 02 02 elevator_bouncing_spr:.db 3Bh, 0, 2, 2 ; DATA XREF: sub_0_1186|o
0000:11A6 ; SUBROUTINE
0000:11A6 ;
0000:11A6 sub_0_11A6: ; CODE XREF: 0000:1003|p
0000:11A6 11 83 66 ; 0000:1073|p ...
0000:11A6      ld      de, unk_0_6683
0000:11A9 01 0E 02      ld      bc, 20Eh
0000:11AC CD EC 11      call    sub_0_11EC
0000:11AF 21 08 3E      ld      hl, hammer_pickup_spr
0000:11B2 11 87 66      ld      de, unk_0_6687
0000:11B5 01 0C 02      ld      bc, 20Ch
0000:11B8 CD 2A 12      call    sub_0_122A
0000:11BB DD 21 80 66      ld      ix, unk_0_6680
0000:11BF DD 36 00 01      ld      (ix+0), 1
0000:11C3 DD 36 10 01      ld      (ix+10h), 1
0000:11C7 21 18 6A      ld      hl, soft_sprite_ram+118h ; location of hammers in sprite ram
0000:11CA 06 02      ld      b, 2
0000:11CC 11 10 00      ld      de, 10h
0000:11CF CD D3 11      call    set_B_sprites_data
0000:11D2 C9      ret
0000:11D2 ; End of function sub_0_11A6
0000:11D2 ;
0000:11D3 ; SUBROUTINE
0000:11D3 ;
0000:11D3 set_B_sprites_data: ; CODE XREF: 0000:1046|p
0000:11D3 DD 7E 03 ; 0000:10DB|p ...
0000:11D3      ld      a, (ix+3)
0000:11D6 77      ld      (hl), a ; set sprite X
0000:11D7 2C      inc     l
0000:11D8 DD 7E 07      ld      a, (ix+7)
0000:11DB 77      ld      (hl), a ; set sprite tile
0000:11DC 2C      inc     l
0000:11DD DD 7E 08      ld      a, (ix+8)
0000:11E0 77      ld      (hl), a ; set sprite vflip/palette
0000:11E1 2C      inc     l
0000:11E2 DD 7E 05      ld      a, (ix+5)

```

```

0000:11E5 77          ld      (hl), a          ; set sprite Y
0000:11E6 2C          inc      l
0000:11E7 DD 19      add      ix, de          ; next sprite data address
0000:11E9 10 E8      djnz    set_B_sprites_data
0000:11EB C9          ret
0000:11EB          ; End of function set_B_sprites_data
0000:11EB
0000:11EC          ; ██████████ SUBROUTINE ██████████
0000:11EC
0000:11EC          sub_0_11EC:          ; CODE XREF: 0000:10C0|p
0000:11EC 7E          ; 0000:1157|p ...
0000:11EC          ld      a, (hl)
0000:11ED 12          ld      (de), a
0000:11EE 23          inc      hl
0000:11EF 1C          inc      e
0000:11F0 1C          inc      e
0000:11F1 7E          ld      a, (hl)
0000:11F2 12          ld      (de), a
0000:11F3 23          inc      hl
0000:11F4 7B          ld      a, e
0000:11F5 81          add      a, c
0000:11F6 5F          ld      e, a
0000:11F7 10 F3      djnz    sub_0_11EC
0000:11F9 C9          ret
0000:11F9          ; End of function sub_0_11EC
0000:11F9
0000:11FA          ; ██████████ SUBROUTINE ██████████
0000:11FA
0000:11FA          sub_0_11FA:          ; CODE XREF: 0000:0FF2|p
0000:11FA DD 21 A0 66      ; 0000:104C|p
0000:11FA          ld      ix, unk_0_66A0
0000:11FE 11 28 6A      ld      de, soft_sprite_ram+128h
0000:1201 DD 36 00 01      ld      (ix+0), l
0000:1205 7E          ld      a, (hl)
0000:1206 DD 77 03      ld      (ix+3), a
0000:1209 12          ld      (de), a
0000:120A 1C          inc      e
0000:120B 23          inc      hl
0000:120C 7E          ld      a, (hl)
0000:120D DD 77 07      ld      (ix+7), a
0000:1210 12          ld      (de), a
0000:1211 1C          inc      e
0000:1212 23          inc      hl
0000:1213 7E          ld      a, (hl)
0000:1214 DD 77 08      ld      (ix+8), a
0000:1217 12          ld      (de), a
0000:1218 1C          inc      e
0000:1219 23          inc      hl
0000:121A 7E          ld      a, (hl)
0000:121B DD 77 05      ld      (ix+5), a
0000:121E 12          ld      (de), a
0000:121F 23          inc      hl
0000:1220 7E          ld      a, (hl)
0000:1221 DD 77 09      ld      (ix+9), a
0000:1224 23          inc      hl
0000:1225 7E          ld      a, (hl)
0000:1226 DD 77 0A      ld      (ix+0Ah), a
0000:1229 C9          ret
0000:1229          ; End of function sub_0_11FA
0000:1229
0000:122A          ; ██████████ SUBROUTINE ██████████
0000:122A
0000:122A          sub_0_122A:          ; CODE XREF: 0000:0FEC|p
0000:122A E5          ; 0000:100F|p ...
0000:122A          push    hl
0000:122B C5          push    bc
0000:122C 06 04      ld      b, 4
0000:122E          loc_0_122E:          ; CODE XREF: sub_0_122A+8|j
0000:122E          ld      a, (hl)
0000:122F 12          ld      (de), a
0000:1230 23          inc      hl
0000:1231 1C          inc      e
0000:1232 10 FA      djnz    loc_0_122E
0000:1234 C1          pop      bc
0000:1235 E1          pop      hl
0000:1236 7B          ld      a, e
0000:1237 81          add      a, c
0000:1238 5F          ld      e, a
0000:1239 10 EF      djnz    sub_0_122A
0000:123B C9          ret
0000:123B          ; End of function sub_0_122A
0000:123B
0000:123C          ; -----
0000:123C          init_mario:          ; DATA XREF: 0000:0718|o
0000:123C DF          ; 0000:074C|o
0000:123C          rst      18h          ; wait for 8-bit countdown
0000:123D 3A 27 62      ld      a, (level_type)
0000:1240 FE 03      cp      3          ; elevators?
0000:1242 01 16 E0      ld      bc, 0E016h   ; mario x,y coords
0000:1245 CA 4B 12      jp      z, loc_0_124B ; yes, skip
0000:1248 01 3F F0      ld      bc, 0F03Fh   ; mario x,y coords
0000:124B          loc_0_124B:          ; CODE XREF: 0000:1245|j
0000:124B          ld      ix, mario_alive_flag
0000:124F 21 4C 69      ld      hl, soft_sprite_ram+4Ch
0000:1252 DD 36 00 01      ld      (ix+0), l
0000:1256 DD 71 03      ld      (ix+3), c
0000:1259 71          ld      (hl), c
0000:125A 2C          inc      l
0000:125B DD 36 07 80      ld      (ix+7), 80h ; 'C'
0000:125F 36 80      ld      (hl), 80h ; 'C'
0000:1261 2C          inc      l
0000:1262 DD 36 08 02      ld      (ix+8), 2
0000:1266 36 02      ld      (hl), 2
0000:1268 2C          inc      l
0000:1269 DD 70 05      ld      (ix+5), b
0000:126C 70          ld      (hl), b
0000:126D DD 36 0F 01      ld      (ix+0Fh), 1

```

```

0000:1271 21 0A 60      ld      hl, main_sequencer
0000:1274 34          inc      (hl)
0000:1275 11 01 06      ld      de, loc_0_601
0000:1278 CD 9F 30      call    sub_0_309F
0000:127B C9          ret
;
0000:127C
0000:127C      died_in_gameplay:
0000:127C      call    sub_0_1DBD
0000:127C      ld      a, (unk_0_639D)
0000:127F 3A 9D 63      rst      28h
0000:1282 EF          ; go!
;
0000:1282
0000:1283 8B 12      .dw delay_before_spin
0000:1285 AC 12      .dw mario_death_spin
0000:1287 DE 12      .dw dead_mario_lying_down
0000:1289 00 00      .dw 0
;
0000:128B
0000:128B      delay_before_spin:
0000:128B      rst      18h
0000:128B      ld      hl, soft_sprite_ram+4Dh
0000:128C 21 4D 69      ld      a, 0F0h ; '-'
0000:128F 3E F0      rl      (hl)
0000:1291 CB 16      rra
0000:1293 1F      ld      (hl), a
0000:1294 77      ld      hl, unk_0_639D
0000:1295 21 9D 63      inc      (hl)
0000:1298 34      ld      a, 0Dh
0000:1299 3E 0D      ld      (unk_0_639E), a
0000:129B 32 9E 63      ld      a, 8
0000:129E 3E 08      ld      (eight_bit_countdown), a
0000:12A0 32 09 60      call    sub_0_30BD
0000:12A3 CD BD 30      ld      a, 3
0000:12A6 3E 03      ld      (unk_0_6088), a
0000:12A8 32 88 60      ret
0000:12AB C9
;
0000:12AC
0000:12AC      mario_death_spin:
0000:12AC      rst      18h
0000:12AC      ld      a, 8
0000:12AD 3E 08      ld      (eight_bit_countdown), a
0000:12AF 32 09 60      ld      hl, unk_0_639E
0000:12B2 21 9E 63      dec      (hl)
0000:12B5 35      jp      z, loc_0_12CB
0000:12B6 CA CB 12      ld      hl, soft_sprite_ram+4Dh
0000:12B9 21 4D 69      ld      a, (hl)
0000:12BC 7E      rra
0000:12BD 1F      ld      a, 2
0000:12BE 3E 02      ld      b, a
0000:12C0 1F      xor      (hl)
0000:12C1 47      ld      (hl), a
0000:12C2 AE      inc      l
0000:12C3 77      ld      a, b
0000:12C4 2C      and      80h ; 'Ç'
0000:12C5 78      xor      (hl)
0000:12C6 E6 80      ld      (hl), a
0000:12C8 AE      ; flip mario upside down?
0000:12C9 77      ret
0000:12CA C9
;
0000:12CB
0000:12CB      loc_0_12CB:
0000:12CB      ld      hl, soft_sprite_ram+4Dh
0000:12CE 3E F4      ld      a, 0F4h ; 'T'
0000:12D0 CB 16      rl      (hl)
0000:12D2 1F      rra
0000:12D3 77      ld      (hl), a
0000:12D4 21 9D 63      ld      hl, unk_0_639D
0000:12D7 34      inc      (hl)
0000:12D8 3E 80      ld      a, 80h ; 'Ç'
0000:12DA 32 09 60      ld      (eight_bit_countdown), a
0000:12DD C9      ret
;
0000:12DE
0000:12DE      dead_mario_lying_down:
0000:12DE      rst      18h
0000:12DE      call    sub_0_30DB
0000:12DE      ld      hl, main_sequencer
0000:12DF CD DB 30      ld      a, (current_player_E)
0000:12E2 21 0A 60      and      a
0000:12E5 3A 9E 60      jp      z, loc_0_12ED
0000:12E8 A7      inc      (hl)
0000:12E9 CA ED 12
0000:12EC 34
;
0000:12ED
0000:12ED      loc_0_12ED:
0000:12ED      inc      (hl)
0000:12ED      dec      hl
0000:12EE 2B      ld      (hl), 1
0000:12EF 36 01      ret
;
0000:12F2
0000:12F2      copy_current_lvl_data_to_P1:
0000:12F2      call    stop_sound
0000:12F2      xor      a
0000:12F5 AF      ld      (unk_0_622C), a
0000:12F6 32 2C 62      ld      hl, lives_left
0000:12F9 21 28 62      dec      (hl)
0000:12FC 35      ld      a, (hl)
0000:12FD 7E      ld      de, p1_level_data
0000:12FE 11 40 60      ld      bc, 8
0000:1301 01 08 00      ldir
0000:1304 ED B0      and      a
0000:1306 A7
;
0000:1307
0000:1307      loc_0_1307:
0000:1307      jp      nz, loc_0_1334
0000:130A 3E 01      ld      a, 1
0000:130C 21 B2 60      ld      hl, p1_score
0000:130F CD CA 13      call    sub_0_13CA
0000:1312 21 D4 76      ld      hl, VRAM_start+2D4h
0000:1315 3A 0F 60      ld      a, (two_players)
0000:1318 A7      and      a
0000:1319 28 07      jr      z, loc_0_1322
0000:131B 11 02 03      ld      de, 302h
0000:131E CD 9F 30      call    sub_0_309F
0000:1321 2B      dec      hl
0000:1322
;
0000:1322      loc_0_1322:
0000:1322
; CODE XREF: 0000:1319|j

```

```

0000:1322 CD 26 18      call    clear_14x5_HL
0000:1325 11 00 03      ld      de, 300h
0000:1328 CD 9F 30      call    sub_0_309F
0000:132B 21 09 60      ld      hl, eight_bit_countdown
0000:132E 36 C0          ld      (hl), 0C0h ; 'L'
0000:1330 23          inc     hl
0000:1331 36 10          ld      (hl), 10h
0000:1333 C9          ret
0000:1334
0000:1334
0000:1334      loc_0_1334:                                ; CODE XREF: 0000:1307|j
0000:1334      ld      c, 8
0000:1336 3A 0F 60      ld      a, (two_players)
0000:1339 A7          and     a
0000:133A CA 3F 13      jp      z, loc_0_133F
0000:133D 0E 17          ld      c, 17h
0000:133F
0000:133F      loc_0_133F:                                ; CODE XREF: 0000:133A|j
0000:133F 79          ld      a, c
0000:1340 32 0A 60      ld      (main_sequencer), a
0000:1343 C9          ret
0000:1344
0000:1344
0000:1344      copy_current_lvl_data_to_P2:                ; DATA XREF: 0000:0720|o
0000:1344 CD 1C 01      call    stop_sound
0000:1347 AF          xor     a
0000:1348 32 2C 62      ld      (unk_0_622C), a
0000:134B 21 28 62      ld      hl, lives_left
0000:134E 35          dec     (hl)
0000:134F 7E          ld      a, (hl)
0000:1350 11 48 60      ld      de, p2_level_data
0000:1353 01 08 00      ld      bc, 8
0000:1356 ED B0          ldir
0000:1358 A7          and     a
0000:1359 C2 7F 13      jp      nz, loc_0_137F
0000:135C 3E 03          ld      a, 3
0000:135E 21 B5 60      ld      hl, p2_score
0000:1361 CD CA 13      call    sub_0_13CA
0000:1364 11 03 03      ld      de, 303h
0000:1367 CD 9F 30      call    sub_0_309F
0000:136A 11 00 03      ld      de, 300h
0000:136D CD 9F 30      call    sub_0_309F
0000:1370 21 D3 76      ld      hl, VRAM_start+2D3h
0000:1373 CD 26 18      call    clear_14x5_HL
0000:1376 21 09 60      ld      hl, eight_bit_countdown
0000:1379 36 C0          ld      (hl), 0C0h ; 'L'
0000:137B 23          inc     hl
0000:137C 36 11          ld      (hl), 11h
0000:137E C9          ret
0000:137F
0000:137F
0000:137F      loc_0_137F:                                ; CODE XREF: 0000:1359|j
0000:137F      ld      c, 17h                                ; set to switch players?
0000:1381 3A 40 60      ld      a, (p1_level_data)
0000:1384 A7          and     a
0000:1385 C2 8A 13      jp      nz, loc_0_138A
0000:1388 0E 08          ld      c, 8
0000:138A
0000:138A      loc_0_138A:                                ; CODE XREF: 0000:1385|j
0000:138A      ld      a, c
0000:138B 32 0A 60      ld      (main_sequencer), a
0000:138E C9          ret
0000:138F
0000:138F
0000:138F      p1_game_over:                            ; DATA XREF: 0000:0722|o
0000:138F DF          rst     18h                                ; wait for 8-bit countdown
0000:1390 0E 17          ld      c, 17h
0000:1392 3A 48 60      ld      a, (p2_level_data)
0000:1395
0000:1395      loc_0_1395:                                ; CODE XREF: 0000:13A7|j
0000:1395      inc     (hl)
0000:1396 A7          and     a
0000:1397 C2 9C 13      jp      nz, loc_0_139C
0000:139A 0E 14          ld      c, 14h
0000:139C
0000:139C      loc_0_139C:                                ; CODE XREF: 0000:1397|j
0000:139C      ld      a, c
0000:139D 32 0A 60      ld      (main_sequencer), a
0000:13A0 C9          ret
0000:13A1
0000:13A1
0000:13A1      p2_game_over:                            ; DATA XREF: 0000:0724|o
0000:13A1 DF          rst     18h                                ; wait for 8-bit countdown
0000:13A2 0E 17          ld      c, 17h
0000:13A4 3A 40 60      ld      a, (p1_level_data)
0000:13A7 C3 95 13      jp      loc_0_1395
0000:13AA
0000:13AA
0000:13AA      set_flip_and_current_P2:                    ; DATA XREF: 0000:0726|o
0000:13AA      ld      a, (upright)
0000:13AD 32 82 7D      ld      (flipscreen), a
0000:13B0 AF          xor     a
0000:13B1 32 0A 60      ld      (main_sequencer), a
0000:13B4 21 01 01      ld      hl, 101h
0000:13B7 22 0D 60      ld      (current_player_D), hl
0000:13BA C9          ret
0000:13BB
0000:13BB
0000:13BB      set_flip_and_current_P1:                    ; DATA XREF: 0000:0728|o
0000:13BB AF          xor     a
0000:13BC 32 0D 60      ld      (current_player_D), a
0000:13BF 32 0E 60      ld      (current_player_E), a
0000:13C2 32 0A 60      ld      (main_sequencer), a
0000:13C5 3C          inc     a
0000:13C6 32 82 7D      ld      (flipscreen), a
0000:13C9 C9          ret
0000:13CA
0000:13CA
0000:13CA
0000:13CA      sub_0_13CA:                                ; CODE XREF: 0000:130F|p
0000:13CA      ld      de, unk_0_61C6
0000:13CD 12          ld      (de), a
0000:13CE CF          rst     8
0000:13CF 13          inc     de                                ; return if attract mode

```

```

0000:13D0 01 03 00      ld      bc, 3
0000:13D3 ED B0        ldir
0000:13D5 06 03        ld      b, 3
0000:13D7 21 B1 61      ld      hl, unk_0_61B1
0000:13DA
0000:13DA              loc_0_13DA:                                ; CODE XREF: sub_0_13CA+1F|j
0000:13DA 1B              dec      de
0000:13DB 1A              ld      a, (de)
0000:13DC 0F              rrca
0000:13DD 0F              rrca
0000:13DE 0F              rrca
0000:13DF 0F              rrca
0000:13E0 E6 0F      and     0Fh
0000:13E2 77              ld      (hl), a
0000:13E3 23              inc     hl
0000:13E4 1A              ld      a, (de)
0000:13E5 E6 0F      and     0Fh
0000:13E7 77              ld      (hl), a
0000:13E8 23              inc     hl
0000:13E9 10 EF      djnz    loc_0_13DA
0000:13EB 06 0E      ld      b, 0Eh
0000:13ED
0000:13ED              loc_0_13ED:                                ; CODE XREF: sub_0_13CA+26|j
0000:13ED 36 10      ld      (hl), 10h
0000:13EF 23              inc     hl
0000:13F0 10 FB      djnz    loc_0_13ED
0000:13F2 36 3F      ld      (hl), 3Fh ; '?'
0000:13F4 06 05      ld      b, 5
0000:13F6 21 A5 61      ld      hl, high_score_tbl_ram+0A5h
0000:13F9 11 C7 61      ld      de, unk_0_61C7
0000:13FC
0000:13FC              loc_0_13FC:                                ; CODE XREF: sub_0_13CA+51|j
0000:13FC 1A              ld      a, (de)
0000:13FD 96              sub     (hl)
0000:13FE 23              inc     hl
0000:13FF 13              inc     de
0000:1400 1A              ld      a, (de)
0000:1401 9E              sbc     a, (hl)
0000:1402 23              inc     hl
0000:1403 13              inc     de
0000:1404 1A              ld      a, (de)
0000:1405 9E              sbc     a, (hl)
0000:1406 D8              ret     c
0000:1407 C5              push    bc
0000:1408 06 19      ld      b, 19h
0000:140A
0000:140A              loc_0_140A:                                ; CODE XREF: sub_0_13CA+47|j
0000:140A 4E              ld      c, (hl)
0000:140B 1A              ld      a, (de)
0000:140C 77              ld      (hl), a
0000:140D 79              ld      a, c
0000:140E 12              ld      (de), a
0000:140F 2B              dec     hl
0000:1410 1B              dec     de
0000:1411 10 F7      djnz    loc_0_140A
0000:1413 01 F5 FF      ld      bc, 0FFF5h
0000:1416 09              add     hl, bc
0000:1417 EB              ex      de, hl
0000:1418 09              add     hl, bc
0000:1419 EB              ex      de, hl
0000:141A C1              pop     bc
0000:141B 10 DF      djnz    loc_0_13FC
0000:141D C9              ret
; End of function sub_0_13CA
0000:141D
0000:141D
0000:141E
0000:141E
0000:141E
0000:141E
0000:141E              draw_name_registered:                        ; DATA XREF: 0000:072A|o
0000:141E CD 16 06      call    display_credits
0000:1421 DF              rst     18h                                ; wait for 8-bit countdown
0000:1422 CD 74 08      call    clear_visible_area_and_sprites
0000:1425 3E 00      ld      a, 0
0000:1427 32 0E 60      ld      (current_player_E), a            ; player 1
0000:142A 32 0D 60      ld      (current_player_D), a            ; player 1
0000:142D 21 1C 61      ld      hl, high_score_tbl_ram+1Ch
0000:1430 11 22 00      ld      de, 22h ; ''
0000:1433 06 05      ld      b, 5
0000:1435 3E 01      ld      a, 1
0000:1437
0000:1437              loc_0_1437:                                ; CODE XREF: 0000:143C|j
0000:1437 BE              cp      (hl)
0000:1438 CA 59 14      jp      z, loc_0_1459
0000:143B 19              add     hl, de
0000:143C 10 F9      djnz    loc_0_1437
0000:143E 21 1C 61      ld      hl, high_score_tbl_ram+1Ch
0000:1441 06 05      ld      b, 5
0000:1443 3E 03      ld      a, 3
0000:1445
0000:1445              loc_0_1445:                                ; CODE XREF: 0000:144A|j
0000:1445 BE              cp      (hl)
0000:1446 CA 4F 14      jp      z, loc_0_144F
0000:1449 19              add     hl, de
0000:144A 10 F9      djnz    loc_0_1445
0000:144C C3 75 14      jp      loc_0_1475
0000:144F
0000:144F
0000:144F              loc_0_144F:                                ; CODE XREF: 0000:1446|j
0000:144F 3E 01      ld      a, 1
0000:1451 32 0E 60      ld      (current_player_E), a            ; player 2
0000:1454 32 0D 60      ld      (current_player_D), a            ; player 2
0000:1457 3E 00      ld      a, 0
0000:1459
0000:1459              loc_0_1459:                                ; CODE XREF: 0000:1438|j
0000:1459 21 26 60      ld      hl, upright
0000:145C B6              or      (hl)
0000:145D 32 82 7D      ld      (flipscreen), a
0000:1460 3E 00      ld      a, 0
0000:1462 32 09 60      ld      (eight_bit_countdown), a
0000:1465 21 0A 60      ld      hl, main_sequencer
0000:1468 34              inc     (hl)
0000:1469 11 0D 03      ld      de, 30Dh
0000:146C 06 0C      ld      b, 0Ch
0000:146E
0000:146E              loc_0_146E:                                ; CODE XREF: 0000:1472|j
0000:146E CD 9F 30      call    sub_0_309F
0000:1471 13              inc     de

```

```

0000:1472 10 FA      djnz      loc_0_146E
0000:1474 C9          ret
0000:1475          ;
0000:1475          ;
0000:1475          loc_0_1475:                                ; CODE XREF: 0000:144C↑j
0000:1475 3E 01          ld        a, 1
0000:1477 32 82 7D      ld        (flipscreen), a
0000:147A 32 05 60      ld        (nmi_sequencer), a
0000:147D 32 07 60      ld        (attract_mode_flag), a          ; set attract mode flag
0000:1480 3E 00          ld        a, 0
0000:1482 32 0A 60      ld        (main_sequencer), a
0000:1485 C9          ret
0000:1486          ;
0000:1486          ;
0000:1486          do_initials_entry:                                ; DATA XREF: 0000:072C↑o
0000:1486 CD 16 06      call     display_credits
0000:1489 21 09 60      ld        hl, eight_bit_countdown
0000:148C 7E          ld        a, (hl)
0000:148D A7          and      a
0000:148E C2 DC 14      jp      nz, loc_0_14DC
0000:1491 32 86 7D      ld        (palette_bank), a
0000:1494 32 87 7D      ld        (palette_bank+1), a
0000:1497 36 01          ld        (hl), 1
0000:1499 21 30 60      ld        hl, unk_0_6030
0000:149C 36 0A          ld        (hl), 0Ah
0000:149E 23          inc      hl
0000:149F 36 00          ld        (hl), 0
0000:14A1 23          inc      hl
0000:14A2 36 10          ld        (hl), 10h
0000:14A4 23          inc      hl
0000:14A5 36 1E          ld        (hl), 1Eh
0000:14A7 23          inc      hl
0000:14A8 36 3E          ld        (hl), 3Eh ; '>'
0000:14AA 23          inc      hl
0000:14AB 36 00          ld        (hl), 0
0000:14AD 21 E8 75      ld        hl, VRAM_start+1E8h
0000:14B0 22 36 60      ld        (unk_0_6036), hl
0000:14B3 21 1C 61      ld        hl, high_score_tbl_ram+1Ch
0000:14B6 3A 0E 60      ld        a, (current_player_E)          ; 0/1
0000:14B9 07          rlca                                     ; 0/2
0000:14BA 3C          inc      a                              ; 1/3
0000:14BB 4F          ld        c, a
0000:14BC 11 22 00      ld        de, 22h ; ''
0000:14BF 06 04          ld        b, 4
0000:14C1          ;
0000:14C1          loc_0_14C1:                                ; CODE XREF: 0000:14C7↑j
0000:14C1 7E          ld        a, (hl)
0000:14C2 B9          cp        c
0000:14C3 CA C9 14      jp      z, loc_0_14C9
0000:14C6 19          add      hl, de
0000:14C7 10 F8      djnz      loc_0_14C1
0000:14C9          ;
0000:14C9          loc_0_14C9:                                ; CODE XREF: 0000:14C3↑j
0000:14C9 22 38 60      ld        (unk_0_6038), hl              ; point to high score
0000:14CC 11 F3 FF      ld        de, 0FFF3h
0000:14CF 19          add      hl, de
0000:14D0 22 3A 60      ld        (unk_0_603A), hl
0000:14D3 06 00          ld        b, 0
0000:14D5 3A 35 60      ld        a, (unk_0_6035)
0000:14D8 4F          ld        c, a
0000:14D9 CD FA 15      call     sub_0_15FA                    ; high score initial select sprite
0000:14DC          ;
0000:14DC          loc_0_14DC:                                ; CODE XREF: 0000:148E↑j
0000:14DC 21 34 60      ld        hl, unk_0_6034
0000:14DF 35          dec      (hl)
0000:14E0 C2 FC 14      jp      nz, loc_0_14FC
0000:14E3 36 3E          ld        (hl), 3Eh ; '>'
0000:14E5 2B          dec      hl
0000:14E6 35          dec      (hl)
0000:14E7 CA C6 15      jp      z, loc_0_15C6
0000:14EA 7E          ld        a, (hl)
0000:14EB 06 FF      ld        b, 0FFh
0000:14ED          ;
0000:14ED          loc_0_14ED:                                ; CODE XREF: 0000:14F0↑j
0000:14ED 04          inc      b
0000:14EE D6 0A          sub      0Ah
0000:14F0 D2 ED 14      jp      nc, loc_0_14ED
0000:14F3 C6 0A          add      a, 0Ah
0000:14F5 32 52 75      ld        (VRAM_start+152h), a
0000:14F8 78          ld        a, b
0000:14F9 32 72 75      ld        (VRAM_start+172h), a
0000:14FC          ;
0000:14FC          loc_0_14FC:                                ; CODE XREF: 0000:14E0↑j
0000:14FC 21 30 60      ld        hl, unk_0_6030
0000:14FF 46          ld        b, (hl)
0000:1500 36 0A          ld        (hl), 0Ah
0000:1502 3A 10 60      ld        a, (controller_in)
0000:1505 CB 7F          bit      7, a                          ; edge-detected inputs
0000:1507 C2 46 15      jp      nz, jump_pressed              ; button pressed?
0000:150A E6 03          and      3                             ; yes, skip
0000:150C C2 14 15      jp      nz, left_right_pressed        ; left/right only
0000:150F 3C          inc      a                              ; yes, skip
0000:1510 77          ld        (hl), a
0000:1511 C3 8A 15      jp      loc_0_158A
0000:1514          ;
0000:1514          ;
0000:1514          left_right_pressed:                            ; CODE XREF: 0000:150C↑j
0000:1514 05          dec      b
0000:1515 CA 1D 15      jp      z, loc_0_151D
0000:1518 78          ld        a, b
0000:1519 77          ld        (hl), a
0000:151A C3 8A 15      jp      loc_0_158A
0000:151D          ;
0000:151D          ;
0000:151D          loc_0_151D:                                ; CODE XREF: 0000:1515↑j
0000:151D CB 4F          bit      1, a
0000:151F C2 39 15      jp      nz, loc_0_1539
0000:1522 3A 35 60      ld        a, (unk_0_6035)
0000:1525 3C          inc      a
0000:1526 FE 1E          cp      1Eh
0000:1528 C2 2D 15      jp      nz, loc_0_152D
0000:152B 3E 00          ld        a, 0
0000:152D          ;
0000:152D          loc_0_152D:                                ; CODE XREF: 0000:1528↑j
0000:152D 32 35 60      ld        (unk_0_6035), a              ; 0000:153E↑j ...
0000:152D          ;

```



```

0000:1530 4F          ld      c, a
0000:1531 06 00          ld      b, 0
0000:1533 CD FA 15      call   sub_0_15FA
0000:1536 C3 8A 15      jp      loc_0_158A
0000:1539          ;
0000:1539          loc_0_1539:                                ; CODE XREF: 0000:151F|j
0000:1539 3A 35 60          ld      a, (unk_0_6035)
0000:153C D6 01          sub     1
0000:153E F2 2D 15          jp      p, loc_0_152D
0000:1541 3E 1D          ld      a, 1Dh
0000:1543 C3 2D 15          jp      loc_0_152D
0000:1546          ;
0000:1546          jump_pressed:                                ; CODE XREF: 0000:1507|j
0000:1546 3A 35 60          ld      a, (unk_0_6035)
0000:1549 FE 1C          cp      1Ch
0000:154B CA 6D 15          jp      z, loc_0_156D
0000:154E FE 1D          cp      1Dh
0000:1550 CA C6 15          jp      z, loc_0_15C6
0000:1553 2A 36 60          ld      hl, (unk_0_6036)
0000:1556 01 88 75          ld      bc, VRAM_start+188h
0000:1559 A7          and     a
0000:155A ED 42          sbc     hl, bc
0000:155C CA 8A 15          jp      z, loc_0_158A
0000:155F 09          add     hl, bc
0000:1560 C6 11          add     a, 11h
0000:1562 77          ld      (hl), a
0000:1563 01 E0 FF          ld      bc, 0FFE0h
0000:1566 09          add     hl, bc
0000:1567          loc_0_1567:                                ; CODE XREF: 0000:1583|j
0000:1567 22 36 60          ld      (unk_0_6036), hl
0000:156A C3 8A 15          jp      loc_0_158A
0000:156D          ;
0000:156D          loc_0_156D:                                ; CODE XREF: 0000:154B|j
0000:156D 2A 36 60          ld      hl, (unk_0_6036)
0000:1570 01 20 00          ld      bc, 20h ; ' '
0000:1573 09          add     hl, bc
0000:1574 A7          and     a
0000:1575 01 08 76          ld      bc, VRAM_start+208h
0000:1578 ED 42          sbc     hl, bc
0000:157A C2 86 15          jp      nz, loc_0_1586
0000:157D 21 E8 75          ld      hl, VRAM_start+1E8h
0000:1580          loc_0_1580:                                ; CODE XREF: 0000:1587|j
0000:1580 3E 10          ld      a, 10h
0000:1582 77          ld      (hl), a
0000:1583 C3 67 15          jp      loc_0_1567
0000:1586          ;
0000:1586          loc_0_1586:                                ; CODE XREF: 0000:157A|j
0000:1586 09          add     hl, bc
0000:1587 C3 80 15          jp      loc_0_1580
0000:158A          ;
0000:158A          loc_0_158A:                                ; CODE XREF: 0000:1511|j
0000:158A 21 32 60          ld      hl, unk_0_6032 ; 0000:151A|j ...
0000:158D 35          dec     (hl)
0000:158E C2 F9 15          jp      nz, locret_0_15F9
0000:1591 3A 31 60          ld      a, (unk_0_6031)
0000:1594 A7          and     a
0000:1595 C2 B8 15          jp      nz, loc_0_15B8
0000:1598 3E 01          ld      a, 1
0000:159A 32 31 60          ld      (unk_0_6031), a
0000:159D 11 BF 01          ld      de, byte_0_1BD+2 ; empty/dummy score
0000:15A0          loc_0_15A0:                                ; CODE XREF: 0000:15C3|j
0000:15A0 FD 2A 38 60          ld      iy, (unk_0_6038) ; ptr high score
0000:15A4 FD 6E 04          ld      l, (iy+4)
0000:15A7 FD 66 05          ld      h, (iy+5)
0000:15AA E5          push    hl
0000:15AB DD E1          pop     ix
0000:15AD CD 7C 05          call    display_score_HL
0000:15B0 3E 10          ld      a, 10h
0000:15B2 32 32 60          ld      (unk_0_6032), a
0000:15B5 C3 F9 15          jp      locret_0_15F9
0000:15B8          ;
0000:15B8          loc_0_15B8:                                ; CODE XREF: 0000:1595|j
0000:15B8 AF          xor     a
0000:15B9 32 31 60          ld      (unk_0_6031), a
0000:15BC ED 5B 38 60          ld      de, (unk_0_6038) ; point to high score
0000:15C0 13          inc     de
0000:15C1 13          inc     de
0000:15C2 13          inc     de
0000:15C3 C3 A0 15          jp      loc_0_15A0
0000:15C6          ;
0000:15C6          loc_0_15C6:                                ; CODE XREF: 0000:14E7|j
0000:15C6 ED 5B 38 60          ld      de, (unk_0_6038) ; 0000:1550|j
0000:15C6          ; point to high score
0000:15CA AF          xor     a
0000:15CB 12          ld      (de), a
0000:15CC 21 09 60          ld      hl, eight_bit_countdown
0000:15CF 36 80          ld      (hl), 80h ; 'C'
0000:15D1 23          inc     hl
0000:15D2 35          dec     (hl)
0000:15D3 06 0C          ld      b, 0Ch
0000:15D5 21 E8 75          ld      hl, VRAM_start+1E8h
0000:15D8 FD 2A 3A 60          ld      iy, (unk_0_603A)
0000:15DC 11 E0 FF          ld      de, 0FFE0h
0000:15DF          loc_0_15DF:                                ; CODE XREF: 0000:15E6|j
0000:15DF 7E          ld      a, (hl)
0000:15E0 FD 77 00          ld      (iy+0), a
0000:15E3 FD 23          inc     iy
0000:15E5 19          add     hl, de
0000:15E6 10 F7          djnz    loc_0_15DF
0000:15E8 06 05          ld      b, 5
0000:15EA 11 14 03          ld      de, 314h
0000:15ED          loc_0_15ED:                                ; CODE XREF: 0000:15F1|j
0000:15ED CD 9F 30          call    sub_0_309F

```

```

0000:15F0 13          inc     de
0000:15F1 10 FA          djnz   loc_0_15ED
0000:15F3 11 1A 03        ld      de, 31Ah
0000:15F6 CD 9F 30        call   sub_0_309F
0000:15F9          locret_0_15F9:          ; CODE XREF: 0000:158E|j
0000:15F9 C9          ret          ; 0000:15B5|j
0000:15FA          ; ██████████ SUBROUTINE ██████████
0000:15FA          sub_0_15FA:          ; CODE XREF: 0000:14D9|p
0000:15FA D5          ; 0000:1533|p
0000:15FA          push    de
0000:15FB E5          push    hl
0000:15FC CB 21          sla     c
0000:15FE 21 0F 36        ld      hl, letter_coords
0000:1601 09          add     hl, bc
0000:1602 EB          ex      de, hl
0000:1603 21 74 69        ld      hl, soft_sprite_ram+74h ; sprite for initials entry
0000:1606 1A          ld      a, (de)
0000:1607 13          inc     de
0000:1608 77          ld      (hl), a          ; X coordinate
0000:1609 23          inc     hl
0000:160A 36 72        ld      (hl), 72h ; 'r'
0000:160C 23          inc     hl
0000:160D 36 0C        ld      (hl), 0Ch          ; palette
0000:160F 23          inc     hl
0000:1610 1A          ld      a, (de)
0000:1611 77          ld      (hl), a          ; Y coordinate
0000:1612 E1          pop     hl
0000:1613 D1          pop     de
0000:1614 C9          ret
0000:1614          ; End of function sub_0_15FA
0000:1614          ;
0000:1615          mario_pauline_reunion: ; DATA XREF: 0000:072E|o
0000:1615 CD BD 30          call   sub_0_30BD
0000:1618 3A 27 62        ld      a, (level_type)
0000:161B 0F          rrca
0000:161C D2 2F 16        jp     nc, loc_0_162F
0000:161F 3A 88 63        ld      a, (unk_0_6388)
0000:1622 EF          rst     28h          ; go!
0000:1622          ;
0000:1623 54 16          .dw loc_0_1654          ; Jump table
0000:1625 70 16          .dw loc_0_1670
0000:1627 8A 16          .dw loc_0_168A
0000:1629 32 17          .dw loc_0_1732
0000:162B 57 17          .dw loc_0_1757
0000:162D 8E 17          .dw loc_0_178E
0000:162F          ;
0000:162F          loc_0_162F:          ; CODE XREF: 0000:161C|j
0000:162F 0F          rrca
0000:1630 D2 41 16        jp     nc, loc_0_1641
0000:1633 3A 88 63        ld      a, (unk_0_6388)
0000:1636 EF          rst     28h          ; go!
0000:1636          ;
0000:1637 A3 16          .dw loc_0_16A3          ; Jump table
0000:1639 BB 16          .dw loc_0_16BB
0000:163B 32 17          .dw loc_0_1732
0000:163D 57 17          .dw loc_0_1757
0000:163F 8E 17          .dw loc_0_178E
0000:1641          ;
0000:1641          loc_0_1641:          ; CODE XREF: 0000:1630|j
0000:1641 CD BD 1D          call   sub_0_1DBD
0000:1644 3A 88 63        ld      a, (unk_0_6388)
0000:1647 EF          rst     28h          ; go!
0000:1647          ;
0000:1648 B6 17          .dw unk_0_17B6          ; Jump table
0000:164A 69 30          .dw loc_0_3069
0000:164C 39 18          .dw loc_0_1839
0000:164E 6F 18          .dw loc_0_186F
0000:1650 80 18          .dw loc_0_1880
0000:1652 C6 18          .dw loc_0_18C6
0000:1654          ;
0000:1654          loc_0_1654:          ; DATA XREF: 0000:1623|o
0000:1654 CD 08 17          call   sub_0_1708
0000:1657 21 5C 38        ld      hl, kong_normal_spr
0000:165A CD 4E 00        call   copy_sprites_2_11_data
0000:165D 3E 20        ld      a, 20h ; ' '
0000:165F 32 09 60        ld      (eight_bit_countdown), a
0000:1662          ;
0000:1662          loc_0_1662:          ; CODE XREF: 0000:16A0|j
0000:1662 21 88 63        ld      hl, unk_0_6388
0000:1665 34          inc     (hl)
0000:1666 3E 01        ld      a, 1
0000:1668 F7          rst     30h          ; return if level bit not set
0000:1669 21 0B 69        ld      hl, soft_sprite_ram+0Bh ; sprite #2, x coord
0000:166C 0E FC        ld      c, 0FCh ; '3'
0000:166E FF          rst     38h          ; subtract 4 from x coord for 10 sprites
0000:166F C9          ret
0000:1670          ;
0000:1670          loc_0_1670:          ; DATA XREF: 0000:1625|o
0000:1670 DF          ; wait for 8-bit countdown
0000:1670          rst     18h
0000:1671 21 32 39        ld      hl, dk_throw_barrel_spr
0000:1674 CD 4E 00        call   copy_sprites_2_11_data
0000:1677 3E 20        ld      a, 20h ; ' '
0000:1679 32 09 60        ld      (eight_bit_countdown), a
0000:167C 21 88 63        ld      hl, unk_0_6388
0000:167F 34          inc     (hl)
0000:1680 3E 04        ld      a, 4
0000:1682 F7          rst     30h          ; return if level bit not set
0000:1683 21 0B 69        ld      hl, soft_sprite_ram+0Bh ; sprite #2, x coord
0000:1686 0E 04        ld      c, 4
0000:1688 FF          rst     38h          ; +4
0000:1689 C9          ret          ; add 4 to x coord for 10 sprites
0000:168A          ;
0000:168A          loc_0_168A:          ; DATA XREF: 0000:1627|o
0000:168A DF          rst     18h          ; wait for 8-bit countdown

```

```

0000:168B 21 8C 38      ld      hl, kong_climbing_spr
0000:168E CD 4E 00      call   copy_sprites_2_1l_data
0000:1691 3E 66          ld      a, 66h ; 'f'
0000:1693 32 0C 69      ld      (soft_sprite_ram+0Ch), a          ; sprite #3, y coord
0000:1696 AF            xor      a
0000:1697 32 24 69      ld      (soft_sprite_ram+24h), a
0000:169A 32 2C 69      ld      (soft_sprite_ram+2Ch), a
0000:169D 32 AF 62      ld      (unk_0_62AF), a
0000:16A0 C3 62 16      jp      loc_0_1662
0000:16A3              ;
0000:16A3
0000:16A3
loc_0_16A3:
0000:16A3 CD 08 17      call   sub_0_1708          ; DATA XREF: 0000:1637|o
0000:16A6 3A 10 69      ld      a, (soft_sprite_ram+10h)          ; sprite #4, y coord
0000:16A9 D6 3B      sub     3Bh ; ';'
0000:16AB 21 5C 38      ld      hl, kong_normal_spr
0000:16AE CD 4E 00      call   copy_sprites_2_1l_data
0000:16B1 21 08 69      ld      hl, soft_sprite_ram+8          ; sprite #2, y coord
0000:16B4 4F            ld      c, a
0000:16B5 FF            rst     38h          ; add C to y coord for 10 sprites
0000:16B6 21 88 63      ld      hl, unk_0_6388
0000:16B9 34            inc     (hl)
0000:16BA C9            ret
0000:16BB              ;
0000:16BB
0000:16BB
loc_0_16BB:
0000:16BB AF            xor      a          ; DATA XREF: 0000:1639|o
0000:16BC 32 A0 62      ld      (unk_0_62A0), a
0000:16BF 3A A3 63      ld      a, (unk_0_63A3)
0000:16C2 4F            ld      c, a
0000:16C3 3A 10 69      ld      a, (soft_sprite_ram+10h)          ; sprite #4, y coord
0000:16C6 FE 5A      cp      5Ah ; 'Z'
0000:16C8 D2 E1 16      jp      nc, loc_0_16E1
0000:16CB CB 79      bit     7, c
0000:16CD CA D5 16      jp      z, loc_0_16D5
0000:16D0
0000:16D0
loc_0_16D0:
0000:16D0 3E 01      ld      a, 1          ; CODE XREF: 0000:16E8|j
0000:16D2 32 A0 62      ld      (unk_0_62A0), a
0000:16D5
0000:16D5
loc_0_16D5:
0000:16D5 CD 02 26      call   sub_0_2602          ; CODE XREF: 0000:16CD|j
0000:16D8 3A A3 63      ld      a, (unk_0_63A3)          ; 0000:16EB|j
0000:16DB 4F            ld      c, a
0000:16DC 21 08 69      ld      hl, soft_sprite_ram+8          ; sprite #2, y coord
0000:16DF FF            rst     38h          ; add C to y coord for 10 sprites
0000:16E0 C9            ret
0000:16E1              ;
0000:16E1
loc_0_16E1:
0000:16E1 FE 5D      cp      5Dh ; ']'          ; CODE XREF: 0000:16C8|j
0000:16E3 DA EE 16      jp      c, loc_0_16EE
0000:16E6 CB 79      bit     7, c
0000:16E8 CA D0 16      jp      z, loc_0_16D0
0000:16EB C3 D5 16      jp      loc_0_16D5
0000:16EE              ;
0000:16EE
loc_0_16EE:
0000:16EE 21 8C 38      ld      hl, kong_climbing_spr          ; CODE XREF: 0000:16E3|j
0000:16F1 CD 4E 00      call   copy_sprites_2_1l_data
0000:16F4 3E 66      ld      a, 66h ; 'f'
0000:16F6 32 0C 69      ld      (soft_sprite_ram+0Ch), a          ; sprite #4, x coord
0000:16F9 AF            xor      a
0000:16FA 32 24 69      ld      (soft_sprite_ram+24h), a
0000:16FD 32 2C 69      ld      (soft_sprite_ram+2Ch), a
0000:1700 32 AF 62      ld      (unk_0_62AF), a
0000:1703 21 88 63      ld      hl, unk_0_6388
0000:1706 34            inc     (hl)
0000:1707 C9            ret
0000:1708              ;
0000:1708
; SUBROUTINE
0000:1708
sub_0_1708:
0000:1708 CD 1C 01      call   stop_sound          ; CODE XREF: 0000:1654|p
0000:1708              ; 0000:16A3|p
0000:170B 21 20 6A      ld      hl, soft_sprite_ram+120h
0000:170E 36 80      ld      (hl), 80h ; 'C'
0000:1710 23      inc     hl
0000:1711 36 76      ld      (hl), 76h ; 'v'
0000:1713 23      inc     hl
0000:1714 36 09      ld      (hl), 9
0000:1716 23      inc     hl
0000:1717 36 20      ld      (hl), 20h ; ' '
0000:1719 21 05 69      ld      hl, soft_sprite_ram+5          ; sprite #1, flipy & code
0000:171C 36 13      ld      (hl), 13h          ; pauline, front-on
0000:171E 21 C4 75      ld      hl, VRAM_start+1C4h
0000:1721 11 20 00      ld      de, 20h ; ' '
0000:1724 3E 10      ld      a, 10h
0000:1726 CD 14 05      call   display_3_tiles_HL
0000:1729 21 8A 60      ld      hl, unk_0_608A
0000:172C 36 07      ld      (hl), 7
0000:172E 23      inc     hl
0000:172F 36 03      ld      (hl), 3
0000:1731 C9            ret
0000:1731              ; End of function sub_0_1708
0000:1731
0000:1732
0000:1732
loc_0_1732:
0000:1732 CD 6F 30      call   sub_0_306F          ; DATA XREF: 0000:1629|o
0000:1732              ; 0000:163B|o
0000:1735 3A 13 69      ld      a, (soft_sprite_ram+13h)
0000:1738 FE 2C      cp      2Ch ; ','
0000:173A D0      ret     nc
0000:173B AF            xor      a
0000:173C 32 00 69      ld      (soft_sprite_ram), a          ; sprite #0, y coord
0000:173F 32 04 69      ld      (soft_sprite_ram+4), a          ; sprite #1, y coord
0000:1742 32 0C 69      ld      (soft_sprite_ram+0Ch), a          ; sprite #3, y coord
0000:1745 3E 6B      ld      a, 6Bh ; 'k'
0000:1747 32 24 69      ld      (soft_sprite_ram+24h), a
0000:174A 3D      dec     a
0000:174B 32 2C 69      ld      (soft_sprite_ram+2Ch), a
0000:174E 21 21 6A      ld      hl, soft_sprite_ram+121h
0000:1751 34      inc     (hl)

```

```

0000:1752 21 88 63      ld      hl, unk_0_6388
0000:1755 34          inc     (hl)
0000:1756 C9          ret
0000:1757          ; -----
0000:1757          loc_0_1757:                                ; DATA XREF: 0000:162B|o
0000:1757 CD 6F 30          ; 0000:163D|o
0000:1757          call    sub_0_306F
0000:175A CD 6C 17          call    sub_0_176C
0000:175D 23          inc     hl
0000:175E 13          inc     de
0000:175F CD 83 17          call    sub_0_1783
0000:1762 3E 40          ld      a, 40h ; '@'
0000:1764 32 09 60          ld      (eight_bit_countdown), a
0000:1767 21 88 63          ld      hl, unk_0_6388
0000:176A 34          inc     (hl)
0000:176B C9          ret
0000:176C          ; [REDACTED] S U B R O U T I N E [REDACTED]
0000:176C          sub_0_176C:                                ; CODE XREF: 0000:175A|p
0000:176C 11 03 00          ld      de, 3
0000:176F 21 2F 69          ld      hl, soft_sprite_ram+2Fh
0000:1772 06 0A          ld      b, 0Ah
0000:1774          loc_0_1774:                                ; CODE XREF: sub_0_176C+14|j
0000:1774 A7          and     a, (hl)
0000:1775 7E          ld      hl, de
0000:1776 ED 52          sbc     hl, de
0000:1778 FE 19          cp      19h
0000:177A D2 7F 17          jp      nc, loc_0_177F
0000:177D 36 00          ld      (hl), 0
0000:177F          loc_0_177F:                                ; CODE XREF: sub_0_176C+E|j
0000:177F 2B          dec     hl
0000:1780 10 F2          djnz    loc_0_1774
0000:1782 C9          ret
0000:1782          ; End of function sub_0_176C
0000:1782          ; [REDACTED] S U B R O U T I N E [REDACTED]
0000:1782          sub_0_1783:                                ; CODE XREF: 0000:175F|p
0000:1783 06 0A          ld      b, 0Ah
0000:1785          loc_0_1785:                                ; CODE XREF: sub_0_1783+8|j
0000:1785 7E          ld      a, (hl)
0000:1786 A7          and     a, a
0000:1787 C2 26 00          jp      nz, pop_hl_ret
0000:178A 19          add     hl, de
0000:178B 10 F8          djnz    loc_0_1785
0000:178D C9          ret
0000:178D          ; End of function sub_0_1783
0000:178E          ; -----
0000:178E          loc_0_178E:                                ; DATA XREF: 0000:162D|o
0000:178E DF          ; 0000:163F|o
0000:178E          ; wait for 8-bit countdown
0000:178F 2A 2A 62          rst     18h
0000:1792 23          ld      hl, (seq_data)
0000:1793 7E          inc     hl
0000:1794 FE 7F          ld      a, (hl)
0000:1796 C2 9D 17          cp      7Fh ; ' '
0000:1799 21 73 3A          jp      nz, loc_0_179D
0000:179C 7E          ld      hl, level_seq_2
0000:179D          ld      a, (hl)
0000:179D          loc_0_179D:                                ; CODE XREF: 0000:1796|j
0000:179D 22 2A 62          ld      (seq_data), hl
0000:17A0 32 27 62          ld      (level_type), a
0000:17A3 11 00 05          ld      de, 500h
0000:17A6 CD 9F 30          call    sub_0_309F
0000:17A9 AF          xor     a
0000:17AA 32 88 63          ld      (unk_0_6388), a
0000:17AD 21 09 60          ld      hl, eight_bit_countdown
0000:17B0 36 30          ld      (hl), 30h ; '0'
0000:17B2 23          inc     hl
0000:17B3 36 08          ld      (hl), 8
0000:17B5 C9          ret
0000:17B5          ; -----
0000:17B6 00          unk_0_17B6: .db 0 ;
0000:17B7          ; -----
0000:17B7 CD 1C 01          call    stop_sound
0000:17BA 21 8A 60          ld      hl, unk_0_608A
0000:17BD 36 0E          ld      (hl), 0Eh
0000:17BF 23          inc     hl
0000:17C0 36 03          ld      (hl), 3
0000:17C2 3E 10          ld      a, 10h
0000:17C4 11 20 00          ld      de, 20h ; ' '
0000:17C7 21 23 76          ld      hl, VRAM_start+223h
0000:17CA CD 14 05          call    display_3_tiles_HL
0000:17CD 21 83 75          ld      hl, VRAM_start+183h
0000:17D0 CD 14 05          call    display_3_tiles_HL
0000:17D3 21 DA 76          ld      hl, VRAM_start+2DAh
0000:17D6 CD 26 18          call    clear_14x5_HL
0000:17D9 11 47 3A          ld      de, draw_data_rivet_end1
0000:17DC CD A7 0D          call    sub_0_DA7
0000:17DF 21 D5 76          ld      hl, VRAM_start+2D5h
0000:17E2 CD 26 18          call    clear_14x5_HL
0000:17E5 11 4D 3A          ld      de, draw_data_rivet_end2
0000:17E8 CD A7 0D          call    sub_0_DA7
0000:17EB 21 D0 76          ld      hl, VRAM_start+2D0h
0000:17EE CD 26 18          call    clear_14x5_HL
0000:17F1 11 53 3A          ld      de, draw_data_rivet_end3
0000:17F4 CD A7 0D          call    sub_0_DA7
0000:17F7 21 CB 76          ld      hl, VRAM_start+2CBh
0000:17FA CD 26 18          call    clear_14x5_HL
0000:17FD 11 59 3A          ld      de, draw_data_rivet_end4
0000:1800 CD A7 0D          call    sub_0_DA7
0000:1803 21 5C 38          ld      hl, kong_normal_spr
0000:1806 CD 4E 00          call    copy_sprites_2_11_data
0000:1809 21 08 69          ld      hl, soft_sprite_ram+8
0000:180C 0E 44          ld      c, 68
0000:180E FF          rst     38h
0000:180E          ; add 68 to y coord for 10 sprites

```

```

0000:180F 21 05 69      ld      hl, soft_sprite_ram+5      ; sprite #1, yflip & code
0000:1812 36 13      ld      (hl), 13h                ; pauline, straight-on
0000:1814 3E 20      ld      a, 20h ; ' '
0000:1816 32 09 60      ld      (eight_bit_countdown), a
0000:1819 3E 80      ld      a, 80h ; 'G'
0000:181B 32 90 63      ld      (kong_thrash_tmr), a
0000:181E 21 88 63      ld      hl, unk_0_6388
0000:1821 34      inc      (hl)
0000:1822 22 C0 63      ld      (unk_0_63C0), hl
0000:1825 C9      ret
0000:1826
0000:1826      ; ██████████ SUBROUTINE ██████████
0000:1826
0000:1826      clear_14x5_HL:                  ; CODE XREF: 0000:1322|p
0000:1826 11 DB FF      ; 0000:1373|p ...
0000:1826      ld      de, 0FFDBh
0000:1829 0E 0E      ld      c, 0Eh
0000:182B 3E 10      ld      a, 10h
0000:182D      ; <space>
0000:182D      loc_0_182D:                    ; CODE XREF: clear_14x5_HL+F|j
0000:182D 06 05      ld      b, 5
0000:182F      loc_0_182F:                    ; CODE XREF: clear_14x5_HL+B|j
0000:182F      ld      (hl), a
0000:1830 23      inc      hl
0000:1831 10 FC      djnz    loc_0_182F
0000:1833 19      add      hl, de
0000:1834 0D      dec      c
0000:1835 C2 2D 18      jp      nz, loc_0_182D
0000:1838 C9      ret
0000:1838      ; End of function clear_14x5_HL
0000:1839
0000:1839      ;
0000:1839      loc_0_1839:                    ; DATA XREF: 0000:164C|o
0000:1839 21 90 63      ld      hl, kong_thrash_tmr
0000:183C 34      inc      (hl)
0000:183D CA 59 18      jp      z, loc_0_1859
0000:1840 7E      ld      a, (hl)
0000:1841 E6 07      and     7
0000:1843 C0      ret      nz
0000:1844 11 CF 39      ld      de, 39CFh
0000:1847 CB 5E      bit     3, (hl)
0000:1849 20 03      jr      nz, loc_0_184E
0000:184B 11 F7 39      ld      de, 39F7h
0000:184E      loc_0_184E:                    ; CODE XREF: 0000:1849|j
0000:184E EB      ex      de, hl
0000:184F CD 4E 00      call    copy_sprites_2_11_data
0000:1852 21 08 69      ld      hl, soft_sprite_ram+8
0000:1855 0E 44      ld      c, 68
0000:1857 FF      rst     38h
0000:1858 C9      ret
0000:1859      ;
0000:1859      loc_0_1859:                    ; CODE XREF: 0000:183D|j
0000:1859      ld      hl, kong_normal_spr
0000:185C CD 4E 00      call    copy_sprites_2_11_data
0000:185F 21 08 69      ld      hl, soft_sprite_ram+8
0000:1862 0E 44      ld      c, 68
0000:1864 FF      rst     38h
0000:1865 3E 20      ld      a, 20h ; ' '
0000:1867 32 09 60      ld      (eight_bit_countdown), a
0000:186A 21 88 63      ld      hl, unk_0_6388
0000:186D 34      inc      (hl)
0000:186E C9      ret
0000:186F      ;
0000:186F      loc_0_186F:                    ; DATA XREF: 0000:164E|o
0000:186F DF      rst     18h
0000:1870 21 1F 3A      ld      hl, fk_falling_spr
0000:1873 CD 4E 00      call    copy_sprites_2_11_data
0000:1876 3E 03      ld      a, 3
0000:1878 32 84 60      ld      (digital_snd_tmr_kong_fall), a
0000:187B 21 88 63      ld      hl, unk_0_6388
0000:187E 34      inc      (hl)
0000:187F C9      ret
0000:1880      ;
0000:1880      loc_0_1880:                    ; DATA XREF: 0000:1650|o
0000:1880      ld      hl, soft_sprite_ram+0Bh
0000:1883 0E 01      ld      c, 1
0000:1885 FF      rst     38h
0000:1886 3A 1B 69      ld      a, (soft_sprite_ram+1Bh)
0000:1889 FE D0      cp      0D0h ; '0'
0000:188B C0      ret      nz
0000:188C 3E 20      ld      a, 20h ; ' '
0000:188E 32 19 69      ld      (soft_sprite_ram+19h), a
0000:1891 21 24 6A      ld      hl, soft_sprite_ram+124h
0000:1894 36 7F      ld      (hl), 7Fh ; ' '
0000:1896 2C      inc      l
0000:1897 36 39      ld      (hl), 39h ; '9'
0000:1899 2C      inc      l
0000:189A 36 01      ld      (hl), 1
0000:189C 2C      inc      l
0000:189D 36 D8      ld      (hl), 0D8h ; 'I'
0000:189F 21 C6 76      ld      hl, VRAM_start+2C6h
0000:18A2 CD 26 18      call    clear_14x5_HL
0000:18A5 11 5F 3A      ld      de, draw_data_rivet_end5
0000:18A8 CD A7 0D      call    sub_0_DA7
0000:18AB 11 04 00      ld      de, 4
0000:18AE 01 28 02      ld      bc, 228h
0000:18B1 21 03 69      ld      hl, soft_sprite_ram+3
0000:18B4 CD 3D 00      call    loc_0_3D
0000:18B7 3E 00      ld      a, 0
0000:18B9 32 AF 62      ld      (unk_0_62AF), a
0000:18BC 3E 03      ld      a, 3
0000:18BE 32 82 60      ld      (digital_snd_tmr_thump), a
0000:18C1 21 88 63      ld      hl, unk_0_6388
0000:18C4 34      inc      (hl)
0000:18C5 C9      ret
0000:18C6      ;
0000:18C6      loc_0_18C6:                    ; DATA XREF: 0000:1652|o
0000:18C6 21 AF 62      ld      hl, unk_0_62AF

```

```

0000:18C9 35      dec      (hl)
0000:18CA CA 3D 19  jp      z, loc_0_193D
0000:18CD 7E      ld      a, (hl)
0000:18CE E6 07      and     7
0000:18D0 C0      ret     nz
0000:18D1 21 25 6A     ld      hl, soft_sprite_ram+125h
0000:18D4 7E      ld      a, (hl)
0000:18D5 EE 80     xor     80h ; 'Ç'
0000:18D7 77      ld      (hl), a
0000:18D8 21 19 69     ld      hl, soft_sprite_ram+19h
0000:18DB 46      ld      b, (hl)
0000:18DC CB A8     res     5, b
0000:18DE AF      xor     a
0000:18DF CD 09 30     call    sub_0_3009
0000:18E2 F6 20     or      20h ; ' '
0000:18E4 77      ld      (hl), a
0000:18E5 21 AF 62     ld      hl, unk_0_62AF
0000:18E8 7E      ld      a, (hl)
0000:18E9 FE E0     cp      0E0h ; 'Ó'
0000:18EB C2 10 19     jp      nz, loc_0_1910
0000:18EE 3E 50     ld      a, 50h ; 'P'
0000:18F0 32 4F 69     ld      (soft_sprite_ram+4Fh), a
0000:18F3 3E 00     ld      a, 0
0000:18F5 32 4D 69     ld      (soft_sprite_ram+4Dh), a
0000:18F8 3E 9F     ld      a, 9Fh ; 'f'
0000:18FA 32 4C 69     ld      (soft_sprite_ram+4Ch), a
0000:18FD 3A 03 62     ld      a, (mario_y_coord)
0000:1900 FE 80     cp      80h ; 'Ç'
0000:1902 D2 0F 19     jp      nc, loc_0_190F
0000:1905 3E 80     ld      a, 80h ; 'Ç'
0000:1907 32 4D 69     ld      (soft_sprite_ram+4Dh), a
0000:190A 3E 5F     ld      a, 5Fh ; '-'
0000:190C 32 4C 69     ld      (soft_sprite_ram+4Ch), a
0000:190F
0000:190F      loc_0_190F:      ld      a, (hl) ; CODE XREF: 0000:1902|j
0000:1910
0000:1910      loc_0_1910:      cp      0C0h ; 'L' ; CODE XREF: 0000:18EB|j
0000:1912 C0      ret     nz
0000:1913 21 8A 60     ld      hl, unk_0_608A
0000:1916 36 0C     ld      (hl), 0Ch
0000:1918 3A 29 62     ld      a, (level)
0000:191B 0F      rrca
0000:191C 38 02     jr      c, loc_0_1920
0000:191E 36 05     ld      (hl), 5
0000:1920
0000:1920      loc_0_1920:      inc     hl ; CODE XREF: 0000:191C|j
0000:1922 23      ld      (hl), 3
0000:1923 21 23 6A     ld      hl, soft_sprite_ram+123h
0000:1926 36 40     ld      (hl), 40h ; '@'
0000:1928 2B      dec     hl
0000:1929 36 09     ld      (hl), 9
0000:192B 2B      dec     hl
0000:192C 36 76     ld      (hl), 76h ; 'v'
0000:192E 2B      dec     hl
0000:192F 36 8F     ld      (hl), 8Fh ; 'À'
0000:1931 3A 03 62     ld      a, (mario_y_coord)
0000:1934 FE 80     cp      80h ; 'Ç'
0000:1936 D0      ret     nc
0000:1937 3E 6F     ld      a, 6Fh ; 'o'
0000:1939 32 20 6A     ld      (soft_sprite_ram+120h), a
0000:193C C9      ret
;
0000:193D
0000:193D      loc_0_193D:      ld      hl, (seq_data) ; CODE XREF: 0000:18CA|j
0000:193E 2A 2A 62     inc     hl
0000:1941 7E      ld      a, (hl)
0000:1942 FE 7F     cp      7Fh ; '-' ; restart repeating levels?
0000:1944 C2 4B 19     jp      nz, loc_0_194B ; no, skip
0000:1947 21 73 3A     ld      hl, level_seq_2 ; start repeating levels
0000:194A 7E      ld      a, (hl) ; get new level
0000:194B
0000:194B      loc_0_194B:      ld      (seq_data), hl ; CODE XREF: 0000:1944|j
0000:194E 32 27 62     ld      (level_type), a
0000:1951 21 29 62     ld      hl, level
0000:1954 34      inc     (hl) ; next level counter
0000:1955 11 00 05     ld      de, 500h
0000:1958 CD 9F 30     call    sub_0_309F
0000:195B AF      xor     a
0000:195C 32 2E 62     ld      (unk_0_622E), a
0000:195F 32 88 63     ld      (unk_0_6388), a
0000:1962 21 09 60     ld      hl, eight_bit_countdown
0000:1965 3E E0     ld      (hl), 0E0h ; 'Ó'
0000:1967 23      inc     hl
0000:1968 36 08     ld      (hl), 8 ; set how high screen
0000:196A C9      ret
;
0000:196B
0000:196B      loc_0_196B:      call    clear_tiles_and_sprites ; DATA XREF: 0000:0730|o
0000:196B      ld      a, (current_player_E) ; 0/1
0000:1971 C6 12     add     a, 18
0000:1973 32 0A 60     ld      (main_sequencer), a ; 18/19
0000:1976 C9      ret
;
0000:1977
0000:1977      attract_mode_gameplay:      call    next_attract_action ; DATA XREF: 0000:074E|o
0000:197A
0000:197A      gameplay:      call    sub_0_1DBD ; DATA XREF: 0000:071A|o
0000:197D CD 8C 1E     call    sub_0_1E8C ; another jump table
0000:1980 CD C3 1A     call    sub_0_1AC3
0000:1983 CD 72 1F     call    sub_0_1F72
0000:1986 CD 8F 2C     call    sub_0_2C8F
0000:1989 CD 03 2C     call    sub_0_2C03
0000:198C CD ED 30     call    sub_0_30ED ; process fireballs?
0000:198F CD 04 2E     call    sub_0_2E04 ; process springs
0000:1992 CD EA 24     call    sub_0_24EA
0000:1995 CD DB 2D     call    sub_0_2DDB
0000:1998 CD D4 2E     call    sub_0_2ED4
0000:199B CD 07 22     call    sub_0_2207

```

```

0000:199E CD 33 1A      call    sub_0_1A33
0000:19A1 CD 85 2A      call    sub_0_2A85
0000:19A4 CD 46 1F      call    sub_0_1F46
0000:19A7 CD FA 26      call    sub_0_26FA
0000:19AA CD F2 25      call    sub_0_25F2
0000:19AD CD DA 19      call    sub_0_19DA
0000:19B0 CD FB 03      call    animate_kong_and_pauline
0000:19B3 CD 08 28      call    sub_0_2808
0000:19B6 CD 1D 28      call    sub_0_281D
0000:19B9 CD 57 1E      call    sub_0_1E57
0000:19BC CD 07 1A      call    sub_0_1A07
0000:19BF CD CB 2F      call    sub_0_2FCB
0000:19C2 00          nop
0000:19C3 00          nop
0000:19C4 00          nop
0000:19C5 3A 00 62      ld      a, (mario_alive_flag)
0000:19C8 A7          and     a, 0x07          ; mario alive?
0000:19C9 C0          ret     nz              ; yes, return
0000:19CA CD 1C 01      call    stop_sound
0000:19CD 21 82 60      ld      hl, digital_snd_tmr_thump
0000:19D0 36 03          ld      (hl), 3          ; tmr=3
0000:19D2          ; CODE XREF: 0000:1A30↑j
loc_0_19D2:          ld      hl, main_sequencer
0000:19D2 21 0A 60      inc     (hl)            ; next sequence
0000:19D5 34          dec     hl              ; 8-bit countdown
0000:19D6 2B          ld      (hl), 64        ; set counter
0000:19D7 36 40          ret
0000:19D9 C9          ; SUBROUTINE
0000:19DA          ; SUBROUTINE
0000:19DA          ; SUBROUTINE
0000:19DA          ; SUBROUTINE
0000:19DA          ; SUBROUTINE
sub_0_19DA:          ; CODE XREF: 0000:19AD↑p
0000:19DA 3A 03 62      ld      a, (mario_y_coord)
0000:19DD 06 03          ld      b, 3
0000:19DF 21 0C 6A      ld      hl, soft_sprite_ram+10Ch
0000:19E2          ; CODE XREF: sub_0_19DA+10↑j
loc_0_19E2:          cp      (hl)
0000:19E2 BE          jp      z, loc_0_19ED
0000:19E3 CA ED 19      inc     l
0000:19E6 2C          inc     l
0000:19E7 2C          inc     l
0000:19E8 2C          inc     l
0000:19E9 2C          inc     l
0000:19EA 10 F6      djnz   loc_0_19E2
0000:19EC C9          ret
0000:19ED          ;
0000:19ED          ;
0000:19ED          ;
loc_0_19ED:          ; CODE XREF: sub_0_19DA+9↑j
0000:19ED 3A 05 62      ld      a, (mario_x_coord)
0000:19F0 2C          inc     l
0000:19F1 2C          inc     l
0000:19F2 2C          inc     l
0000:19F3 BE          cp      (hl)
0000:19F4 C0          ret     nz
0000:19F5 2D          dec     l
0000:19F6 2D          dec     l
0000:19F7 CB 5E      bit     3, (hl)
0000:19F9 C0          ret     nz
0000:19FA 2D          dec     l
0000:19FB 22 43 63      ld      (unk_0_6343), hl
0000:19FE AF          xor     a
0000:19FF 32 42 63      ld      (unk_0_6342), a
0000:1A02 3C          inc     a
0000:1A03 32 40 63      ld      (unk_0_6340), a
0000:1A06 C9          ret
0000:1A06          ; End of function sub_0_19DA
0000:1A06          ; SUBROUTINE
0000:1A07          ; SUBROUTINE
0000:1A07          ; SUBROUTINE
0000:1A07          ; SUBROUTINE
sub_0_1A07:          ; CODE XREF: 0000:19BC↑p
0000:1A07 3A 86 63      ld      a, (unk_0_6386)
0000:1A0A EF          rst     28h            ; go!
0000:1A0A          ;
0000:1A0B 1E 1A      .dw locret_0_1A1E      ; Jump table
0000:1A0D 15 1A      .dw loc_0_1A15
0000:1A0F 1F 1A      .dw loc_0_1A1F
0000:1A11 2A 1A      .dw loc_0_1A2A
0000:1A13 00 00      .dw 0
0000:1A15          ;
0000:1A15          ;
0000:1A15          ;
loc_0_1A15:          ; DATA XREF: sub_0_1A07+6↑o
0000:1A15 AF          xor     a
0000:1A16 32 87 63      ld      (unk_0_6387), a
0000:1A19 3E 02      ld      a, 2
0000:1A1B 32 86 63      ld      (unk_0_6386), a
0000:1A1E          ; DATA XREF: sub_0_1A07+4↑o
locret_0_1A1E:      ret
0000:1A1E          ; End of function sub_0_1A07
0000:1A1E          ;
0000:1A1F          ;
0000:1A1F          ;
0000:1A1F          ;
loc_0_1A1F:          ; DATA XREF: sub_0_1A07+8↑o
0000:1A1F 21 87 63      ld      hl, 6387h
0000:1A22 35          dec     (hl)
0000:1A23 C0          ret     nz
0000:1A24 3E 03      ld      a, 3
0000:1A26 32 86 63      ld      (unk_0_6386), a
0000:1A29 C9          ret
0000:1A2A          ;
0000:1A2A          ;
0000:1A2A          ;
loc_0_1A2A:          ; DATA XREF: sub_0_1A07+A↑o
0000:1A2A 3A 16 62      ld      a, (mario_jumping)
0000:1A2D A7          and     a
0000:1A2E C0          ret     nz
0000:1A2F E1          pop     hl
0000:1A30 C3 D2 19      jp      loc_0_19D2
0000:1A33          ; SUBROUTINE
0000:1A33          ; SUBROUTINE
0000:1A33          ; SUBROUTINE
0000:1A33          ; SUBROUTINE
sub_0_1A33:          ; CODE XREF: 0000:199E↑p
0000:1A33 3E 08          ld      a, 8

```

```

0000:1A35 F7          rst      30h          ; return if level bit not set
0000:1A36 3A 03 62    ld      a, (mario_y_coord)
0000:1A39 FE 4B        cp      4Bh ; 'K'
0000:1A3B CA 4B 1A    jp      z, loc_0_1A4B
0000:1A3E FE B3        cp      0B3h ; '|'
0000:1A40 CA 4B 1A    jp      z, loc_0_1A4B
0000:1A43 3A 91 62    ld      a, (unk_0_6291)
0000:1A46 3D          dec      a
0000:1A47 CA 51 1A    jp      z, loc_0_1A51
0000:1A4A C9          ret
;
0000:1A4B
0000:1A4B
0000:1A4B          loc_0_1A4B:          ; CODE XREF: sub_0_1A33+8↑j
0000:1A4B 3E 01          ld      a, 1          ; sub_0_1A33+D↑j
0000:1A4D 32 91 62    ld      (unk_0_6291), a
0000:1A50 C9          ret
;
0000:1A51
0000:1A51          loc_0_1A51:          ; CODE XREF: sub_0_1A33+14↑j
0000:1A51 32 91 62    ld      (unk_0_6291), a
0000:1A54 47          ld      b, a
0000:1A55 3A 05 62    ld      a, (mario_x_coord)
0000:1A58 3D          dec      a
0000:1A59 FE D0        cp      0D0h ; 'ð'
0000:1A5B D0          ret      nc
0000:1A5C 07          rlca
0000:1A5D D2 62 1A    jp      nc, loc_0_1A62
0000:1A60 CB D0        set     2, b
0000:1A62
0000:1A62          loc_0_1A62:          ; CODE XREF: sub_0_1A33+2A↑j
0000:1A62 07          rlca
0000:1A63 07          rlca
0000:1A64 D2 69 1A    jp      nc, loc_0_1A69
0000:1A67 CB C8        set     1, b
0000:1A69
0000:1A69          loc_0_1A69:          ; CODE XREF: sub_0_1A33+31↑j
0000:1A69 E6 07        and     7
0000:1A6B FE 06        cp      6
0000:1A6D C2 72 1A    jp      nz, loc_0_1A72
0000:1A70 CB C8        set     1, b
0000:1A72
0000:1A72          loc_0_1A72:          ; CODE XREF: sub_0_1A33+3A↑j
0000:1A72 3A 03 62    ld      a, (mario_y_coord)
0000:1A75 07          rlca
0000:1A76 D2 7B 1A    jp      nc, loc_0_1A7B
0000:1A79 CB C0        set     0, b
0000:1A7B
0000:1A7B          loc_0_1A7B:          ; CODE XREF: sub_0_1A33+43↑j
0000:1A7B 21 92 62    ld      hl, unk_0_6292
0000:1A7E 78          ld      a, b
0000:1A7F 85          add     a, 1
0000:1A80 6F          ld      l, a
0000:1A81 7E          ld      a, (hl)
0000:1A82 A7          and     a
0000:1A83 C8          ret      z
0000:1A84 36 00        ld      (hl), 0
0000:1A86 21 90 62    ld      hl, unk_0_6290
0000:1A89 35          dec     (hl)
0000:1A8A 78          ld      a, b
0000:1A8B 01 05 00    ld      bc, 5
0000:1A8E 1F          rra
0000:1A8F DA BD 1A    jp      c, loc_0_1ABD
0000:1A92 21 CB 02    ld      hl, 2CBh
0000:1A95
0000:1A95          loc_0_1A95:          ; CODE XREF: sub_0_1A33+8D↑j
0000:1A95 A7          and     a
0000:1A96 CA 9E 1A    jp      z, loc_0_1A9E
0000:1A99
0000:1A99          loc_0_1A99:          ; CODE XREF: sub_0_1A33+68↑j
0000:1A99 09          add     hl, bc
0000:1A9A 3D          dec     a
0000:1A9B C2 99 1A    jp      nz, loc_0_1A99
0000:1A9E
0000:1A9E          loc_0_1A9E:          ; CODE XREF: sub_0_1A33+63↑j
0000:1A9E 01 00 74    ld      bc, VRAM_start
0000:1AA1 09          add     hl, bc
0000:1AA2 3E 10    ld      a, 10h
0000:1AA4 77          ld      (hl), a
0000:1AA5 2D          dec     l
0000:1AA6 77          ld      (hl), a
0000:1AA7 2C          inc     l
0000:1AA8 2C          inc     l
0000:1AA9 77          ld      (hl), a
0000:1AAA 3E 01    ld      a, 1
0000:1AAC 32 40 63    ld      (unk_0_6340), a
0000:1AAF 32 42 63    ld      (unk_0_6342), a
0000:1AB2 32 25 62    ld      (unk_0_6225), a
0000:1AB5 3A 16 62    ld      a, (mario_jumping)
0000:1AB8 A7          and     a
0000:1AB9 CC 95 1D    call    z, sub_0_1D95
0000:1ABC C9          ret
;
0000:1ABD
0000:1ABD
0000:1ABD          loc_0_1ABD:          ; CODE XREF: sub_0_1A33+5C↑j
0000:1ABD 21 2B 01    ld      hl, 12Bh
0000:1AC0 C3 95 1A    jp      loc_0_1A95
; End of function sub_0_1A33
;
; SUBROUTINE
;
0000:1AC3
0000:1AC3
0000:1AC3          sub_0_1AC3:          ; CODE XREF: 0000:1980↑p
0000:1AC3 3A 16 62    ld      a, (mario_jumping)
0000:1AC6 3D          dec     a
0000:1AC7 CA B2 1B    jp      z, loc_0_1BB2
0000:1ACA 3A 1E 62    ld      a, (unk_0_621E)
0000:1ACD A7          and     a
0000:1ACE C2 55 1B    jp      nz, loc_0_1B55
0000:1AD1 3A 17 62    ld      a, (unk_0_6217)
0000:1AD4 3D          dec     a
0000:1AD5 CA E6 1A    jp      z, loc_0_1AE6
0000:1AD8 3A 15 62    ld      a, (mario_climbing)
0000:1ADB 3D          dec     a
0000:1ADC CA 38 1B    jp      z, loc_0_1B38

```



```

0000:1ADF 3A 10 60      ld      a, (controller_in)
0000:1AE2 17              rla
0000:1AE3 DA 6E 1B      jp      c, loc_0_1B6E      ; jump pressed?
0000:1AE6              ; yes, skip
0000:1AE6      loc_0_1AE6:      ; CODE XREF: sub_0_1AC3+121j
0000:1AE6 CD 1F 24      call   sub_0_241F
0000:1AE9 3A 10 60      ld      a, (controller_in)
0000:1AEC 1D              dec     e
0000:1AED CA F5 1A      jp      z, loc_0_1AF5
0000:1AF0 CB 47              bit     0, a
0000:1AF2 C2 8F 1C      jp      nz, loc_0_1C8F
0000:1AF5      loc_0_1AF5:      ; CODE XREF: sub_0_1AC3+2A1j
0000:1AF5 15              dec     d
0000:1AF6 CA FE 1A      jp      z, loc_0_1AFE
0000:1AF9 CB 4F              bit     1, a
0000:1AFB C2 AB 1C      jp      nz, loc_0_1CAB
0000:1AFE      loc_0_1AFE:      ; CODE XREF: sub_0_1AC3+331j
0000:1AFE 3A 17 62      ld      a, (unk_0_6217)
0000:1B01 3D              dec     a
0000:1B02 C8              ret     z
0000:1B03 3A 05 62      ld      a, (mario_x_coord)
0000:1B06 C6 08              add     a, 8
0000:1B08 57              ld      d, a
0000:1B09 3A 03 62      ld      a, (mario_y_coord)
0000:1B0C F6 03              or      3
0000:1B0E CB 97              res     2, a
0000:1B10 01 15 00      ld      bc, 15h
0000:1B13 CD 6E 23      call   sub_0_236E
0000:1B16 F5              push    af
0000:1B17 21 07 62      ld      hl, mario_flipy_sprite_tile
0000:1B1A 7E              ld      a, (hl)
0000:1B1B E6 80              and     80h ; 'Ç'
0000:1B1D F6 06              or      6      ; mario climbing character
0000:1B1F 77              ld      (hl), a
0000:1B20 21 1A 62      ld      hl, unk_0_621A
0000:1B23 3E 04              ld      a, 4
0000:1B25 B9              cp      c
0000:1B26 36 01              ld      (hl), 1      ; set as broken ladder
0000:1B28 D2 2C 1B      jp      nc, loc_0_1B2C
0000:1B2B 35              dec     (hl)      ; set as normal ladder
0000:1B2C      loc_0_1B2C:      ; CODE XREF: sub_0_1AC3+651j
0000:1B2C F1              pop     af
0000:1B2D A7              and     a
0000:1B2E CA 4E 1B      jp      z, loc_0_1B4E
0000:1B31 7E              ld      a, (hl)
0000:1B32 A7              and     a
0000:1B33 C0              ret     nz
0000:1B34 2C              inc     l
0000:1B35 72              ld      (hl), d
0000:1B36 2C              inc     l      ; set top coordinate of ladder
0000:1B37 70              ld      (hl), b      ; set bottom Y coordinate of ladder
0000:1B38      loc_0_1B38:      ; CODE XREF: sub_0_1AC3+191j
0000:1B38 3A 10 60      ld      a, (controller_in)
0000:1B3B CB 5F              bit     3, a      ; down?
0000:1B3D C2 F2 1C      jp      nz, loc_0_1CF2      ; yes, go
0000:1B40 3A 15 62      ld      a, (mario_climbing)
0000:1B43 A7              and     a
0000:1B44 C8              ret     z
0000:1B45      loc_0_1B45:      ; CODE XREF: sub_0_1AC3+8F1j
0000:1B45 3A 10 60      ld      a, (controller_in)
0000:1B48 CB 57              bit     2, a      ; up?
0000:1B4A C2 03 1D      jp      nz, loc_0_1D03      ; yes, go
0000:1B4D C9              ret
0000:1B4E      ; _____
0000:1B4E      loc_0_1B4E:      ; CODE XREF: sub_0_1AC3+6B1j
0000:1B4E 2C              inc     l
0000:1B4F 70              ld      (hl), b
0000:1B50 2C              inc     l      ; set top Y corordinate of ladder
0000:1B51 72              ld      (hl), d      ; set bottom coordinate of ladder
0000:1B52 C3 45 1B      jp      loc_0_1B45
0000:1B55      ; _____
0000:1B55      loc_0_1B55:      ; CODE XREF: sub_0_1AC3+B1j
0000:1B55 21 1E 62      ld      hl, unk_0_621E
0000:1B58 35              dec     (hl)
0000:1B59 C0              ret     nz
0000:1B5A 3A 18 62      ld      a, (unk_0_6218)
0000:1B5D 32 17 62      ld      (unk_0_6217), a
0000:1B60 21 07 62      ld      hl, mario_flipy_sprite_tile
0000:1B63 7E              ld      a, (hl)
0000:1B64 E6 80              and     80h ; 'Ç'
0000:1B66 77              ld      (hl), a      ; h-flip mario
0000:1B67 AF              xor     a
0000:1B68 32 02 62      ld      (unk_0_6202), a
0000:1B6B C3 A6 1D      jp      loc_0_1DA6
0000:1B6E      ; _____
0000:1B6E      loc_0_1B6E:      ; CODE XREF: sub_0_1AC3+201j
0000:1B6E 3E 01              ld      a, 1      ; start_jump
0000:1B70 32 16 62      ld      (mario_jumping), a      ; set mario jumping
0000:1B73 21 10 62      ld      hl, unk_0_6210
0000:1B76 3A 10 60      ld      a, (controller_in)
0000:1B79 01 80 00      ld      bc, 80h ; 'Ç'
0000:1B7C 1F              rra      ; right?
0000:1B7D DA 8A 1B      jp      c, loc_0_1B8A      ; yes, skip
0000:1B80 01 80 FF      ld      bc, 0FF80h
0000:1B83 1F              rra      ; left?
0000:1B84 DA 8A 1B      jp      c, loc_0_1B8A      ; yes, skip
0000:1B87 01 00 00      ld      bc, 0
0000:1B8A      loc_0_1B8A:      ; CODE XREF: sub_0_1AC3+BA1j
0000:1B8A AF              ; sub_0_1AC3+C11j
0000:1B8A      xor     a
0000:1B8B 70              ld      (hl), b
0000:1B8C 2C              inc     l
0000:1B8D 71              ld      (hl), c
0000:1B8E 2C              inc     l
0000:1B8F 36 01              ld      (hl), 1
0000:1B91 2C              inc     l
0000:1B92 36 48              ld      (hl), 48h ; 'H'

```

```

0000:1B94 2C          inc     1
0000:1B95 77          ld      (hl), a
0000:1B96 32 04 62     ld      (unk_0_6204), a
0000:1B99 32 06 62     ld      (unk_0_6206), a
0000:1B9C 3A 07 62     ld      a, (mario_flipy_sprite_tile)
0000:1B9F E6 80        and     80h ; 'Ç'
0000:1BA1 F6 0E        or      0Eh ; mario jumping character
0000:1BA3 32 07 62     ld      (mario_flipy_sprite_tile), a
0000:1BA6 3A 05 62     ld      a, (mario_x_coord)
0000:1BA9 32 0E 62     ld      (unk_0_620E), a
0000:1BAC 21 81 60     ld      hl, digital_snd_tmr_jump
0000:1BAF 36 03      ld      (hl), 3 ; tmr=3
0000:1BB1 C9          ret
;
0000:1BB2
0000:1BB2
0000:1BB2      loc_0_1BB2:          ld      ix, mario_alive_flag ; CODE XREF: sub_0_1AC3+41j
0000:1BB2 DD 21 00 62     ld      a, (mario_y_coord)
0000:1BB6 3A 03 62     ld      (ix+0Bh), a ; store X position before a jump
0000:1BB9 DD 77 0B     ld      a, (mario_x_coord)
0000:1BBC 3A 05 62     ld      (ix+0Ch), a ; store Y position before a jump
0000:1BBF DD 77 0C     call    sub_0_239C
0000:1BC2 CD 9C 23     call    sub_0_241F
0000:1BC5 CD 1F 24     dec     d
0000:1BC8 15      jp      nz, loc_0_1BF2
0000:1BC9 C2 F2 1B     ld      (ix+10h), 0
0000:1BCC DD 36 10 00     ld      (ix+11h), 80h ; 'Ç'
0000:1BD0 DD 36 11 80     set     7, (ix+7) ; h-flip sprite
0000:1BD4 DD CB 07 FE
0000:1BD8      loc_0_1BD8:          ; CODE XREF: sub_0_1AC3+13F1j
0000:1BD8 3A 20 62     ld      a, (unk_0_6220)
0000:1BDB 3D      dec     a
0000:1BDC CA EC 1B     jp      z, loc_0_1BEC
0000:1BDF CD 07 24     call    sub_0_2407
0000:1BE2 DD 74 12     ld      (ix+12h), h
0000:1BE5 DD 75 13     ld      (ix+13h), l
0000:1BE8 DD 36 14 00     ld      (ix+14h), 0
0000:1BEC
0000:1BEC      loc_0_1BEC:          ; CODE XREF: sub_0_1AC3+1191j
0000:1BEC CD 9C 23     call    sub_0_239C
0000:1BEF C3 05 1C     jp      loc_0_1C05
;
0000:1BF2
0000:1BF2      loc_0_1BF2:          ; CODE XREF: sub_0_1AC3+1061j
0000:1BF2 1D      dec     e
0000:1BF3 C2 05 1C     jp      nz, loc_0_1C05
0000:1BF6 DD 36 10 FF     ld      (ix+10h), 0FFh
0000:1BFA DD 36 11 80     ld      (ix+11h), 80h ; 'Ç'
0000:1BFE DD CB 07 BE     res     7, (ix+7) ; un-hflip sprite
0000:1C02 C3 D8 1B     jp      loc_0_1BD8
;
0000:1C05
0000:1C05      loc_0_1C05:          ; CODE XREF: sub_0_1AC3+12C1j
0000:1C05 CD 1C 2B     ; sub_0_1AC3+1301j
0000:1C05
0000:1C08 3D      call    sub_0_2B1C ; are we jumping?
0000:1C09 CA 3A 1C     dec     a
0000:1C0C 3A 1F 62     jp      z, loc_0_1C3A
0000:1C0F 3D      ld      a, (unk_0_621F)
0000:1C10 CA 76 1C     dec     a
0000:1C13 3A 14 62     jp      z, loc_0_1C76
0000:1C16 D6 14      ld      a, (unk_0_6214)
0000:1C18 C2 33 1C     sub     14h
0000:1C1B 3E 01     jp      nz, loc_0_1C33
0000:1C1D 32 1F 62     ld      a, 1 ; peak of the jump
0000:1C20 CD 53 28     ld      (unk_0_621F), a
0000:1C23 A7      call    sub_0_2853 ; check for bonus points?
0000:1C24 CA A6 1D     and     a ; any bonus points?
0000:1C27 32 42 63     jp      z, loc_0_1DA6 ; no, exit
0000:1C2A 3E 01     ld      a, 1
0000:1C2C 32 40 63     ld      (unk_0_6340), a ; register bonus
0000:1C2F 32 25 62     ld      (unk_0_6225), a
0000:1C32 00      nop
0000:1C33
0000:1C33      loc_0_1C33:          ; CODE XREF: sub_0_1AC3+1551j
0000:1C33 3C      inc     a
0000:1C34 CC 54 29     call    z, sub_0_2954
0000:1C37 C3 A6 1D     jp      loc_0_1DA6
;
0000:1C3A
0000:1C3A      loc_0_1C3A:          ; CODE XREF: sub_0_1AC3+1461j
0000:1C3A 05      dec     b
0000:1C3B CA 4F 1C     jp      z, loc_0_1C4F
0000:1C3E 3C      inc     a
0000:1C3F 32 1F 62     ld      (unk_0_621F), a
0000:1C42 AF      xor     a
0000:1C43 21 10 62     ld      hl, 6210h
0000:1C46 06 05      ld      b, 5
0000:1C48
0000:1C48      loc_0_1C48:          ; CODE XREF: sub_0_1AC3+1871j
0000:1C48 77      ld      (hl), a
0000:1C49 2C      inc     l
0000:1C4A 10 FC     djnz    loc_0_1C48
0000:1C4C C3 A6 1D     jp      loc_0_1DA6
;
0000:1C4F
0000:1C4F      loc_0_1C4F:          ; CODE XREF: sub_0_1AC3+1781j
0000:1C4F 32 16 62     ld      (mario_jumping), a
0000:1C52 3A 20 62     ld      a, (unk_0_6220)
0000:1C55 EE 01      xor     1
0000:1C57 32 00 62     ld      (mario_alive_flag), a ; set whether mario survives a jump
0000:1C5A 21 07 62     ld      hl, mario_flipy_sprite_tile
0000:1C5D 7E      ld      a, (hl)
0000:1C5E E6 80        and     80h ; 'Ç'
0000:1C60 F6 0F        or      0Fh ; mario landing character
0000:1C62 77      ld      (hl), a
0000:1C63 3E 04      ld      a, 4
0000:1C65 32 1E 62     ld      (unk_0_621E), a
0000:1C68 AF      xor     a
0000:1C69 32 1F 62     ld      (unk_0_621F), a
0000:1C6C 3A 25 62     ld      a, (unk_0_6225)
0000:1C6F 3D      dec     a
0000:1C70 CC 95 1D     call    z, sub_0_1D95
0000:1C73 C3 A6 1D     jp      loc_0_1DA6
;
0000:1C76
0000:1C76

```

```

0000:1C76      loc_0_1C76:      ld      a, (mario_x_coord)      ; CODE XREF: sub_0_1AC3+14D1j
0000:1C76 3A 05 62      ld      hl, unk_0_620E
0000:1C79 21 0E 62      sub     0Fh
0000:1C7C D6 0F          cp      (hl)
0000:1C7E BE          jp      c, loc_0_1DA6
0000:1C7F DA A6 1D      ld      a, 1
0000:1C82 3E 01      ld      (unk_0_6220), a
0000:1C84 32 20 62      ld      hl, 6084h
0000:1C87 21 84 60      ld      (hl), 3
0000:1C8A 36 03      ld      (hl), 3
0000:1C8C C3 A6 1D      jp      loc_0_1DA6
0000:1C8F      ;
0000:1C8F      loc_0_1C8F:      ld      b, 1      ; CODE XREF: sub_0_1AC3+2F1j
0000:1C8F 06 01      ld      a, (unk_0_620F)
0000:1C91 3A 0F 62      and     a
0000:1C94 A7          jp      nz, loc_0_1CD2
0000:1C95 C2 D2 1C      ld      a, (unk_0_6202)
0000:1C98 3A 02 62      ld      b, a
0000:1C9B 47          ld      b, a
0000:1C9C 3E 05      ld      a, 5
0000:1C9E CD 09 30      call    sub_0_3009
0000:1CA1 32 02 62      ld      (unk_0_6202), a
0000:1CA4 E6 03      and     3
0000:1CA6 F6 80      or      80h ; 'Ç'
0000:1CA8 C3 C2 1C      jp      loc_0_1CC2
0000:1CAB      ;
0000:1CAB      loc_0_1CAB:      ld      b, 0FFh      ; CODE XREF: sub_0_1AC3+381j
0000:1CAB 06 FF      ld      a, (unk_0_620F)
0000:1CAD 3A 0F 62      and     a
0000:1CB0 A7          jp      nz, loc_0_1CD2
0000:1CB1 C2 D2 1C      ld      a, (unk_0_6202)
0000:1CB4 3A 02 62      ld      b, a
0000:1CB7 47          ld      b, a
0000:1CB8 3E 01      ld      a, 1
0000:1CBA CD 09 30      call    sub_0_3009
0000:1CBD 32 02 62      ld      (unk_0_6202), a
0000:1CC0 E6 03      and     3      ; animate mario running
0000:1CC2      ;
0000:1CC2      loc_0_1CC2:      ld      hl, mario_flip_sprite_tile      ; CODE XREF: sub_0_1AC3+1E51j
0000:1CC2 21 07 62      ld      (hl), a      ; set amrio character
0000:1CC5 77          rra
0000:1CC6 1F          call    c, sub_0_1D8F
0000:1CC7 DC 8F 1D      ld      a, 2
0000:1CCA 3E 02      ld      (unk_0_620F), a
0000:1CCC 32 0F 62      ld      (unk_0_620F), a
0000:1CCF C3 A6 1D      jp      loc_0_1DA6
0000:1CD2      ;
0000:1CD2      loc_0_1CD2:      ld      hl, mario_y_coord      ; CODE XREF: sub_0_1AC3+1D21j
0000:1CD2 21 03 62      ld      a, (hl)      ; sub_0_1AC3+1EE1j
0000:1CD2 7E          add     a, b
0000:1CD6 80          ld      (hl), a
0000:1CD7 77          ld      a, (level_type)
0000:1CD8 3A 27 62      dec     a
0000:1CDB 3D          jp      nz, loc_0_1CEB
0000:1CDC C2 EB 1C      ld      h, (hl)
0000:1CDF 66          ld      a, (mario_x_coord)
0000:1CE0 3A 05 62      ld      l, a
0000:1CE3 6F          call    sub_0_2333
0000:1CE4 CD 33 23      ld      a, 1
0000:1CE7 7D          ld      (mario_x_coord), a
0000:1CE8 32 05 62      ld      (mario_x_coord), a
0000:1CEB      ;
0000:1CEB      loc_0_1CEB:      ld      hl, unk_0_620F      ; CODE XREF: sub_0_1AC3+2191j
0000:1CEB 21 0F 62      dec     (hl)
0000:1CEE 35          jp      loc_0_1DA6
0000:1CEF C3 A6 1D      ;
0000:1CF2      ;
0000:1CF2      loc_0_1CF2:      ld      a, (unk_0_620F)      ; CODE XREF: sub_0_1AC3+7A1j
0000:1CF2 3A 0F 62      and     a      ; check timer
0000:1CF5 A7          and     a      ; expired?
0000:1CF6 C2 8A 1D      jp      nz, loc_0_1D8A      ; no, skip
0000:1CF9 3E 03      ld      a, 3
0000:1CFB 32 0F 62      ld      (unk_0_620F), a      ; reset timer
0000:1CFE 3E 02      ld      a, 2
0000:1D00 C3 11 1D      jp      loc_0_1D11
0000:1D03      ;
0000:1D03      loc_0_1D03:      ld      a, (unk_0_620F)      ; CODE XREF: sub_0_1AC3+871j
0000:1D03 3A 0F 62      and     a      ; check timer
0000:1D06 A7          and     a      ; expired?
0000:1D07 C2 76 1D      jp      nz, loc_0_1D76      ; no, skip
0000:1D0A 3E 04      ld      a, 4
0000:1D0C 32 0F 62      ld      (unk_0_620F), a      ; reset timer
0000:1D0F 3E FE      ld      a, 0FEh ; '■'
0000:1D11      ;
0000:1D11      loc_0_1D11:      ld      hl, mario_x_coord      ; CODE XREF: sub_0_1AC3+23D1j
0000:1D11 21 05 62      add     a, (hl)
0000:1D14 86          ld      (hl), a
0000:1D15 77          ld      b, a
0000:1D16 47          ld      a, (unk_0_6222)
0000:1D17 3A 22 62      xor     l, 1
0000:1D1A EE 01      ld      (unk_0_6222), a
0000:1D1C 32 22 62      ld      nz, loc_0_1D51
0000:1D1F C2 51 1D      ld      a, b
0000:1D22 78          add     a, 8
0000:1D23 C6 08      ld      hl, unk_0_621C
0000:1D25 21 1C 62      cp      (hl)
0000:1D28 BE          jp      z, loc_0_1D67      ; bottom y coordinate of ladder
0000:1D29 CA 67 1D      dec     l      ; stop from climbing
0000:1D2C 2D          sub     (hl)
0000:1D2D 96          jp      z, loc_0_1D67      ; top y coordinate of ladder
0000:1D2E CA 67 1D      ld      b, 5      ; stop from climbing
0000:1D31 06 05      sub     8
0000:1D33 D6 08      jp      z, loc_0_1D3F
0000:1D35 CA 3F 1D      dec     b
0000:1D38 05          sub     4
0000:1D39 D6 04      jp      z, loc_0_1D3F
0000:1D3B CA 3F 1D      dec     b
0000:1D3E 05          ;
0000:1D3F      loc_0_1D3F:      ; CODE XREF: sub_0_1AC3+2721j

```

```

0000:1D3F 3E 80                                ; sub_0_1AC3+278|j
0000:1D3F                                ld      a, 80h ; 'G'
0000:1D41 21 07 62                            ld      hl, mario_flipy_sprite_tile
0000:1D44 A6                                and     (hl)
0000:1D45 EE 80                                xor     80h ; 'G'                                ; hflip mario
0000:1D47 B0                                or      b
0000:1D48 77                                ld      (hl), a
0000:1D49
0000:1D49                                loc_0_1D49:                                ; CODE XREF: sub_0_1AC3+2A1|j
0000:1D49 3E 01                                ld      a, 1                                ; flag mario climbing a ladder
0000:1D4B 32 15 62                            ld      (mario_climbing), a
0000:1D4E C3 A6 1D                            jp      loc_0_1DA6
;
0000:1D51                                loc_0_1D51:                                ; CODE XREF: sub_0_1AC3+25C|j
0000:1D51                                dec     1
0000:1D52 2D                                dec     1
0000:1D53 7E                                ld      a, (hl)
0000:1D54 F6 03                                or      3
0000:1D56 CB 97                                res     2, a
0000:1D58 77                                ld      (hl), a
0000:1D59 3A 24 62                            ld      a, (unk_0_6224)
0000:1D5C EE 01                                xor     1
0000:1D5E 32 24 62                            ld      (unk_0_6224), a
0000:1D61 CC 8F 1D                            call    z, sub_0_1D8F
0000:1D64 C3 49 1D                            jp      loc_0_1D49
;
0000:1D67                                loc_0_1D67:                                ; CODE XREF: sub_0_1AC3+266|j
0000:1D67                                ld      a, 6                                ; sub_0_1AC3+26B|j
0000:1D69 32 07 62                            ld      (mario_flipy_sprite_tile), a        ; mario climbing character
0000:1D6C AF                                xor     a
0000:1D6D 32 19 62                            ld      (unk_0_6219), a
0000:1D70 32 15 62                            ld      (mario_climbing), a                ; flaf not climbing a ladder
0000:1D73 C3 A6 1D                            jp      loc_0_1DA6
;
0000:1D76                                loc_0_1D76:                                ; CODE XREF: sub_0_1AC3+244|j
0000:1D76 3A 1A 62                            ld      a, (unk_0_621A)
0000:1D79 A7                                and     a
0000:1D7A CA 8A 1D                            jp      z, loc_0_1D8A
0000:1D7D 32 19 62                            ld      (unk_0_6219), a
0000:1D80 3A 1C 62                            ld      a, (unk_0_621C)
0000:1D83 D6 13                                sub     13h
0000:1D85 21 05 62                            ld      hl, mario_x_coord
0000:1D88 BE                                cp      (hl)
0000:1D89 D0                                ret     nc
0000:1D8A                                loc_0_1D8A:                                ; CODE XREF: sub_0_1AC3+233|j
0000:1D8A                                ld      hl, unk_0_620F                        ; sub_0_1AC3+2B7|j
0000:1D8D 35                                dec     (hl)
0000:1D8E C9                                ret
; End of function sub_0_1AC3
;
; SUBROUTINE
;
0000:1D8F                                sub_0_1D8F:                                ; CODE XREF: sub_0_1AC3+204|p
0000:1D8F 3E 03                                ld      a, 3                                ; sub_0_1AC3+29E|p
0000:1D91 32 80 60                            ld      (digital_snd_tmr_walk), a          ; tmr=3
0000:1D94 C9                                ret
; End of function sub_0_1D8F
;
; SUBROUTINE
;
0000:1D95                                sub_0_1D95:                                ; CODE XREF: sub_0_1A33+86|p
0000:1D95                                ld      (unk_0_6225), a                      ; sub_0_1AC3+1AD|p
0000:1D98 3A 27 62                            ld      a, (level_type)
0000:1D9B 3D                                dec     a
0000:1D9C C8                                ret     z
0000:1D9D 21 8A 60                            ld      hl, unk_0_608A
0000:1DA0 36 0D                            ld      (hl), 0Dh
0000:1DA2 2C                                inc     1
0000:1DA3 36 03                            ld      (hl), 3
0000:1DA5 C9                                ret
; End of function sub_0_1D95
;
;
0000:1DA6                                loc_0_1DA6:                                ; CODE XREF: sub_0_1AC3+A8|j
0000:1DA6                                ld      hl, soft_sprite_ram+4Ch              ; sub_0_1AC3+161|j ...
0000:1DA9 3A 03 62                            ld      a, (mario_y_coord)
0000:1DAC 77                                ld      (hl), a
0000:1DAD 3A 07 62                            ld      a, (mario_flipy_sprite_tile)
0000:1DB0 2C                                inc     1
0000:1DB1 77                                ld      (hl), a
0000:1DB2 3A 08 62                            ld      a, (mario_flipx_colour)
0000:1DB5 2C                                inc     1
0000:1DB6 77                                ld      (hl), a
0000:1DB7 3A 05 62                            ld      a, (mario_x_coord)
0000:1DBA 2C                                inc     1
0000:1DBB 77                                ld      (hl), a
0000:1DBC C9                                ret
;
; SUBROUTINE
;
0000:1DBD                                sub_0_1DBD:                                ; CODE XREF: 0000:127C|p
0000:1DBD                                ld      a, (unk_0_6340)                      ; 0000:1641|p ...
0000:1DBD EF                                rst     28h                                ; go!
;
;
; .dw ret
; .dw loc_0_1DC9
; .dw loc_0_1E4A
; .db 0 ;
; .db 0 ;
;

```

```

0000:1DC9          loc_0_1DC9:          ; DATA XREF: sub_0_1DBD+6[o
0000:1DC9          ld          a, 40h ; '@'
0000:1DCB 3E 40          ld          (unk_0_6341), a
0000:1DCB 32 41 63          ld          a, 2
0000:1DCE 3E 02          ld          (unk_0_6340), a
0000:1DD0 32 40 63          ld          a, (unk_0_6342)
0000:1DD3 3A 42 63          rra
0000:1DD6 1F          jp          c, loc_0_3E70
0000:1DD7 DA 70 3E          rra
0000:1DDA 1F          jp          c, loc_0_1E00
0000:1ddb DA 00 1E          rra
0000:1dde 1F          jp          c, loc_0_1DF5
0000:1ddf DA F5 1D          ld          hl, digital_snd_tmr_barrel_jump_priz
0000:1DE2 21 85 60          ld          (hl), 3          ; tmr=3
0000:1DE5 36 03          ld          a, (level)
0000:1DE7 3A 29 62          dec
0000:1DEA 3D          jp          z, loc_0_1E00
0000:1DEB CA 00 1E          dec
0000:1DEE 3D          jp          z, award_500_pts
0000:1DEF CA 08 1E          jp          award_800_pts
0000:1DF2 C3 10 1E          ;
0000:1DF5          loc_0_1DF5:          ; CODE XREF: sub_0_1DBD+22[j
0000:1DF5          ld          a, (random_no)
0000:1DF8 1F          rra          ; 50% chance for 500 pts
0000:1DF9 DA 08 1E          jp          c, award_500_pts      ; award 500 pts
0000:1DFC 1F          rra          ; 25% chance for 800 pts
0000:1DFD DA 10 1E          jp          c, award_800_pts      ; award 800 pts
0000:1E00          loc_0_1E00:          ; CODE XREF: sub_0_1DBD+1E[j
0000:1E00          ld          b, 7Dh ; '}'
0000:1E02 11 03 00          ld          de, 3          ; '300' sprite tile
0000:1E05 C3 15 1E          jp          award_points          ; award 3 (300) points
0000:1E08          ;
0000:1E08          award_500_pts:          ; CODE XREF: sub_0_1DBD+32[j
0000:1E08          ld          b, 7Eh ; '~'
0000:1E0A 11 05 00          ld          de, 5          ; '500' sprite tile
0000:1E0D C3 15 1E          jp          award_points          ; award 5 (500) points
0000:1E10          ;
0000:1E10          award_800_pts:          ; CODE XREF: sub_0_1DBD+35[j
0000:1E10          ld          b, 7Fh ; ' '
0000:1E12 11 08 00          ld          de, 8          ; '800' sprite tile
0000:1E15          ; award 8 (800) points
0000:1E15          award_points:          ; CODE XREF: sub_0_1DBD+48[j
0000:1E15          call       sub_0_309F          ; sub_0_1DBD+50[j
0000:1E18 2A 43 63          ld          hl, (unk_0_6343)      ; schedule award points
0000:1E1B 7E          ld          a, (hl)          ; ptr x position
0000:1E1C 36 00          ld          (hl), 0          ; prize x position
0000:1E1E 2C          inc         l          ; erase prize
0000:1E1F 2C          inc         l          ; go to y position
0000:1E20 2C          inc         l
0000:1E21 4E          ld          c, (hl)          ; get y position
0000:1E22 C3 36 1E          jp          loc_0_1E36          ; program award sprite
0000:1E25          ;
0000:1E25 11 01 00          ld          de, 1          ; award 1 (100) pts
0000:1E28          loc_0_1E28:          ; CODE XREF: 0000:3E76[j
0000:1E28          call       sub_0_309F          ; 0000:3E7E[j ...
0000:1E28          ld          a, (mario_x_coord)      ; schedule award points
0000:1E2B 3A 05 62          add         a, 14h
0000:1E2E C6 14          ld          c, a
0000:1E30 4F          ld          a, (mario_y_coord)
0000:1E31 3A 03 62          nop
0000:1E34 00          nop
0000:1E35 00
0000:1E36          loc_0_1E36:          ; CODE XREF: sub_0_1DBD+65[j
0000:1E36          ld          hl, soft_sprite_ram+130h ; add bonus points sprite to display
0000:1E39 77          ld          (hl), a
0000:1E3A 2C          inc         l
0000:1E3B 70          ld          (hl), b
0000:1E3C 2C          inc         l
0000:1E3D 36 07          ld          (hl), 7
0000:1E3F 2C          inc         l
0000:1E40 71          ld          (hl), c
0000:1E41 3E 05          ld          a, 5
0000:1E43 F7          rst         30h          ; return if level bit not set
0000:1E44 21 85 60          ld          hl, digital_snd_tmr_barrel_jump_priz
0000:1E47 36 03          ld          (hl), 3          ; tmr=3
0000:1E49          ret:          ; DATA XREF: sub_0_1DBD+4[o
0000:1E49          ret
0000:1E49          ; End of function sub_0_1DBD
0000:1E4A          ;
0000:1E4A          loc_0_1E4A:          ; DATA XREF: sub_0_1DBD+8[o
0000:1E4A          ld          hl, unk_0_6341
0000:1E4D 35          dec         (hl)
0000:1E4E C0          ret         nz
0000:1E4F AF          xor         a
0000:1E50 32 30 6A          ld          (soft_sprite_ram+130h), a
0000:1E53 32 40 63          ld          (unk_0_6340), a
0000:1E56 C9          ret
0000:1E57          ; SUBROUTINE
0000:1E57          sub_0_1E57:          ; CODE XREF: 0000:19B9[p
0000:1E57          ld          a, (level_type)
0000:1E5A CB 57          bit         2, a
0000:1E5C C2 80 1E          jp          nz, loc_0_1E80
0000:1E5F 1F          rra
0000:1E60 3A 05 62          ld          a, (mario_x_coord)
0000:1E63 DA 7A 1E          jp          c, loc_0_1E7A
0000:1E66 FE 51          cp          51h ; 'Q'
0000:1E68 D0          ret         nc
0000:1E69 3A 03 62          ld          a, (mario_y_coord)
0000:1E6C 17          rla

```

```

0000:1E6D
0000:1E6D      loc_0_1E6D:                                     ; CODE XREF: sub_0_1E57+26↑j
0000:1E6D 3E 00      ld      a, 0
0000:1E6F DA 74 1E      jp      c, loc_0_1E74
0000:1E72 3E 80      ld      a, 80h ; 'Ç'
0000:1E74
0000:1E74      loc_0_1E74:                                     ; CODE XREF: sub_0_1E57+18↑j
0000:1E74 32 4D 69      ld      (soft_sprite_ram+4Dh), a
0000:1E77 C3 85 1E      jp      loc_0_1E85
0000:1E7A
0000:1E7A      loc_0_1E7A:                                     ; CODE XREF: sub_0_1E57+C↑j
0000:1E7A FE 31      cp      31h ; '1'
0000:1E7C D0          ret      nc
0000:1E7D C3 6D 1E      jp      loc_0_1E6D
0000:1E80
0000:1E80      loc_0_1E80:                                     ; CODE XREF: sub_0_1E57+5↑j
0000:1E80 3A 90 62      ld      a, (unk_0_6290)
0000:1E83 A7          and      a
0000:1E84 C0          ret      nz
0000:1E85
0000:1E85      loc_0_1E85:                                     ; CODE XREF: sub_0_1E57+20↑j
0000:1E85 3E 16      ld      a, 16h
0000:1E87 32 0A 60      ld      (main_sequencer), a
0000:1E8A E1          pop      hl
0000:1E8B C9          ret
; End of function sub_0_1E57
0000:1E8B
0000:1E8C      ; ██████████ SUBROUTINE ██████████
0000:1E8C
0000:1E8C      sub_0_1E8C:                                     ; CODE XREF: 0000:197D↑p
0000:1E8C 3A 50 63      ld      a, (unk_0_6350)
0000:1E8F A7          and      a
0000:1E90 C8          ret      z
0000:1E91 CD 96 1E      call    sub_0_1E96
0000:1E94 E1          pop      hl
0000:1E95 C9          ret
; End of function sub_0_1E8C
0000:1E95
0000:1E96      ; ██████████ SUBROUTINE ██████████
0000:1E96
0000:1E96      sub_0_1E96:                                     ; CODE XREF: sub_0_1E8C+5↑p
0000:1E96 3A 45 63      ld      a, (unk_0_6345)
0000:1E99 EF          rst      28h                                     ; go!
0000:1E99
;
0000:1E9A A0 1E      .dw loc_0_1EA0
0000:1E9C 09 1F      .dw loc_0_1F09
0000:1E9E 23 1F      .dw loc_0_1F23                                     ; Jump table
0000:1EA0
0000:1EA0      loc_0_1EA0:                                     ; DATA XREF: sub_0_1E96+4↑o
0000:1EA0 3A 52 63      ld      a, (unk_0_6352)                         ; hammer just hit something
0000:1EA3 FE 65      cp      65h ; 'e'
0000:1EA5 21 B8 69      ld      hl, soft_sprite_ram+0B8h                ; process hammer hit effect (start)
0000:1EA8 CA B4 1E      jp      z, loc_0_1EB4
0000:1EAB 21 D0 69      ld      hl, soft_sprite_ram+0D0h                ; fireball area in sprite ram
0000:1EAE DA B4 1E      jp      c, loc_0_1EB4
0000:1EB1 21 80 69      ld      hl, soft_sprite_ram+80h
0000:1EB4
0000:1EB4      loc_0_1EB4:                                     ; CODE XREF: sub_0_1E96+12↑j
0000:1EB4 DD 2A 51 63      ; sub_0_1E96+18↑j
0000:1EB4      ld      ix, (unk_0_6351)
0000:1EB8 16 00      ld      d, 0
0000:1EBA 3A 53 63      ld      a, (unk_0_6353)
0000:1EBD 5F          ld      e, a
0000:1EBE 01 04 00      ld      bc, 4
0000:1EC1 3A 54 63      ld      a, (unk_0_6354)
0000:1EC4 A7          and      a
0000:1EC5 CA CF 1E      jp      z, loc_0_1ECF
0000:1EC8
0000:1EC8      loc_0_1EC8:                                     ; CODE XREF: sub_0_1E96+36↑j
0000:1EC8 09          add      hl, bc
0000:1EC9 DD 19      add      ix, de
0000:1ECB 3D          dec      a
0000:1ECC C2 C8 1E      jp      nz, loc_0_1EC8
0000:1ECF
0000:1ECF      loc_0_1ECF:                                     ; CODE XREF: sub_0_1E96+2F↑j
0000:1ECF DD 36 00 00      ld      (ix+0), 0
0000:1ED3 DD 7E 15      ld      a, (ix+15h)
0000:1ED6 A7          and      a
0000:1ED7 3E 02      ld      a, 2
0000:1ED9 CA DE 1E      jp      z, loc_0_1EDE
0000:1EDC 3E 04      ld      a, 4
0000:1EDE
0000:1EDE      loc_0_1EDE:                                     ; CODE XREF: sub_0_1E96+43↑j
0000:1EDE 32 42 63      ld      (unk_0_6342), a
0000:1EE1 01 2C 6A      ld      bc, soft_sprite_ram+12Ch
0000:1EE4 7E          ld      a, (hl)
0000:1EE5 36 00      ld      (hl), 0
0000:1EE7 02          ld      (bc), a                                     ; flash sprite x coord
0000:1EE8 0C          inc      c
0000:1EE9 2C          inc      l
0000:1EEA 3E 60      ld      a, 60h ; ''
0000:1EEC 02          ld      (bc), a                                     ; initial hit sprite character
0000:1EED 0C          inc      c                                     ; flash sprite character
0000:1EEE 2C          inc      l
0000:1EEF 3E 0C      ld      a, 0Ch
0000:1EF1 02          ld      (bc), a
0000:1EF2 0C          inc      c
0000:1EF3 2C          inc      l
0000:1EF4 7E          ld      a, (hl)
0000:1EF5 02          ld      (bc), a                                     ; flash sprite y coord
0000:1EF6 21 45 63      ld      hl, unk_0_6345
0000:1EF9 34          inc      (hl)
0000:1EFA 2C          inc      l
0000:1EFB 36 06      ld      (hl), 6
0000:1EFD 2C          inc      l
0000:1EFE 36 05      ld      (hl), 5
0000:1F00 21 8A 60      ld      hl, unk_0_608A
0000:1F03 36 06      ld      (hl), 6
0000:1F05 2C          inc      l

```

```

0000:1F06 36 03      ld      (hl), 3
0000:1F08 C9          ret
; End of function sub_0_1E96
0000:1F08
; -----
0000:1F09
0000:1F09      loc_0_1F09:      ; DATA XREF: sub_0_1E96+6[o
0000:1F09 21 46 63      ld      hl, unk_0_6346      ; process hammer hit effect (middle)
0000:1F0C 35      dec      (hl)
0000:1F0D C0          ret      nz
0000:1F0E 36 06      ld      (hl), 6
0000:1F10 2C      inc      l
0000:1F11 35      dec      (hl)
0000:1F12 CA 1D 1F      jp      z, loc_0_1F1D
0000:1F15 21 2D 6A      ld      hl, 6A2Dh
0000:1F18 7E      ld      a, (hl)
0000:1F19 EE 01      xor      l      ; animate hit flash
0000:1F1B 77      ld      (hl), a
0000:1F1C C9          ret
; -----
0000:1F1D
0000:1F1D      loc_0_1F1D:      ; CODE XREF: 0000:1F12[j
0000:1F1D 36 04      ld      (hl), 4
0000:1F1F 2D      dec      l
0000:1F20 2D      dec      l
0000:1F21 34      inc      (hl)
0000:1F22 C9          ret
; -----
0000:1F23
0000:1F23      loc_0_1F23:      ; DATA XREF: sub_0_1E96+8[o
0000:1F23 21 46 63      ld      hl, unk_0_6346      ; process hammer hit effect (end)
0000:1F26 35      dec      (hl)
0000:1F27 C0          ret      nz
0000:1F28 36 0C      ld      (hl), 0Ch
0000:1F2A 2C      inc      l
0000:1F2B 35      dec      (hl)
0000:1F2C CA 34 1F      jp      z, loc_0_1F34
0000:1F2F 21 2D 6A      ld      hl, soft_sprite_ram+12Dh
0000:1F32 34      inc      (hl)      ; animate hit flash
0000:1F33 C9          ret
; -----
0000:1F34
0000:1F34      loc_0_1F34:      ; CODE XREF: 0000:1F2C[j
0000:1F34 2D      dec      l
0000:1F35 2D      dec      l
0000:1F36 AF      xor      a
0000:1F37 77      ld      (hl), a
0000:1F38 32 50 63      ld      (unk_0_6350), a      ; stop effect process
0000:1F3B 3C      inc      a
0000:1F3C 32 40 63      ld      (unk_0_6340), a
0000:1F3F 21 2C 6A      ld      hl, soft_sprite_ram+12Ch
0000:1F42 22 43 63      ld      (unk_0_6343), hl
0000:1F45 C9          ret
; -----
0000:1F46
0000:1F46      ; SUBROUTINE
0000:1F46
0000:1F46      sub_0_1F46:      ; CODE XREF: 0000:19A4[p
0000:1F46 3A 21 62      ld      a, (unk_0_6221)
0000:1F49 A7      and      a
0000:1F4A C8      ret      z
0000:1F4B AF      xor      a
0000:1F4C 32 04 62      ld      (unk_0_6204), a
0000:1F4F 32 06 62      ld      (unk_0_6206), a
0000:1F52 32 21 62      ld      (unk_0_6221), a
0000:1F55 32 10 62      ld      (unk_0_6210), a
0000:1F58 32 11 62      ld      (unk_0_6211), a
0000:1F5B 32 12 62      ld      (unk_0_6212), a
0000:1F5E 32 13 62      ld      (unk_0_6213), a
0000:1F61 32 14 62      ld      (unk_0_6214), a
0000:1F64 3C      inc      a
0000:1F65 32 16 62      ld      (mario_jumping), a
0000:1F68 32 1F 62      ld      (unk_0_621F), a
0000:1F6B 3A 05 62      ld      a, (mario_x_coord)
0000:1F6E 32 0E 62      ld      (unk_0_620E), a
0000:1F71 C9          ret
; End of function sub_0_1F46
0000:1F71
0000:1F72
0000:1F72      ; SUBROUTINE
0000:1F72
0000:1F72      sub_0_1F72:      ; CODE XREF: 0000:1983[p
0000:1F72 3A 27 62      ld      a, (level_type)
0000:1F75 3D      dec      a
0000:1F76 C0          ret      nz
0000:1F77 DD 21 00 67      ld      ix, unk_0_6700
0000:1F7B 21 80 69      ld      hl, soft_sprite_ram+80h
0000:1F7E 11 20 00      ld      de, 20h ; ' '
0000:1F81 06 0A      ld      b, 0Ah
0000:1F83
0000:1F83      loc_0_1F83:      ; CODE XREF: sub_0_1F72+1E[j
0000:1F83 DD 7E 00      ld      a, (ix+0)
0000:1F86 3D      dec      a
0000:1F87 CA 93 1F      jp      z, loc_0_1F93
0000:1F8A 2C      inc      l
0000:1F8B 2C      inc      l
0000:1F8C 2C      inc      l
0000:1F8D
0000:1F8D      loc_0_1F8D:      ; CODE XREF: 0000:21CE[j
0000:1F8D 2C      inc      l
0000:1F8E DD 19      add      ix, de
0000:1F90 10 F1      djnz    loc_0_1F83
0000:1F92 C9          ret
; -----
0000:1F93
0000:1F93      loc_0_1F93:      ; CODE XREF: sub_0_1F72+15[j
0000:1F93 DD 7E 01      ld      a, (ix+1)
0000:1F96 3D      dec      a
0000:1F97 CA EC 20      jp      z, loc_0_20EC
0000:1F9A DD 7E 02      ld      a, (ix+2)
0000:1F9D 1F      rra
0000:1F9E DA AC 1F      jp      c, loc_0_1FAC
0000:1FA1 1F      rra
0000:1FA2 DA E5 1F      jp      c, loc_0_1FE5
0000:1FA5 1F      rra

```

```

0000:1FA6 DA EF 1F      jp      c, loc_0_1FEF
0000:1FA9 C3 53 20      jp      loc_0_2053
0000:1FAC      ; _____
0000:1FAC      loc_0_1FAC:                                     ; CODE XREF: sub_0_1F72+2C|j
0000:1FAC D9              exx
0000:1FAD DD 34 05      inc     (ix+5)
0000:1FB0 DD 7E 17      ld      a, (ix+17h)
0000:1FB3 DD BE 05      cp      (ix+5)
0000:1FB6 C2 CE 1F      jp      nz, loc_0_1FCE
0000:1FB9 DD 7E 15      ld      a, (ix+15h)
0000:1FBC 07          rlca
0000:1FBD 07          rlca
0000:1FBE C6 15      add     a, 15h                                     ; switch downwards (sideways) barrel to rolling barrel
0000:1FC0 DD 77 07      ld      (ix+7), a
0000:1FC3 DD 7E 02      ld      a, (ix+2)
0000:1FC6 EE 07      xor     7
0000:1FC8 DD 77 02      ld      (ix+2), a
0000:1FCB C3 BA 21      jp      loc_0_21BA
0000:1FCE      ; _____
0000:1FCE      loc_0_1FCE:                                     ; CODE XREF: sub_0_1F72+44|j
0000:1FCE DD 7E 0F      ; sub_0_1F72+199|j
0000:1FCE      ld      a, (ix+0Fh)
0000:1FD1 3D      dec     a
0000:1FD2 C2 DF 1F      jp      nz, loc_0_1FDF
0000:1FD5 DD 7E 07      ld      a, (ix+7)                                     ; animate sideways barrel sprite
0000:1FD8 EE 01      xor     1
0000:1FDA DD 77 07      ld      (ix+7), a
0000:1FDD 3E 04      ld      a, 4
0000:1FDF      loc_0_1FDF:                                     ; CODE XREF: sub_0_1F72+60|j
0000:1FDF DD 77 0F      ld      (ix+0Fh), a
0000:1FE2 C3 BA 21      jp      loc_0_21BA
0000:1FE5      ; _____
0000:1FE5      loc_0_1FE5:                                     ; CODE XREF: sub_0_1F72+30|j
0000:1FE5 D9              exx
0000:1FE6 01 00 01      ld      bc, 100h
0000:1FE9 DD 34 03      inc     (ix+3)
0000:1FEC C3 F6 1F      jp      loc_0_1FF6
0000:1FEF      ; _____
0000:1FEF      loc_0_1FEF:                                     ; CODE XREF: sub_0_1F72+34|j
0000:1FEF D9              exx
0000:1FF0 01 04 FF      ld      bc, 0FF04h
0000:1FF3 DD 35 03      dec     (ix+3)
0000:1FF6      loc_0_1FF6:                                     ; CODE XREF: sub_0_1F72+7A|j
0000:1FF6 DD 66 03      ld      h, (ix+3)
0000:1FF9 DD 6E 05      ld      l, (ix+5)
0000:1FFC 7C      ld      a, h
0000:1FFD E6 07      and     7
0000:1FFF FE 03      cp      3
0000:2001 CA 5F 21      jp      z, loc_0_215F
0000:2004 2D      dec     l
0000:2005 2D      dec     l
0000:2006 2D      dec     l
0000:2007 CD 33 23      call    sub_0_2333
0000:200A 2C      inc     l
0000:200B 2C      inc     l
0000:200C 2C      inc     l
0000:200D 7D      ld      a, l
0000:200E DD 77 05      ld      (ix+5), a
0000:2011 CD DE 23      call    sub_0_23DE
0000:2014 CD B4 24      call    sub_0_24B4
0000:2017 DD 7E 03      ld      a, (ix+3)
0000:201A FE 1C      cp      1Ch
0000:201C DA 2F 20      jp      c, loc_0_202F
0000:201F FE E4      cp      0E4h ; 'δ'
0000:2021 DA BA 21      jp      c, loc_0_21BA
0000:2024 AF      xor     a
0000:2025 DD 77 10      ld      (ix+10h), a
0000:2028 DD 36 11 60      ld      (ix+11h), 60h ; ''
0000:202C C3 38 20      jp      loc_0_2038
0000:202F      ; _____
0000:202F      loc_0_202F:                                     ; CODE XREF: sub_0_1F72+AA|j
0000:202F AF      xor     a
0000:2030 DD 36 10 FF      ld      (ix+10h), 0FFh
0000:2034 DD 36 11 A0      ld      (ix+11h), 0A0h ; 'á'
0000:2038      loc_0_2038:                                     ; CODE XREF: sub_0_1F72+BA|j
0000:2038 DD 36 12 FF      ld      (ix+12h), 0FFh
0000:203C DD 36 13 F0      ld      (ix+13h), 0F0h ; '-'
0000:2040 DD 77 14      ld      (ix+14h), a
0000:2043 DD 77 0E      ld      (ix+0Eh), a
0000:2046 DD 77 04      ld      (ix+4), a
0000:2049 DD 77 06      ld      (ix+6), a
0000:204C DD 36 02 08      ld      (ix+2), 8
0000:2050 C3 BA 21      jp      loc_0_21BA
0000:2053      ; _____
0000:2053      loc_0_2053:                                     ; CODE XREF: sub_0_1F72+37|j
0000:2053 D9              exx
0000:2054 CD 9C 23      call    sub_0_239C
0000:2057 CD 2F 2A      call    sub_0_2A2F
0000:205A A7      and     a
0000:205B C2 83 20      jp      nz, loc_0_2083
0000:205E DD 7E 03      ld      a, (ix+3)
0000:2061 C6 08      add     a, 8
0000:2063 FE 10      cp      10h
0000:2065 DA 79 20      jp      c, loc_0_2079
0000:2068 CD B4 24      call    sub_0_24B4
0000:206B DD 7E 10      ld      a, (ix+10h)
0000:206E E6 01      and     1
0000:2070 07          rlca
0000:2071 07          rlca
0000:2072 4F      ld      c, a
0000:2073 CD DE 23      call    sub_0_23DE
0000:2076 C3 BA 21      jp      loc_0_21BA
0000:2079      ; _____
0000:2079      loc_0_2079:                                     ; CODE XREF: sub_0_1F72+F3|j
0000:2079 AF      xor     a
0000:207A DD 77 00      ld      (ix+0), a

```



```

0000:207D DD 77 03      ld      (ix+3), a
0000:2080 C3 BA 21      jp      loc_0_21BA
0000:2083      ;
0000:2083      loc_0_2083:      ; CODE XREF: sub_0_1F72+E9|j
0000:2083 DD 34 0E      inc      (ix+0Eh)
0000:2086 DD 7E 0E      ld      a, (ix+0Eh)
0000:2089 3D          dec      a
0000:208A CA A2 20      jp      z, loc_0_20A2
0000:208D 3D          dec      a
0000:208E CA C3 20      jp      z, loc_0_20C3
0000:2091 DD 7E 10      ld      a, (ix+10h)
0000:2094 3D          dec      a
0000:2095 3E 04      ld      a, 4
0000:2097 C2 9C 20      jp      nz, loc_0_209C
0000:209A 3E 02      ld      a, 2
0000:209C      ;
0000:209C      loc_0_209C:      ; CODE XREF: sub_0_1F72+125|j
0000:209C DD 77 02      ld      (ix+2), a
0000:209F C3 BA 21      jp      loc_0_21BA
0000:20A2      ;
0000:20A2      loc_0_20A2:      ; CODE XREF: sub_0_1F72+118|j
0000:20A2 DD 7E 15      ld      a, (ix+15h)
0000:20A5 A7          and      a
0000:20A6 C2 B5 20      jp      nz, loc_0_20B5
0000:20A9 21 05 62      ld      hl, mario_x_coord
0000:20AC DD 7E 05      ld      a, (ix+5)
0000:20AF D6 16      sub      16h
0000:20B1 BE          cp      (hl)
0000:20B2 D2 C3 20      jp      nc, loc_0_20C3
0000:20B5      ;
0000:20B5      loc_0_20B5:      ; CODE XREF: sub_0_1F72+134|j
0000:20B5 DD 7E 10      ld      a, (ix+10h)
0000:20B8 A7          and      a
0000:20B9 C2 E1 20      jp      nz, loc_0_20E1
0000:20BC DD 77 11      ld      (ix+11h), a
0000:20BF DD 36 10 FF      ld      (ix+10h), 0FFh
0000:20C3      ;
0000:20C3      loc_0_20C3:      ; CODE XREF: sub_0_1F72+11C|j
0000:20C3 CD 07 24      ; sub_0_1F72+140|j ...
0000:20C3      call     sub_0_2407
0000:20C6 CB 3C      srl      h
0000:20C8 CB 1D      rr      l
0000:20CA CB 3C      srl      h
0000:20CC CB 1D      rr      l
0000:20CE DD 74 12      ld      (ix+12h), h
0000:20D1 DD 75 13      ld      (ix+13h), l
0000:20D4 AF          xor      a
0000:20D5 DD 77 14      ld      (ix+14h), a
0000:20D8 DD 77 04      ld      (ix+4), a
0000:20DB DD 77 06      ld      (ix+6), a
0000:20DE C3 BA 21      jp      loc_0_21BA
0000:20E1      ;
0000:20E1      loc_0_20E1:      ; CODE XREF: sub_0_1F72+147|j
0000:20E1 DD 36 10 01      ld      (ix+10h), 1
0000:20E5 DD 36 11 00      ld      (ix+11h), 0
0000:20E9 C3 C3 20      jp      loc_0_20C3
0000:20EC      ;
0000:20EC      loc_0_20EC:      ; CODE XREF: sub_0_1F72+25|j
0000:20EC D9          exx
0000:20ED CD 9C 23      call     sub_0_239C
0000:20F0 7C          ld      a, h
0000:20F1 D6 1A      sub      1Ah
0000:20F3 DD 46 19      ld      b, (ix+19h)
0000:20F6 B8          cp      b
0000:20F7 DA 04 21      jp      c, loc_0_2104
0000:20FA CD 2F 2A      call     sub_0_2A2F
0000:20FD A7          and      a
0000:20FE C2 18 21      jp      nz, loc_0_2118
0000:2101 CD B4 24      call     sub_0_24B4
0000:2104      ;
0000:2104      loc_0_2104:      ; CODE XREF: sub_0_1F72+185|j
0000:2104 DD 7E 03      ld      a, (ix+3)
0000:2107 C6 08      add      a, 8
0000:2109 FE 10      cp      10h
0000:210B D2 CE 1F      jp      nc, loc_0_1FCE
0000:210E AF          xor      a
0000:210F DD 77 00      ld      (ix+0), a
0000:2112 DD 77 03      ld      (ix+3), a
0000:2115 C3 BA 21      jp      loc_0_21BA
0000:2118      ;
0000:2118      loc_0_2118:      ; CODE XREF: sub_0_1F72+18C|j
0000:2118 DD 7E 05      ld      a, (ix+5)
0000:211B FE E0      cp      0E0h ; 'Ó'
0000:211D DA 46 21      jp      c, loc_0_2146
0000:2120 DD 7E 07      ld      a, (ix+7)
0000:2123 E6 FC      and      0FCh ; '3'
0000:2125 F6 01      or      l
0000:2127 DD 77 07      ld      (ix+7), a
0000:212A AF          xor      a
0000:212B DD 77 01      ld      (ix+1), a
0000:212E DD 77 02      ld      (ix+2), a
0000:2131 DD 36 10 FF      ld      (ix+10h), 0FFh
0000:2135 DD 77 11      ld      (ix+11h), a
0000:2138 DD 77 12      ld      (ix+12h), a
0000:213B DD 36 13 B0      ld      (ix+13h), 0B0h ; '0'
0000:213F DD 36 0E 01      ld      (ix+0Eh), 1
0000:2143 C3 53 21      jp      loc_0_2153
0000:2146      ;
0000:2146      loc_0_2146:      ; CODE XREF: sub_0_1F72+1AB|j
0000:2146 CD 07 24      call     sub_0_2407
0000:2149 CD CB 22      call     sub_0_22CB
0000:214C DD 7E 05      ld      a, (ix+5)
0000:214F DD 77 19      ld      (ix+19h), a
0000:2152 AF          xor      a
0000:2153      ;
0000:2153      loc_0_2153:      ; CODE XREF: sub_0_1F72+1D1|j
0000:2153 DD 77 14      ld      (ix+14h), a
0000:2156 DD 77 04      ld      (ix+4), a
0000:2159 DD 77 06      ld      (ix+6), a
0000:215C C3 BA 21      jp      loc_0_21BA

```

```

0000:215F      ; -----
0000:215F
0000:215F      loc_0_215F:                                     ; CODE XREF: sub_0_1F72+8F!j
0000:215F 7D          ld      a, 1
0000:2160 C6 05      add     a, 5
0000:2162 57          ld      d, a
0000:2163 7C          ld      a, h
0000:2164 01 15 00      ld      bc, 15h
0000:2167 CD 6D 21      call   sub_0_216D
0000:216A C3 BA 21      jp      loc_0_21BA
0000:216A      ; End of function sub_0_1F72
0000:216A
0000:216D      ; [REDACTED] SUBROUTINE [REDACTED]
0000:216D
0000:216D      sub_0_216D:                                     ; CODE XREF: sub_0_1F72+1F5!p
0000:216D CD 6E 23      call   sub_0_236E
0000:2170 3D          dec     a
0000:2171 C0          ret     nz
0000:2172 78          ld      a, b
0000:2173 D6 05      sub     5
0000:2175 DD 77 17      ld      (ix+17h), a
0000:2178 3A 48 63      ld      a, (unk_0_6348)
0000:217B A7          and     a
0000:217C CA B2 21      jp      z, loc_0_21B2
0000:217F 3A 05 62      ld      a, (mario_x_coord)
0000:2182 D6 04      sub     4
0000:2184 BA          cp      d
0000:2185 D8          ret     c
0000:2186 3A 80 63      ld      a, (unk_0_6380)
0000:2189 1F          rra
0000:218A 3C          inc     a
0000:218B 47          ld      b, a
0000:218C 3A 18 60      ld      a, (random_no)
0000:218F 4F          ld      c, a
0000:2190 E6 03      and     3
0000:2192 B8          cp      b
0000:2193 D0          ret     nc
0000:2194 21 10 60      ld      hl, controller_in
0000:2197 3A 03 62      ld      a, (mario_y_coord)
0000:219A BB          cp      e
0000:219B CA B2 21      jp      z, loc_0_21B2
0000:219E D2 A9 21      jp      nc, loc_0_21A9
0000:21A1 CB 46          bit     0, (hl)
0000:21A3 CA AE 21      jp      z, loc_0_21AE
0000:21A6 C3 B2 21      jp      loc_0_21B2
0000:21A9      ; -----
0000:21A9      loc_0_21A9:                                     ; CODE XREF: sub_0_216D+31!j
0000:21A9 CB 4E          bit     1, (hl)
0000:21AB C2 B2 21      jp      nz, loc_0_21B2
0000:21AE      ; CODE XREF: sub_0_216D+36!j
0000:21AE      loc_0_21AE:
0000:21AE 79          ld      a, c
0000:21AF E6 18      and     18h
0000:21B1 C0          ret     nz
0000:21B2      ; CODE XREF: sub_0_216D+F!j
0000:21B2      loc_0_21B2:                                     ; sub_0_216D+2E!j ...
0000:21B2      ; sprite tile #
0000:21B2 DD 34 07      inc     (ix+7)
0000:21B5 DD CB 02 C6      set     0, (ix+2)
0000:21B9 C9          ret
0000:21B9      ; End of function sub_0_216D
0000:21B9      ; -----
0000:21BA      ; -----
0000:21BA      loc_0_21BA:                                     ; CODE XREF: sub_0_1F72+59!j
0000:21BA D9          ; sub_0_1F72+70!j ...
0000:21BA      exx
0000:21BB DD 7E 03      ld      a, (ix+3)
0000:21BE 77          ld      (hl), a
0000:21BF 2C          inc     l
0000:21C0 DD 7E 07      ld      a, (ix+7)
0000:21C3 77          ld      (hl), a
0000:21C4 2C          inc     l
0000:21C5 DD 7E 08      ld      a, (ix+8)
0000:21C8 77          ld      (hl), a
0000:21C9 2C          inc     l
0000:21CA DD 7E 05      ld      a, (ix+5)
0000:21CD 77          ld      (hl), a
0000:21CE C3 8D 1F      jp      loc_0_1F8D
0000:21CE      ; -----
0000:21D1 80 FE      attract_mario_inputs: .dw 0FE80h
0000:21D1      ; DATA XREF: next_attract_action!o
0000:21D3 01 C0      ; 1st byte is input, 2nd is timer
0000:21D5 04 50      .dw 5004h
0000:21D7 02 10      .dw 1002h
0000:21D9 82 60      .dw 6082h
0000:21DB 02 10      .dw 1002h
0000:21DD 82 CA      .dw 0CA82h
0000:21DF 01 10      .dw 1001h
0000:21E1 81 FF      .dw 0FF81h
0000:21E3 02 38      .dw 3802h
0000:21E5 01 80      .dw 8001h
0000:21E7 02 FF      .dw 0FF02h
0000:21E9 04 80      .dw 8004h
0000:21EB 04 60      .dw 6004h
0000:21ED 80      .db 80h
0000:21EE      ; [REDACTED] SUBROUTINE [REDACTED]
0000:21EE
0000:21EE      next_attract_action:                             ; CODE XREF: 0000:1977!p
0000:21EE 11 D1 21      ld      de, attract_mario_inputs
0000:21F1 21 CC 63      ld      hl, attract_movement_entry
0000:21F4 7E          ld      a, (hl)
0000:21F5 07          rlca
0000:21F6 83          add     a, e
0000:21F7 5F          ld      e, a
0000:21F8 1A          ld      a, (de)
0000:21F9 32 10 60      ld      (controller_in), a
0000:21FC 2C          inc     l
0000:21FD 7E          ld      a, (hl)
0000:21FE 35          dec     (hl)
0000:21FF A7          and     a

```

```

0000:2200 C0          ret     nz                ; no, return
0000:2201 1C          inc     e                ; ptr 2nd byte of entry
0000:2202 1A          ld      a, (de)            ; get 2nd byte
0000:2203 77          ld      (hl), a            ; store as timer
0000:2204 2D          dec     l                ; back to entry
0000:2205 34          inc     (hl)             ; next entry
0000:2206 C9          ret
0000:2206          ; End of function next_attract_action
0000:2206
0000:2207          ; ██████████ SUBROUTINE ██████████
0000:2207
0000:2207          sub_0_2207:                ; CODE XREF: 0000:199B|p
0000:2207 3E 02          ld      a, 2
0000:2209 F7          rst     30h                ; return if level bit not set
0000:220A 3A 1A 60          ld      a, (gen_purpose_timer)
0000:220D 1F          rra
0000:220E 21 80 62          ld      hl, unk_0_6280
0000:2211 7E          ld      a, (hl)
0000:2212 DA 19 22          jp      c, loc_0_2219
0000:2215 21 88 62          ld      hl, unk_0_6288
0000:2218 7E          ld      a, (hl)
0000:2219          loc_0_2219:                ; CODE XREF: sub_0_2207+B|j
0000:2219 E5          push    hl
0000:221A EF          rst     28h                ; go!
0000:221B 27          daa
0000:221C 22 59 22          ld      (loc_0_2259), hl
0000:221F 99          sbc     a, c
0000:2220 22 A2 22          ld      (loc_0_22A2), hl
0000:2223 00          nop
0000:2224 00          nop
0000:2225 00          nop
0000:2226 00          nop
0000:2227 E1          pop     hl
0000:2228 2C          inc     l
0000:2229 35          dec     (hl)
0000:222A C2 3A 22          jp      nz, loc_0_223A
0000:222D 2D          dec     l
0000:222E 34          inc     (hl)
0000:222F 2C          inc     l
0000:2230 2C          inc     l
0000:2231 CD 43 22          call    sub_0_2243
0000:2234 3E 01          ld      a, 1
0000:2236 32 1A 62          ld      (unk_0_621A), a
0000:2239 C9          ret
0000:223A          ;
0000:223A          loc_0_223A:                ; CODE XREF: sub_0_2207+23|j
0000:223A 2C          inc     l
0000:223B CD 43 22          call    sub_0_2243
0000:223E AF          xor     a
0000:223F 32 1A 62          ld      (unk_0_621A), a
0000:2242 C9          ret
0000:2242          ; End of function sub_0_2207
0000:2242
0000:2243          ; ██████████ SUBROUTINE ██████████
0000:2243
0000:2243          sub_0_2243:                ; CODE XREF: sub_0_2207+2A|p
0000:2243          ; sub_0_2207+34|p ...
0000:2243 3A 05 62          ld      a, (mario_x_coord)
0000:2246 FE 7A          cp      7Ah ; 'z'
0000:2248 D2 57 22          jp      nc, loc_0_2257
0000:224B 3A 16 62          ld      a, (mario_jumping)
0000:224E A7          and     a
0000:224F C2 57 22          jp      nz, loc_0_2257
0000:2252 3A 03 62          ld      a, (mario_y_coord)
0000:2255 BE          cp      (hl)
0000:2256 C8          ret     z
0000:2257          loc_0_2257:                ; CODE XREF: sub_0_2243+5|j
0000:2257          ; sub_0_2243+C|j
0000:2257 E1          pop     hl
0000:2258 C9          ret
0000:2258          ; End of function sub_0_2243
0000:2258
0000:2259          ;
0000:2259          loc_0_2259:                ; DATA XREF: sub_0_2207+15|w
0000:2259 E1          pop     hl
0000:225A 2C          inc     l
0000:225B 2C          inc     l
0000:225C 2C          inc     l
0000:225D 2C          inc     l
0000:225E 35          dec     (hl)
0000:225F C0          ret     nz
0000:2260 3E 04          ld      a, 4
0000:2262 77          ld      (hl), a
0000:2263 2D          dec     l
0000:2264 34          inc     (hl)
0000:2265 CD BD 22          call    sub_0_22BD
0000:2268 3E 78          ld      a, 78h ; 'x'
0000:226A BE          cp      (hl)
0000:226B C2 75 22          jp      nz, loc_0_2275
0000:226E 2D          dec     l
0000:226F 2D          dec     l
0000:2270 2D          dec     l
0000:2271 34          inc     (hl)
0000:2272 2C          inc     l
0000:2273 2C          inc     l
0000:2274 2C          inc     l
0000:2275          loc_0_2275:                ; CODE XREF: 0000:226B|j
0000:2275 2D          dec     l
0000:2276 CD 43 22          call    sub_0_2243
0000:2279 3A 05 62          ld      a, (mario_x_coord)
0000:227C FE 68          cp      68h ; 'h'
0000:227E D2 8A 22          jp      nc, loc_0_228A
0000:2281          loc_0_2281:                ; CODE XREF: 0000:228B|j
0000:2281 21 05 62          ld      hl, mario_x_coord
0000:2284 34          inc     (hl)
0000:2285 CD C0 3F          call    sub_0_3FC0

```

```

0000:2288 34          inc      (hl)
0000:2289 C9          ret
; -----
0000:228A
0000:228A      loc_0_228A:                                ; CODE XREF: 0000:227E↑j
0000:228A 1F          rra
0000:228B DA 81 22      jp      c, loc_0_2281
0000:228E 1F          rra
0000:228F 3E 01          ld      a, 1
0000:2291 DA 95 22      jp      c, loc_0_2295
0000:2294 AF          xor      a
0000:2295
0000:2295      loc_0_2295:                                ; CODE XREF: 0000:2291↑j
0000:2295 32 22 62      ld      (unk_0_6222), a
0000:2298 C9          ret
; -----
0000:2299
0000:2299 E1          pop      hl
0000:229A 3A 18 60      ld      a, (random_no)
0000:229D F6 3C          and     3Ch ; '<'
0000:229F C0          ret      nz
0000:22A0 34          inc     (hl)
0000:22A1 C9          ret
; -----
0000:22A2*
0000:22A2*
0000:22A2*      loc_0_22A2:                                ; DATA XREF: sub_0_2207+19↑w
0000:22A2*E1      pop      hl
0000:22A3 2C          inc     l
0000:22A4 2C          inc     l
0000:22A5 2C          inc     l
0000:22A6 2C          inc     l
0000:22A7 35          dec     (hl)
0000:22A8 C0          ret      nz
0000:22A9 36 02          ld      (hl), 2
0000:22AB 2D          dec     l
0000:22AC 35          dec     (hl)
0000:22AD CD BD 22      call    sub_0_22BD
0000:22B0 3E 68          ld      a, 68h ; 'h'
0000:22B2 BE          cp      (hl)
0000:22B3 C0          ret      nz
0000:22B4 AF          xor      a
0000:22B5 06 80          ld      b, 80h ; 'G'
0000:22B7 2D          dec     l
0000:22B8 2D          dec     l
0000:22B9 70          ld      (hl), b
0000:22BA 2D          dec     l
0000:22BB 77          ld      (hl), a
0000:22BC C9          ret
; [REDACTED] SUBROUTINE [REDACTED]
0000:22BD
0000:22BD
0000:22BD
0000:22BD      sub_0_22BD:                                ; CODE XREF: 0000:2265↑p
0000:22BD 7E                                ; 0000:22AD↑p
0000:22BD          ld      a, (hl)
0000:22BE CB 5D          bit     3, l
0000:22C0 11 4B 69      ld      de, soft_sprite_ram+4Bh
0000:22C3 C2 C9 22      jp      nz, loc_0_22C9
0000:22C6 11 47 69      ld      de, soft_sprite_ram+47h
0000:22C9
0000:22C9      loc_0_22C9:                                ; CODE XREF: sub_0_22BD+6↑j
0000:22C9 12          ld      (de), a
0000:22CA C9          ret
; End of function sub_0_22BD
0000:22CA
0000:22CA      ; [REDACTED] SUBROUTINE [REDACTED]
0000:22CB
0000:22CB
0000:22CB      sub_0_22CB:                                ; CODE XREF: sub_0_1F72+1D↑p
0000:22CB          ld      a, (unk_0_6348)
0000:22CE A7          and     a
0000:22CF CA E1 22      jp      z, loc_0_22E1
0000:22D2 3A 80 63      ld      a, (unk_0_6380)
0000:22D5 3D          dec     a
0000:22D6 EF          rst      28h ; go!
; -----
0000:22D7 F6 22          .dw loc_0_22F6 ; Jump table
0000:22D9 F6 22          .dw loc_0_22F6
0000:22DB 03 23          .dw loc_0_2303
0000:22DD 03 23          .dw loc_0_2303
0000:22DF 1A 23          .dw loc_0_231A
; -----
0000:22E1
0000:22E1      loc_0_22E1:                                ; CODE XREF: sub_0_22CB+4↑j
0000:22E1          ld      a, (level)
0000:22E4 47          ld      b, a
0000:22E5 05          dec     b
0000:22E6 3E 01          ld      a, 1
0000:22E8 CA F9 22      jp      z, loc_0_22F9
0000:22EB 05          dec     b
0000:22EC 3E B1          ld      a, 0B1h ; '1'
0000:22EE CA F9 22      jp      z, loc_0_22F9
0000:22F1 3E E9          ld      a, 0E9h ; 'U'
0000:22F3 C3 F9 22      jp      loc_0_22F9
; -----
0000:22F6
0000:22F6      loc_0_22F6:                                ; DATA XREF: sub_0_22CB+C↑o
0000:22F6 3A 18 60          ld      a, (random_no) ; sub_0_22CB+E↑o
0000:22F9
0000:22F9      loc_0_22F9:                                ; CODE XREF: sub_0_22CB+1D↑j
0000:22F9          ; sub_0_22CB+23↑j ...
0000:22F9          ld      (ix+11h), a
0000:22FC E6 01          and     1
0000:22FE 3D          dec     a
0000:22FF DD 77 10          ld      (ix+10h), a
0000:2302 C9          ret
; End of function sub_0_22CB
; -----
0000:2302
0000:2302
0000:2303
0000:2303      loc_0_2303:                                ; DATA XREF: sub_0_22CB+10↑o
0000:2303          ; sub_0_22CB+12↑o
0000:2303          ld      a, (random_no)
0000:2306 DD 77 11          ld      (ix+11h), a
0000:2309 3A 03 62          ld      a, (mario_y_coord)

```

```

0000:230C DD BE 03      cp      (ix+3)
0000:230F 3E 01      ld      a, 1
0000:2311 D2 16 23      jp      nc, loc_0_2316
0000:2314 3D          dec     a
0000:2315 3D          dec     a
0000:2316
0000:2316      loc_0_2316:                                ; CODE XREF: 0000:2311|j
0000:2316 DD 77 10      ld      (ix+10h), a
0000:2319 C9          ret
;
0000:231A
0000:231A      loc_0_231A:                                ; DATA XREF: sub_0_22CB+14|o
0000:231A 3A 03 62      ld      a, (mario_y_coord)
0000:231D DD 96 03      sub     (ix+3)
0000:2320 0E FF      ld      c, 0FFh
0000:2322 DA 26 23      jp      c, loc_0_2326
0000:2325 0C          inc     c
0000:2326
0000:2326      loc_0_2326:                                ; CODE XREF: 0000:2322|j
0000:2326 07          rlca
0000:2327 CB 11      rl      c
0000:2329 07          rlca
0000:232A CB 11      rl      c
0000:232C DD 71 10      ld      (ix+10h), c
0000:232F DD 77 11      ld      (ix+11h), a
0000:2332 C9          ret
;
0000:2333
0000:2333      ; SUBROUTINE
0000:2333
0000:2333      sub_0_2333:                                ; CODE XREF: sub_0_1AC3+221|p
0000:2333 3E 0F      ld      a, 0Fh                                ; sub_0_1F72+95|p ...
0000:2335 A4          and     h
0000:2336 05          dec     b
0000:2337 CA 42 23      jp      z, loc_0_2342
0000:233A FE 0F      cp      0Fh
0000:233C D8          ret     c
0000:233D 06 FF      ld      b, 0FFh
0000:233F C3 47 23      jp      loc_0_2347
;
0000:2342
0000:2342      loc_0_2342:                                ; CODE XREF: sub_0_2333+4|j
0000:2342 FE 01      cp      1
0000:2344 D0          ret     nc
0000:2345 06 01      ld      b, 1
0000:2347
0000:2347      loc_0_2347:                                ; CODE XREF: sub_0_2333+C|j
0000:2347 3E F0      ld      a, 0F0h ; '-'
0000:2349 BD          cp      1
0000:234A CA 60 23      jp      z, loc_0_2360
0000:234D 3E 4C      ld      a, 4Ch ; 'L'
0000:234F BD          cp      1
0000:2350 CA 66 23      jp      z, loc_0_2366
0000:2353 7D          ld      a, 1
0000:2354 CB 6F      bit     5, a
0000:2356 CA 5C 23      jp      z, loc_0_235C
0000:2359
0000:2359      loc_0_2359:                                ; CODE XREF: sub_0_2333+2F|j
0000:2359 90          sub     b
0000:235A
0000:235A      loc_0_235A:                                ; CODE XREF: sub_0_2333+2A|j
0000:235A 6F          ld      1, a
0000:235B C9          ret
;
0000:235C
0000:235C      loc_0_235C:                                ; CODE XREF: sub_0_2333+23|j
0000:235C 80          add     a, b                                ; sub_0_2333+38|j
0000:235D C3 5A 23      jp      loc_0_235A
;
0000:2360
0000:2360      loc_0_2360:                                ; CODE XREF: sub_0_2333+17|j
0000:2360 CB 7C      bit     7, h
0000:2362 C2 59 23      jp      nz, loc_0_2359
0000:2365 C9          ret
;
0000:2366
0000:2366      loc_0_2366:                                ; CODE XREF: sub_0_2333+1D|j
0000:2366 7C          ld      a, h
0000:2367 FE 98      cp      98h ; 'x'
0000:2369 D8          ret     c
0000:236A 7D          ld      a, 1
0000:236B C3 5C 23      jp      loc_0_235C
; End of function sub_0_2333
0000:236B
0000:236B      ; SUBROUTINE
0000:236B
0000:236B      sub_0_236E:                                ; CODE XREF: sub_0_1AC3+50|p
0000:236E 21 00 63      ld      hl, unk_0_6300                                ; sub_0_216D|p ...
0000:2371
0000:2371      loc_0_2371:                                ; CODE XREF: sub_0_236E+1E|j
0000:2371 ED B1      cpir
0000:2373 C2 9A 23      jp      nz, loc_0_239A
0000:2376 E5          push    hl
0000:2377 C5          push    bc
0000:2378 01 14 00      ld      bc, 14h
0000:237B 09          add     hl, bc
0000:237C 0C          inc     c
0000:237D 5F          ld      e, a
0000:237E 7A          ld      a, d
0000:237F BE          cp      (hl)
0000:2380 CA 8F 23      jp      z, loc_0_238F
0000:2383 09          add     hl, bc
0000:2384 BE          cp      (hl)
0000:2385 CA 95 23      jp      z, loc_0_2395
0000:2388 57          ld      d, a
0000:2389 7B          ld      a, e
0000:238A C1          pop     bc
0000:238B E1          pop     hl
0000:238C C3 71 23      jp      loc_0_2371
;
0000:238F
0000:238F

```

```

0000:238F      loc_0_238F:      add     hl, bc      ; CODE XREF: sub_0_236E+12↑j
0000:238F 09      ld     a, 1
0000:2390 3E 01      jp     loc_0_2398
0000:2392 C3 98 23      ;
0000:2395
0000:2395      loc_0_2395:      xor     a          ; CODE XREF: sub_0_236E+17↑j
0000:2395 AF      sbc     hl, bc
0000:2396 ED 42
0000:2398      loc_0_2398:      pop     bc          ; CODE XREF: sub_0_236E+24↑j
0000:2398 C1      ld     b, (hl)
0000:2399 46
0000:239A      loc_0_239A:      pop     hl          ; CODE XREF: sub_0_236E+5↑j
0000:239A E1      ret
0000:239B C9
0000:239B      ; End of function sub_0_236E
0000:239C
0000:239C      ; [REDACTED] SUBROUTINE [REDACTED]
0000:239C
0000:239C      sub_0_239C:      ; CODE XREF: sub_0_1AC3+FF↑p
0000:239C DD 7E 04      ; sub_0_1AC3+129↑p ...
0000:239C      ld     a, (ix+4)
0000:239F DD 86 11      add     a, (ix+11h)
0000:23A2 DD 77 04      ld     (ix+4), a
0000:23A5 DD 7E 03      ld     a, (ix+3)
0000:23A8 DD 8E 10      adc     a, (ix+10h)
0000:23AB DD 77 03      ld     (ix+3), a
0000:23AE DD 7E 06      ld     a, (ix+6)
0000:23B1 DD 96 13      sub     (ix+13h)
0000:23B4 6F      ld     l, a
0000:23B5 DD 7E 05      ld     a, (ix+5)
0000:23B8 DD 9E 12      sbc     a, (ix+12h)
0000:23BB 67      ld     h, a
0000:23BC DD 7E 14      ld     a, (ix+14h)
0000:23BF A7      and     a
0000:23C0 17      rla
0000:23C1 3C      inc     a
0000:23C2 06 00      ld     b, 0
0000:23C4 CB 10      rl     b
0000:23C6 CB 27      sla     a
0000:23C8 CB 10      rl     b
0000:23CA CB 27      sla     a
0000:23CC CB 10      rl     b
0000:23CE CB 27      sla     a
0000:23D0 CB 10      rl     b
0000:23D2 4F      ld     c, a
0000:23D3 09      add     hl, bc
0000:23D4 DD 74 05      ld     (ix+5), h
0000:23D7 DD 75 06      ld     (ix+6), l
0000:23DA DD 34 14      inc     (ix+14h)
0000:23DD C9      ret
0000:23DD      ; End of function sub_0_239C
0000:23DD
0000:23DE      ; [REDACTED] SUBROUTINE [REDACTED]
0000:23DE
0000:23DE      sub_0_23DE:      ; CODE XREF: sub_0_1F72+9F↑p
0000:23DE DD 7E 0F      ; sub_0_1F72+101↑p
0000:23DE      ld     a, (ix+0Fh)
0000:23E1 3D      dec     a
0000:23E2 C2 03 24      jp     nz, loc_0_2403
0000:23E5 AF      xor     a          ; animate rolling barrels
0000:23E6 DD CB 07 26      sla     (ix+7)      ; toggle H & V flips
0000:23EA 17      rla
0000:23EB DD CB 08 26      sla     (ix+8)      ; toggle H & V flips
0000:23EF 17      rla
0000:23F0 47      ld     b, a
0000:23F1 3E 03      ld     a, 3
0000:23F3 B1      or     c
0000:23F4 CD 09 30      call    sub_0_3009
0000:23F7 1F      rra
0000:23F8 DD CB 08 1E      rr     (ix+8)      ; toggle H & V flips
0000:23FC 1F      rra
0000:23FD DD CB 07 1E      rr     (ix+7)      ; toggle H & V flips
0000:2401 3E 04      ld     a, 4
0000:2403
0000:2403      loc_0_2403:      ld     (ix+0Fh), a      ; CODE XREF: sub_0_23DE+4↑j
0000:2403 DD 77 0F      ret
0000:2406 C9
0000:2406      ; End of function sub_0_23DE
0000:2406
0000:2407      ; [REDACTED] SUBROUTINE [REDACTED]
0000:2407
0000:2407      sub_0_2407:      ; CODE XREF: sub_0_1AC3+11C↑p
0000:2407 DD 7E 14      ; sub_0_1F72+151↑p ...
0000:2407      ld     a, (ix+14h)
0000:240A 07      rlca
0000:240B 07      rlca
0000:240C 07      rlca
0000:240D 07      rlca
0000:240E 4F      ld     c, a
0000:240F E6 0F      and     0Fh
0000:2411 67      ld     h, a
0000:2412 79      ld     a, c
0000:2413 E6 F0      and     0F0h ; '-'
0000:2415 6F      ld     l, a
0000:2416 DD 4E 13      ld     c, (ix+13h)
0000:2419 DD 46 12      ld     b, (ix+12h)
0000:241C ED 42      sbc     hl, bc
0000:241E C9      ret
0000:241E      ; End of function sub_0_2407
0000:241E
0000:241F      ; [REDACTED] SUBROUTINE [REDACTED]
0000:241F
0000:241F      sub_0_241F:      ; CODE XREF: sub_0_1AC3+23↑p
0000:241F 11 00 01      ; sub_0_1AC3+102↑p ...
0000:241F      ld     de, 100h

```

```

0000:2422 3A 03 62      ld      a, (mario_y_coord)
0000:2425 FE 16      cp      16h
0000:2427 D8      ret      c
0000:2428 15      dec      d
0000:2429 1C      inc      e
0000:242A FE EA      cp      0EAh ; 'û'
0000:242C D0      ret      nc
0000:242D 1D      dec      e
0000:242E 3A 27 62      ld      a, (level_type)
0000:2431 0F      rrca
0000:2432 D0      ret      nc
0000:2433 3A 05 62      ld      a, (mario_x_coord)
0000:2436 FE 58      cp      58h ; 'X'
0000:2438 D0      ret      nc
0000:2439 3A 03 62      ld      a, (mario_y_coord)
0000:243C FE 6C      cp      6Ch ; 'l'
0000:243E D0      ret      nc
0000:243F 14      inc      d
0000:2440 C9      ret
0000:2440      ; End of function sub_0_241F
0000:2440
0000:2441
0000:2441      ; ██████████ S U B R O U T I N E ██████████
0000:2441
0000:2441      sub_0_2441:                                ; CODE XREF: 0000:0D62↑p
0000:2441 21 0C 3F      ld      hl, byte_0_3F09+3                ; anti-tamper check?
0000:2444 3E 5E      ld      a, 5Eh ; '^'
0000:2446 06 06      ld      b, 6
0000:2448
0000:2448      loc_0_2448:                                ; CODE XREF: sub_0_2441+9↑j
0000:2448 86      add      a, (hl)
0000:2449 23      inc      hl
0000:244A 10 FC      djnz    loc_0_2448
0000:244C FD 21 10 63      ld      iy, unk_0_6310
0000:2450 A7      and      a
0000:2451 CA 56 24      jp      z, loc_0_2456
0000:2454 FD 23      inc      iy
0000:2456
0000:2456      loc_0_2456:                                ; CODE XREF: sub_0_2441+10↑j
0000:2456 3A 27 62      ld      a, (level_type)
0000:2459 3D      dec      a
0000:245A 21 E4 3A      ld      hl, barrel_level_tilemap_data
0000:245D CA 71 24      jp      z, loc_0_2471
0000:2460 3D      dec      a
0000:2461 21 5D 3B      ld      hl, cement_pie_level_tilemap_data
0000:2464 CA 71 24      jp      z, loc_0_2471
0000:2467 3D      dec      a
0000:2468 21 E5 3B      ld      hl, elevator_level_tilemap_data
0000:246B CA 71 24      jp      z, loc_0_2471
0000:246E 21 8B 3C      ld      hl, rivet_level_tilemap_data
0000:2471
0000:2471      loc_0_2471:                                ; CODE XREF: sub_0_2441+1C↑j
0000:2471 DD 21 00 63      ld      ix, unk_0_6300                ; sub_0_2441+23↑j ...
0000:2475 11 05 00      ld      de, 5
0000:2478
0000:2478      loc_0_2478:                                ; CODE XREF: sub_0_2441+44↑j
0000:2478 7E      ; sub_0_2441+5A↑j ...
0000:2478      ld      a, (hl)
0000:2479 A7      and      a
0000:247A CA 88 24      jp      z, loc_0_2488
0000:247D 3D      dec      a
0000:247E CA 9E 24      jp      z, loc_0_249E
0000:2481 FE A9      cp      0A9h ; '@'
0000:2483 C8      ret      z
0000:2484 19      add      hl, de
0000:2485 C3 78 24      jp      loc_0_2478
0000:2488
0000:2488      ;
0000:2488
0000:2488      loc_0_2488:                                ; CODE XREF: sub_0_2441+39↑j
0000:2488 23      inc      hl
0000:2489 7E      ld      a, (hl)
0000:248A DD 77 00      ld      (ix+0), a
0000:248D 23      inc      hl
0000:248E 7E      ld      a, (hl)
0000:248F DD 77 15      ld      (ix+15h), a
0000:2492 23      inc      hl
0000:2493 23      inc      hl
0000:2494 7E      ld      a, (hl)
0000:2495 DD 77 2A      ld      (ix+2Ah), a
0000:2498 DD 23      inc      ix
0000:249A 23      inc      hl
0000:249B C3 78 24      jp      loc_0_2478
0000:249E
0000:249E      ;
0000:249E
0000:249E      loc_0_249E:                                ; CODE XREF: sub_0_2441+3D↑j
0000:249E 23      inc      hl
0000:249F 7E      ld      a, (hl)
0000:24A0 FD 77 00      ld      (iy+0), a
0000:24A3 23      inc      hl
0000:24A4 7E      ld      a, (hl)
0000:24A5 FD 77 15      ld      (iy+15h), a
0000:24A8 23      inc      hl
0000:24A9 23      inc      hl
0000:24AA 7E      ld      a, (hl)
0000:24AB FD 77 2A      ld      (iy+2Ah), a
0000:24AE FD 23      inc      iy
0000:24B0 23      inc      hl
0000:24B1 C3 78 24      jp      loc_0_2478
0000:24B1      ; End of function sub_0_2441
0000:24B1
0000:24B4
0000:24B4      ; ██████████ S U B R O U T I N E ██████████
0000:24B4
0000:24B4      sub_0_24B4:                                ; CODE XREF: sub_0_1F72+A2↑p
0000:24B4 DD 7E 05      ; sub_0_1F72+F6↑p ...
0000:24B4      ld      a, (ix+5)
0000:24B7 FE E8      cp      0E8h ; 'P'
0000:24B9 D8      ret      c
0000:24BA DD 7E 03      ld      a, (ix+3)
0000:24BD FE 2A      cp      2Ah ; '*'
0000:24BF D0      ret      nc
0000:24C0 FE 20      cp      20h ; ' '
0000:24C2 D8      ret      c

```

```

0000:24C3 DD 7E 15      ld      a, (ix+15h)
0000:24C6 A7              and     a
0000:24C7 CA D0 24      jp      z, loc_0_24D0
0000:24CA 3E 03         ld      a, 3
0000:24CC 32 B9 62      ld      (unk_0_62B9), a
0000:24CF AF           xor     a
0000:24D0
0000:24D0      loc_0_24D0:                                ; CODE XREF: sub_0_24B4+13|j
0000:24D0 DD 77 00      ld      (ix+0), a
0000:24D3 DD 77 03      ld      (ix+3), a
0000:24D6 21 82 60      hl, digital_snd_tmr_thump
0000:24D9 36 03         ld      (hl), 3                                ; tmr=3
0000:24DB E1          pop     hl
0000:24DC 3A 48 63      ld      a, (unk_0_6348)
0000:24DF A7              and     a
0000:24E0 C2 BA 21      jp      nz, loc_0_21BA
0000:24E3 3C           inc     a
0000:24E4 32 48 63      ld      (unk_0_6348), a
0000:24E7 C3 BA 21      jp      loc_0_21BA
0000:24E7      ; End of function sub_0_24B4
0000:24E7
0000:24EA
0000:24EA      ; ██████████ S U B R O U T I N E ██████████
0000:24EA
0000:24EA      sub_0_24EA:                                ; CODE XREF: 0000:1992|p
0000:24EA 3E 02         ld      a, 2
0000:24EC F7          rst     30h                                ; return if level bit not set
0000:24ED CD 23 25      call    sub_0_2523
0000:24F0 CD 91 25      call    sub_0_2591
0000:24F3 DD 21 A0 65      ld      ix, unk_0_65A0
0000:24F7 06 06         ld      b, 6                                ; 6 sprites to update
0000:24F9 21 B8 69      ld      hl, soft_sprite_ram+0B8h
0000:24FC
0000:24FC      loc_0_24FC:                                ; CODE XREF: sub_0_24EA+2F|j
0000:24FC DD 7E 00      ld      a, (ix+0)
0000:24FF A7              and     a
0000:2500 CA 1C 25      jp      z, loc_0_251C
0000:2503 DD 7E 03      ld      a, (ix+3)                                ; sprite X
0000:2506 77          ld      (hl), a
0000:2507 2C          inc     l
0000:2508 DD 7E 07      ld      a, (ix+7)                                ; sprite tile #
0000:250B 77          ld      (hl), a
0000:250C 2C          inc     l
0000:250D DD 7E 08      ld      a, (ix+8)                                ; sprite v flip & palette
0000:2510 77          ld      (hl), a
0000:2511 2C          inc     l
0000:2512 DD 7E 05      ld      a, (ix+5)                                ; sprite Y
0000:2515 77          ld      (hl), a
0000:2516 2C          inc     l
0000:2517
0000:2517      loc_0_2517:                                ; CODE XREF: sub_0_24EA+36|j
0000:2517 DD 19          add     ix, de
0000:2519 10 E1      djnz    loc_0_24FC
0000:251B C9          ret
0000:251C
0000:251C      ; _____
0000:251C      loc_0_251C:                                ; CODE XREF: sub_0_24EA+16|j
0000:251C 7D          ld      a, 1
0000:251D C6 04          add     a, 4
0000:251F 6F          ld      l, a
0000:2520 C3 17 25      jp      loc_0_2517
0000:2520      ; End of function sub_0_24EA
0000:2520
0000:2523
0000:2523      ; ██████████ S U B R O U T I N E ██████████
0000:2523
0000:2523      sub_0_2523:                                ; CODE XREF: sub_0_24EA+3|p
0000:2523 21 9B 63      ld      hl, unk_0_639B
0000:2526 7E          ld      a, (hl)
0000:2527 A7              and     a
0000:2528 C2 8F 25      jp      nz, loc_0_258F
0000:252B 3A 9A 63      ld      a, (unk_0_639A)
0000:252E A7              and     a
0000:252F C8          ret     z
0000:2530 06 06         ld      b, 6
0000:2532 11 10 00      ld      de, 10h
0000:2535 DD 21 A0 65      ld      ix, unk_0_65A0
0000:2539
0000:2539      loc_0_2539:                                ; CODE XREF: sub_0_2523+1F|j
0000:2539 DD CB 00 46      bit     0, (ix+0)
0000:253D CA 45 25      jp      z, loc_0_2545
0000:2540 DD 19          add     ix, de
0000:2542 10 F5      djnz    loc_0_2539
0000:2544 C9          ret
0000:2545
0000:2545      ; _____
0000:2545      loc_0_2545:                                ; CODE XREF: sub_0_2523+1A|j
0000:2545 CD 57 00      call    rand
0000:2548 FE 60          cp      60h ; '0'
0000:254A DE 36 05 7C      ld      (ix+5), 7Ch ; '7'
0000:254E DA 58 25      jp      c, loc_0_2558
0000:2551 3A A3 62      ld      a, (unk_0_62A3)
0000:2554 3D          dec     a
0000:2555 C2 6E 25      jp      nz, loc_0_256E
0000:2558
0000:2558      loc_0_2558:                                ; CODE XREF: sub_0_2523+2B|j
0000:2558 DD 36 05 CC      ld      (ix+5), 0CCh ; 'C'
0000:255C 3A A6 62      ld      a, (unk_0_62A6)
0000:255F 07          rlca
0000:2560
0000:2560      loc_0_2560:                                ; CODE XREF: sub_0_2523+50|j
0000:2560 DD 36 03 07      ld      (ix+3), 7
0000:2564 D2 76 25      jp      nc, loc_0_2576
0000:2567 DD 36 03 F8      ld      (ix+3), 0F8h ; '8'
0000:256B C3 76 25      jp      loc_0_2576
0000:256E
0000:256E      ; _____
0000:256E      loc_0_256E:                                ; CODE XREF: sub_0_2523+32|j
0000:256E CD 57 00      call    rand
0000:2571 FE 68          cp      68h ; 'h'
0000:2573 C3 60 25      jp      loc_0_2560
0000:2576
0000:2576      ; _____
0000:2576      loc_0_2576:                                ; CODE XREF: sub_0_2523+41|j

```



```

0000:2576 DD 36 00 01          ld      (ix+0), 1          ; sub_0_2523+48|j
0000:2576          ld      (ix+7), 4Bh ; 'K'          ; cement pie sprite tile
0000:257A DD 36 07 4B          ld      (ix+9), 8
0000:257E DD 36 09 08          ld      (ix+0Ah), 3
0000:2582 DD 36 0A 03          ld      a, 7Ch ; '|'
0000:2586 3E 7C              ld      (unk_0_639B), a
0000:2588 32 9B 63          xor      a
0000:258B AF              ld      (unk_0_639A), a
0000:258C 32 9A 63
0000:258F          loc_0_258F:          ; CODE XREF: sub_0_2523+5|j
0000:258F          dec      (hl)
0000:258F 35          ret
0000:2590 C9          ; End of function sub_0_2523
0000:2590
0000:2591          ; ██████████ S U B R O U T I N E ██████████
0000:2591
0000:2591          sub_0_2591:          ; CODE XREF: sub_0_24EA+6|p
0000:2591 DD 21 A0 65          ld      ix, unk_0_65A0
0000:2595 11 10 00          ld      de, 10h
0000:2598 06 06          ld      b, 6
0000:259A          loc_0_259A:          ; CODE XREF: sub_0_2591+2C|j
0000:259A DD CB 00 46          bit     0, (ix+0)
0000:259E CA BB 25          jp      z, loc_0_25BB
0000:25A1 DD 7E 03          ld      a, (ix+3)
0000:25A4 67          ld      h, a
0000:25A5 C6 07          add     a, 7
0000:25A7 FE 0E          cp      0Eh
0000:25A9 DA D6 25          jp      c, loc_0_25D6
0000:25AC DD 7E 05          ld      a, (ix+5)
0000:25AF FE 7C          cp      7Ch ; '|'
0000:25B1 CA C0 25          jp      z, loc_0_25C0
0000:25B4 3A A6 63          ld      a, (unk_0_63A6)
0000:25B7 84          add     a, h
0000:25B8 DD 77 03          ld      (ix+3), a
0000:25BB          loc_0_25BB:          ; CODE XREF: sub_0_2591+D|j
0000:25BB          ; sub_0_2591+42|j ...
0000:25BB DD 19          add     ix, de
0000:25BD 10 DB          djnz   loc_0_259A
0000:25BF C9          ret
0000:25C0          ;
0000:25C0          loc_0_25C0:          ; CODE XREF: sub_0_2591+20|j
0000:25C0 7C          ld      a, h
0000:25C1 FE 80          cp      80h ; 'C'
0000:25C3 CA D6 25          jp      z, loc_0_25D6
0000:25C6 3A A5 63          ld      a, (unk_0_63A5)
0000:25C9 D2 CF 25          jp      nc, loc_0_25CF
0000:25CC 3A A4 63          ld      a, (unk_0_63A4)
0000:25CF          loc_0_25CF:          ; CODE XREF: sub_0_2591+38|j
0000:25CF 84          add     a, h
0000:25D0 DD 77 03          ld      (ix+3), a
0000:25D3 C3 BB 25          jp      loc_0_25BB
0000:25D6          ;
0000:25D6          loc_0_25D6:          ; CODE XREF: sub_0_2591+18|j
0000:25D6          ; sub_0_2591+32|j
0000:25D6 21 B8 69          ld      hl, soft_sprite_ram+0B8h
0000:25D9 3E 06          ld      a, 6
0000:25DB 90          sub     b
0000:25DC          loc_0_25DC:          ; CODE XREF: sub_0_2591+53|j
0000:25DC CA E7 25          jp      z, loc_0_25E7
0000:25DF 2C          inc     l
0000:25E0 2C          inc     l
0000:25E1 2C          inc     l
0000:25E2 2C          inc     l
0000:25E3 3D          dec     a
0000:25E4 C3 DC 25          jp      loc_0_25DC
0000:25E7          ;
0000:25E7          loc_0_25E7:          ; CODE XREF: sub_0_2591+4B|j
0000:25E7 AF          xor      a
0000:25E8 DD 77 00          ld      (ix+0), a
0000:25EB DD 77 03          ld      (ix+3), a
0000:25EE 77          ld      (hl), a
0000:25EF C3 BB 25          jp      loc_0_25BB
0000:25EF          ; End of function sub_0_2591
0000:25F2          ; ██████████ S U B R O U T I N E ██████████
0000:25F2
0000:25F2          sub_0_25F2:          ; CODE XREF: 0000:19AA|p
0000:25F2 3E 02          ld      a, 2
0000:25F4 F7          rst     30h          ; return if level bit not set
0000:25F5 CD 02 26          call    sub_0_2602
0000:25F8 CD 2F 26          call    sub_0_262F
0000:25FB CD 79 26          call    sub_0_2679
0000:25FE CD D3 2A          call    sub_0_2AD3
0000:2601 C9          ret
0000:2601          ; End of function sub_0_25F2
0000:2601
0000:2602          ; ██████████ S U B R O U T I N E ██████████
0000:2602
0000:2602          sub_0_2602:          ; CODE XREF: 0000:16D5|p
0000:2602          ; sub_0_25F2+3|p
0000:2602          ld      a, (gen_purpose_timer)
0000:2605 0F          rrca
0000:2606 DA 16 26          jp      c, loc_0_2616
0000:2609 21 A0 62          ld      hl, unk_0_62A0
0000:260C 35          dec     (hl)
0000:260D C2 16 26          jp      nz, loc_0_2616
0000:2610 36 80          ld      (hl), 80h ; 'C'
0000:2612 2C          inc     l
0000:2613 CD DE 26          call    sub_0_26DE
0000:2616          loc_0_2616:          ; CODE XREF: sub_0_2602+4|j
0000:2616          ; sub_0_2602+B|j
0000:2616 21 A1 62

```

```

0000:2616          ld      hl, unk_0_62A1
0000:2619 CD E9 26    call   sub_0_26E9
0000:261C 32 A3 63     ld      (unk_0_63A3), a
0000:261F 3A 1A 60     ld      a, (gen_purpose_timer)
0000:2622 E6 1F        and     1Fh
0000:2624 FE 01     cp      1
0000:2626 C0          ret     nz
0000:2627 11 E4 69     ld      de, soft_sprite_ram+0E4h
0000:262A EB        ex      de, hl
0000:262B CD A6 26    call   sub_0_26A6
0000:262E C9          ret
0000:262E          ; End of function sub_0_2602
0000:262E
0000:262F          ; ██████████ SUBROUTINE ██████████
0000:262F
0000:262F          sub_0_262F:                                ; CODE XREF: sub_0_25F2+6↑p
0000:262F 21 A3 62     ld      hl, unk_0_62A3
0000:2632 3A 05 62     ld      a, (mario_x_coord)
0000:2635 FE C0     cp      0C0h ; 'L'
0000:2637 DA 6F 26    jp      c, loc_0_266F
0000:263A 3A 1A 60     ld      a, (gen_purpose_timer)
0000:263D 0F        rrca
0000:263E DA 4C 26    jp      c, loc_0_264C
0000:2641 2D        dec     l
0000:2642 35        dec     (hl)
0000:2643 C2 4C 26    jp      nz, loc_0_264C
0000:2646 36 C0     ld      (hl), 0C0h ; 'L'
0000:2648 2C        inc     l
0000:2649 CD DE 26    call   sub_0_26DE
0000:264C          loc_0_264C:                                ; CODE XREF: sub_0_262F+F↑j
0000:264C          ; sub_0_262F+14↑j ...
0000:264C          ld      hl, unk_0_62A3
0000:264F CD E9 26    call   sub_0_26E9
0000:2652 32 A5 63     ld      (unk_0_63A5), a
0000:2655 ED 44     neg
0000:2657 32 A4 63     ld      (unk_0_63A4), a
0000:265A 3A 1A 60     ld      a, (gen_purpose_timer)
0000:265D E6 1F        and     1Fh
0000:265F C0          ret     nz
0000:2660 2D        dec     l
0000:2661 11 EC 69     ld      de, soft_sprite_ram+0ECh
0000:2664 EB        ex      de, hl
0000:2665 CD A6 26    call   sub_0_26A6
0000:2668 E6 7F        and     7Fh ; ' '
0000:266A 21 ED 69     ld      hl, soft_sprite_ram+0EDh
0000:266D 77        ld      (hl), a
0000:266E C9          ret
0000:266F          ;
0000:266F          loc_0_266F:                                ; CODE XREF: sub_0_262F+8↑j
0000:266F CB 7E        bit     7, (hl)
0000:2671 C2 4C 26    jp      nz, loc_0_264C
0000:2674 36 FF     ld      (hl), 0FFh
0000:2676 C3 4C 26    jp      loc_0_264C
0000:2676          ; End of function sub_0_262F
0000:2676
0000:2679          ; ██████████ SUBROUTINE ██████████
0000:2679
0000:2679          sub_0_2679:                                ; CODE XREF: sub_0_25F2+9↑p
0000:2679 3A 1A 60     ld      a, (gen_purpose_timer)
0000:267C 0F        rrca
0000:267D DA 8D 26    jp      c, loc_0_268D
0000:2680 21 A5 62     ld      hl, unk_0_62A5
0000:2683 35        dec     (hl)
0000:2684 C2 8D 26    jp      nz, loc_0_268D
0000:2687 36 FF     ld      (hl), 0FFh
0000:2689 2C        inc     l
0000:268A CD DE 26    call   sub_0_26DE
0000:268D          loc_0_268D:                                ; CODE XREF: sub_0_2679+4↑j
0000:268D          ; sub_0_2679+B↑j
0000:268D          ld      hl, unk_0_62A6
0000:2690 CD E9 26    call   sub_0_26E9
0000:2693 32 A6 63     ld      (unk_0_63A6), a
0000:2696 3A 1A 60     ld      a, (gen_purpose_timer)
0000:2699 E6 1F        and     1Fh
0000:269B FE 02     cp      2
0000:269D C0          ret     nz
0000:269E 11 F4 69     ld      de, soft_sprite_ram+0F4h
0000:26A1 EB        ex      de, hl
0000:26A2 CD A6 26    call   sub_0_26A6
0000:26A5 C9          ret
0000:26A5          ; End of function sub_0_2679
0000:26A5
0000:26A6          ; ██████████ SUBROUTINE ██████████
0000:26A6
0000:26A6          sub_0_26A6:                                ; CODE XREF: sub_0_2602+29↑p
0000:26A6          ; sub_0_262F+36↑p ...
0000:26A6 2C        inc     l
0000:26A7 1A        ld      a, (de)
0000:26A8 17        rla
0000:26A9 DA C5 26    jp      c, loc_0_26C5
0000:26AC 7E        ld      a, (hl)
0000:26AD 3C        inc     a
0000:26AE FE 53     cp      53h ; 'S'
0000:26B0 C2 B5 26    jp      nz, loc_0_26B5
0000:26B3 3E 50     ld      a, 50h ; 'P'
0000:26B5          loc_0_26B5:                                ; CODE XREF: sub_0_26A6+A↑j
0000:26B5          ld      (hl), a
0000:26B6 7D        ld      a, 1
0000:26B7 C6 04     add     a, 4
0000:26B9 6F        ld      l, a
0000:26BA 7E        ld      a, (hl)
0000:26BB 3D        dec     a
0000:26BC FE CF     cp      0CFh ; 'q'
0000:26BE C2 C3 26    jp      nz, loc_0_26C3
0000:26C1 3E D2     ld      a, 0D2h ; 'R'
0000:26C3

```

```

0000:26C3      loc_0_26C3:      ld      (hl), a      ; CODE XREF: sub_0_26A6+18↑j
0000:26C3 77      ret
0000:26C4 C9      ;
0000:26C5      loc_0_26C5:      ld      a, (hl)      ; CODE XREF: sub_0_26A6+3↑j
0000:26C5 7E      dec     a
0000:26C6 3D      cp      4Fh ; 'O'
0000:26C7 FE 4F      jp      nz, loc_0_26CE
0000:26C9 C2 CE 26      ld      a, 52h ; 'R'
0000:26CC 3E 52
0000:26CE      loc_0_26CE:      ld      (hl), a      ; CODE XREF: sub_0_26A6+23↑j
0000:26CE 77      ld      a, 1
0000:26CF 7D      add     a, 4
0000:26D0 C6 04      ld      l, a
0000:26D2 6F      ld      a, (hl)
0000:26D3 7E      inc     a
0000:26D4 3C      cp      0D3h ; 'ë'
0000:26D5 FE D3      jp      nz, loc_0_26DC
0000:26D7 C2 DC 26      ld      a, 0D0h ; 'ð'
0000:26DA 3E D0
0000:26DC      loc_0_26DC:      ld      (hl), a      ; CODE XREF: sub_0_26A6+31↑j
0000:26DC 77      ret
0000:26DD C9      ; End of function sub_0_26A6
0000:26DD
0000:26DE      ; ██████████ SUBROUTINE ██████████
0000:26DE
0000:26DE      sub_0_26DE:      ; CODE XREF: sub_0_2602+11↑p
0000:26DE CB 7E      ; sub_0_262F+1A↑p ...
0000:26DE      bit     7, (hl)
0000:26E0 CA E6 26      jp      z, loc_0_26E6
0000:26E3 36 02      ld      (hl), 2
0000:26E5 C9      ret
0000:26E6      loc_0_26E6:      ld      (hl), 0FEh ; '■'
0000:26E6 36 FE      ret
0000:26E8 C9      ; End of function sub_0_26DE
0000:26E8
0000:26E9      ; ██████████ SUBROUTINE ██████████
0000:26E9
0000:26E9      sub_0_26E9:      ; CODE XREF: sub_0_2602+17↑p
0000:26E9 3A 1A 60      ; sub_0_262F+20↑p ...
0000:26E9      ld      a, (gen_purpose_timer)
0000:26EC E6 01      and     1
0000:26EE C8      ret     z
0000:26EF CB 7E      bit     7, (hl)
0000:26F1 3E FF      ld      a, 0FFh
0000:26F3 C2 F8 26      jp      nz, loc_0_26F8
0000:26F6 3E 01      ld      a, 1
0000:26F8      loc_0_26F8:      ld      (hl), a      ; CODE XREF: sub_0_26E9+A↑j
0000:26F8 77      ret
0000:26F9 C9      ; End of function sub_0_26E9
0000:26F9
0000:26FA      ; ██████████ SUBROUTINE ██████████
0000:26FA
0000:26FA      sub_0_26FA:      ; CODE XREF: 0000:19A7↑p
0000:26FA 3E 04      ld      a, 4
0000:26FC F7      rst     30h      ; return if level bit not set
0000:26FD 3A 05 62      ld      a, (mario_x_coord)
0000:2700 FE F0      cp      0F0h ; '-'
0000:2702 D2 7F 27      jp      nc, mario_dies_on_elevator      ; make mario die
0000:2705 3A 29 62      ld      a, (level)
0000:2708 3D      dec     a
0000:2709 3A 1A 60      ld      a, (gen_purpose_timer)
0000:270C C2 1A 27      jp      nz, loc_0_271A
0000:270F E6 03      and     3
0000:2711 FE 01      cp      1
0000:2713 CA 1E 27      jp      z, loc_0_271E
0000:2716 DA 22 27      jp      c, loc_0_2722
0000:2719 C9      ret
0000:271A      loc_0_271A:      rrca
0000:271A 0F      jp      c, loc_0_2722
0000:271B DA 22 27
0000:271E      loc_0_271E:      call    sub_0_2745      ; CODE XREF: sub_0_26FA+19↑j
0000:2721 C9      ret
0000:2722      loc_0_2722:      call    sub_0_2797      ; CODE XREF: sub_0_26FA+1C↑j
0000:2722      call    sub_0_27DA      ; sub_0_26FA+21↑j
0000:2725 CD DA 27      ld      b, 6      ; six elevators
0000:2728 06 06      ld      de, 10h
0000:272A 11 10 00      ld      hl, soft_sprite_ram+58h
0000:272D 21 58 69      ld      ix, unk_0_6600
0000:2730 DD 21 00 66
0000:2734      loc_0_2734:      ld      a, (ix+3)      ; CODE XREF: sub_0_26FA+48↑j
0000:2734 DD 7E 03      ld      (hl), a      ; store coordinates
0000:2737 77      inc     l
0000:2738 2C      inc     l
0000:2739 2C      inc     l
0000:273A 2C      inc     l
0000:273B DD 7E 05      ld      a, (ix+5)
0000:273E 77      ld      (hl), a
0000:273F 2C      inc     l
0000:2740 DD 19      add     ix, de
0000:2742 10 F0      djnz    loc_0_2734
0000:2744 C9      ret
0000:2744      ; End of function sub_0_26FA

```

```

0000:2744
0000:2745
0000:2745 ; SUBROUTINE
0000:2745
0000:2745 sub_0_2745: ; CODE XREF: sub_0_26FA+24|p
0000:2745 3A 98 63 ld a, (mario_on_elevator) ; on elevator?
0000:2748 A7 and a ; no, return
0000:2749 C8 ret
0000:274A 3A 16 62 ld a, (mario_jumping) ; jumping?
0000:274D A7 and a ; yes, return
0000:274E C0 ret
0000:274F 3A 03 62 ld a, (mario_y_coord)
0000:2752 FE 2C cp 2Ch ; 'c'
0000:2754 DA 66 27 jp c, loc_0_2766 ; not not elevator
0000:2757 FE 43 cp 43h ; 'C'
0000:2759 DA 6F 27 jp c, loc_0_276F ; on left elevator
0000:275C FE 6C cp 6Ch ; 'l'
0000:275E DA 66 27 jp c, loc_0_2766 ; not on elevator
0000:2761 FE 83 cp 83h ; 'a'
0000:2763 DA 87 27 jp c, loc_0_2787 ; on right elevator
0000:2766
0000:2766 loc_0_2766: ; CODE XREF: sub_0_2745+F|j
0000:2766 AF ; sub_0_2745+19|j
0000:2766 ; mark off elevator
0000:2766 xor a
0000:2767 32 98 63 ld (mario_on_elevator), a
0000:276A 3C inc a
0000:276B 32 21 62 ld (unk_0_6221), a
0000:276E C9 ret
0000:276F
0000:276F ;
0000:276F loc_0_276F: ; CODE XREF: sub_0_2745+14|j
0000:276F 3A 05 62 ld a, (mario_x_coord)
0000:2772 FE 71 cp 71h ; 'q'
0000:2774 DA 7F 27 jp c, mario_dies_on_elevator ; make mario die
0000:2777 3D dec a ; on upwards moving elevator
0000:2778 32 05 62 ld (mario_x_coord), a
0000:277B 32 4F 69 ld (soft_sprite_ram+4Fh), a
0000:277E C9 ret
0000:277F
0000:277F ;
0000:277F mario_dies_on_elevator: ; CODE XREF: sub_0_26FA+8|j
0000:277F AF ; sub_0_2745+2F|j ...
0000:277F xor a
0000:2780 32 00 62 ld (mario_alive_flag), a
0000:2783 32 98 63 ld (mario_on_elevator), a
0000:2786 C9 ret
0000:2787
0000:2787 ;
0000:2787 loc_0_2787: ; CODE XREF: sub_0_2745+1E|j
0000:2787 3A 05 62 ld a, (mario_x_coord)
0000:278A FE E8 cp 0E8h ; 'b'
0000:278C D2 7F 27 jp nc, mario_dies_on_elevator
0000:278F 3C inc a ; on downwards moving elevator
0000:2790 32 05 62 ld (mario_x_coord), a
0000:2793 32 4F 69 ld (soft_sprite_ram+4Fh), a
0000:2796 C9 ret
0000:2796 ; End of function sub_0_2745
0000:2796
0000:2797
0000:2797 ; SUBROUTINE
0000:2797
0000:2797 sub_0_2797: ; CODE XREF: sub_0_26FA+28|p
0000:2797 06 06 ld b, 6 ; move elevators to the right side
0000:2799 11 10 00 ld de, 10h
0000:279C DD 21 00 66 ld ix, unk_0_6600
0000:27A0
0000:27A0 loc_0_27A0: ; CODE XREF: sub_0_2797+2D|j
0000:27A0 DD CB 00 46 bit 0, (ix+0)
0000:27A4 CA C2 27 jp z, loc_0_27C2
0000:27A7 DD CB 0D 5E bit 3, (ix+0Dh)
0000:27AB CA C7 27 jp z, loc_0_27C7
0000:27AE DD 7E 05 ld a, (ix+5)
0000:27B1 3D dec a
0000:27B2 DD 77 05 ld (ix+5), a
0000:27B5 FE 60 cp 60h ; 'f'
0000:27B7 C2 C2 27 jp nz, loc_0_27C2
0000:27BA DD 36 03 77 ld (ix+3), 77h ; 'w'
0000:27BE DD 36 0D 04 ld (ix+0Dh), 4
0000:27C2
0000:27C2 loc_0_27C2: ; CODE XREF: sub_0_2797+D|j
0000:27C2 DD 19 ; sub_0_2797+20|j ...
0000:27C2 add ix, de
0000:27C4 10 DA djnz loc_0_27A0
0000:27C6 C9 ret
0000:27C7
0000:27C7 ;
0000:27C7 loc_0_27C7: ; CODE XREF: sub_0_2797+14|j
0000:27C7 DD 7E 05 ld a, (ix+5)
0000:27CA 3C inc a
0000:27CB DD 77 05 ld (ix+5), a
0000:27CE FE F8 cp 0F8h ; 'o'
0000:27D0 C2 C2 27 jp nz, loc_0_27C2
0000:27D3 DD 36 00 00 ld (ix+0), 0
0000:27D7 C3 C2 27 jp loc_0_27C2
0000:27D7 ; End of function sub_0_2797
0000:27D7
0000:27DA
0000:27DA ; SUBROUTINE
0000:27DA
0000:27DA sub_0_27DA: ; CODE XREF: sub_0_26FA+2B|p
0000:27DA 21 A7 62 ld hl, unk_0_62A7 ; move elevators to the left side
0000:27DD 7E ld a, (hl)
0000:27DE A7 and a
0000:27DF C2 06 28 jp nz, loc_0_2806
0000:27E2 06 06 ld b, 6
0000:27E4 DD 21 00 66 ld ix, unk_0_6600
0000:27E8
0000:27E8 loc_0_27E8: ; CODE XREF: sub_0_27DA+17|j
0000:27E8 DD CB 00 46 bit 0, (ix+0)
0000:27EC CA F4 27 jp z, loc_0_27F4
0000:27EF DD 19 add ix, de
0000:27F1 10 F5 djnz loc_0_27E8
0000:27F3 C9 ret

```

```

0000:27F4      ;
0000:27F4
0000:27F4      loc_0_27F4:                                ; CODE XREF: sub_0_27DA+12↑j
0000:27F4 DD 36 00 01      ld      (ix+0), 1
0000:27F8 DD 36 03 37      ld      (ix+3), 37h ; '7'
0000:27FC DD 36 05 F8      ld      (ix+5), 0F8h ; 'o'
0000:2800 DD 36 0D 08      ld      (ix+0Dh), 8
0000:2804 36 34          ld      (hl), 34h ; '4'
0000:2806
0000:2806      loc_0_2806:                                ; CODE XREF: sub_0_27DA+5↑j
0000:2806 35          dec      (hl)
0000:2807 C9          ret
0000:2807      ; End of function sub_0_27DA
0000:2808
0000:2808      ; SUBROUTINE
0000:2808
0000:2808      sub_0_2808:                                ; CODE XREF: 0000:19B3↑p
0000:2808 FD 21 00 62      ld      iy, mario_alive_flag
0000:280C 3A 05 62      ld      a, (mario_x_coord)
0000:280F 4F          ld      c, a
0000:2810 21 07 04      ld      hl, 407h
0000:2813 CD 6F 28      call   sub_0_286F
0000:2816 A7          and     a
0000:2817 C8          ret     z
0000:2818 3D          dec     a ; die
0000:2819 32 00 62      ld      (mario_alive_flag), a
0000:281C C9          ret
0000:281C      ; End of function sub_0_2808
0000:281D
0000:281D      ; SUBROUTINE
0000:281D
0000:281D      sub_0_281D:                                ; CODE XREF: 0000:19B6↑p
0000:281D 06 02          ld      b, 2
0000:281F 11 10 00      ld      de, 10h
0000:2822 FD 21 80 66      ld      iy, unk_0_6680 ; hammer character data
0000:2826
0000:2826      loc_0_2826:                                ; CODE XREF: sub_0_281D+12↑j
0000:2826 FD CB 01 46      bit     0, (iy+1)
0000:282A C2 32 28      jp      nz, loc_0_2832
0000:282D FD 19          add     iy, de
0000:282F 10 F5      djnz    loc_0_2826
0000:2831 C9          ret
0000:2832
0000:2832      loc_0_2832:                                ; CODE XREF: sub_0_281D+D↑j
0000:2832 FD 4E 05      ld      c, (iy+5)
0000:2835 FD 66 09      ld      h, (iy+9)
0000:2838 FD 6E 0A      ld      l, (iy+0Ah)
0000:283B CD 6F 28      call   sub_0_286F
0000:283E A7          and     a
0000:283F C8          ret     z
0000:2840 32 50 63      ld      (unk_0_6350), a
0000:2843 3A B9 63      ld      a, (unk_0_63B9)
0000:2846 90          sub     b
0000:2847 32 54 63      ld      (unk_0_6354), a
0000:284A 7B          ld      a, e
0000:284B 32 53 63      ld      (unk_0_6353), a
0000:284E DD 22 51 63      ld      (unk_0_6351), ix
0000:2852 C9          ret
0000:2852      ; End of function sub_0_281D
0000:2853
0000:2853      ; SUBROUTINE
0000:2853
0000:2853      sub_0_2853:                                ; CODE XREF: sub_0_1AC3+15D↑p
0000:2853 FD 21 00 62      ld      iy, mario_alive_flag
0000:2857 3A 05 62      ld      a, (mario_x_coord)
0000:285A
0000:285A      loc_0_285A:
0000:285A C6 0C          add     a, 0Ch
0000:285C 4F          ld      c, a
0000:285D 3A 10 60      ld      a, (controller_in)
0000:2860 E6 03          and     3 ; left/right only
0000:2862 21 08 05      ld      hl, 508h
0000:2865 CA 6B 28      jp      z, loc_0_286B ; not left/right
0000:2868 21 08 13      ld      hl, 1308h
0000:286B
0000:286B      loc_0_286B:                                ; CODE XREF: sub_0_2853+12↑j
0000:286B CD 88 3E      call   sub_0_3E88
0000:286E C9          ret
0000:286E      ; End of function sub_0_2853
0000:286F
0000:286F      ; SUBROUTINE
0000:286F
0000:286F      sub_0_286F:                                ; CODE XREF: sub_0_2808+B↑p
0000:286F 3A 27 62      ld      a, (level_type) ; sub_0_281D+1E↑p
0000:2872 E5          push    hl
0000:2873 EF          rst     28h ; go!
0000:2873
0000:2873      ;
0000:2874 00 00          .dw 0 ; Jump table
0000:2876 80 28          .dw 11_check_hammer_hit
0000:2878 B0 28          .dw 12_check_hammer_hit
0000:287A E0 28          .dw 13_check_hammer_hit
0000:287C 01 29          .dw 14_check_hammer_hit
0000:287E 00 00          .dw 0
0000:2880
0000:2880      11_check_hammer_hit: ; DATA XREF: sub_0_286F+7↑o
0000:2880 E1          pop     hl
0000:2881 06 0A      ld      b, 0Ah
0000:2883 78          ld      a, b
0000:2884 32 B9 63      ld      (unk_0_63B9), a
0000:2887 11 20 00      ld      de, 20h ; ' '
0000:288A DD 21 00 67      ld      ix, unk_0_6700
0000:288E CD 13 29      call   sub_0_2913
0000:2891 06 05      ld      b, 5
0000:2893 78          ld      a, b
0000:2894 32 B9 63      ld      (unk_0_63B9), a

```

```

0000:2897 1E 20          ld      e, 20h ; ' '
0000:2899 DD 21 00 64      ld      ix, unk_0_6400          ; fireball character data
0000:289D CD 13 29      call   sub_0_2913
0000:28A0 06 01          ld      b, 1
0000:28A2 78          ld      a, b
0000:28A3 32 B9 63      ld      (unk_0_63B9), a
0000:28A6 1E 00          ld      e, 0
0000:28A8 DD 21 A0 66      ld      ix, unk_0_66A0
0000:28AC CD 13 29      call   sub_0_2913
0000:28AF C9          ret
0000:28AF          ; End of function sub_0_286F
0000:28AF
0000:28B0          ;
0000:28B0
0000:28B0          12_check_hammer_hit:          ; DATA XREF: sub_0_286F+9↑o
0000:28B0 E1          ; sub_0_3E88+9↑o
0000:28B0          pop      hl
0000:28B1 06 05          ld      b, 5
0000:28B3 78          ld      a, b
0000:28B4 32 B9 63      ld      (unk_0_63B9), a
0000:28B7 11 20 00      ld      de, 20h ; ' '
0000:28BA DD 21 00 64      ld      ix, unk_0_6400          ; fireball character data
0000:28BE CD 13 29      call   sub_0_2913
0000:28C1 06 06          ld      b, 6
0000:28C3 78          ld      a, b
0000:28C4 32 B9 63      ld      (unk_0_63B9), a
0000:28C7 1E 10          ld      e, 10h
0000:28C9 DD 21 A0 65      ld      ix, unk_0_65A0
0000:28CD CD 13 29      call   sub_0_2913
0000:28D0 06 01          ld      b, 1
0000:28D2 78          ld      a, b
0000:28D3 32 B9 63      ld      (unk_0_63B9), a
0000:28D6 1E 00          ld      e, 0
0000:28D8 DD 21 A0 66      ld      ix, unk_0_66A0
0000:28DC CD 13 29      call   sub_0_2913
0000:28DF C9          ret
0000:28E0          ;
0000:28E0
0000:28E0          13_check_hammer_hit:          ; DATA XREF: sub_0_286F+B↑o
0000:28E0 E1          ; sub_0_3E88+B↑o
0000:28E0          pop      hl
0000:28E1 06 05          ld      b, 5
0000:28E3 78          ld      a, b
0000:28E4 32 B9 63      ld      (unk_0_63B9), a
0000:28E7 11 20 00      ld      de, 20h ; ' '
0000:28EA DD 21 00 64      ld      ix, unk_0_6400          ; fireball character data
0000:28EE CD 13 29      call   sub_0_2913
0000:28F1 06 0A          ld      b, 0Ah
0000:28F3 78          ld      a, b
0000:28F4 32 B9 63      ld      (unk_0_63B9), a
0000:28F7 1E 10          ld      e, 10h
0000:28F9 DD 21 00 65      ld      ix, unk_0_6500          ; check if hammer hits a spring
0000:28FD CD 13 29      call   sub_0_2913
0000:2900 C9          ret
0000:2901          ;
0000:2901
0000:2901          14_check_hammer_hit:          ; DATA XREF: sub_0_286F+D↑o
0000:2901 E1          ; sub_0_3E88+D↑o
0000:2901          pop      hl
0000:2902 06 07          ld      b, 7
0000:2904 78          ld      a, b
0000:2905 32 B9 63      ld      (unk_0_63B9), a
0000:2908 11 20 00      ld      de, 20h ; ' '
0000:290B DD 21 00 64      ld      ix, unk_0_6400          ; fireball character data
0000:290F CD 13 29      call   sub_0_2913
0000:2912 C9          ret
0000:2913          ;
0000:2913          SUBROUTINE
0000:2913
0000:2913          sub_0_2913:          ; CODE XREF: sub_0_286F+1F↑p
0000:2913 DD E5          ; sub_0_286F+2E↑p ...
0000:2913          push    ix
0000:2915          loc_0_2915:          ; CODE XREF: sub_0_2913+3B↑j
0000:2915 DD CB 00 46      bit     0, (ix+0)          ; check if hammer hits something else
0000:2919 CA 4C 29      jp      z, loc_0_294C
0000:291C 79          ld      a, c
0000:291D DD 96 05      sub     (ix+5)
0000:2920 D2 25 29      jp      nc, loc_0_2925
0000:2923 ED 44          neg
0000:2925          loc_0_2925:          ; CODE XREF: sub_0_2913+D↑j
0000:2925 3C          inc     a
0000:2926 95          sub     l
0000:2927 DA 30 29      jp      c, loc_0_2930
0000:292A DD 96 0A      sub     (ix+0Ah)
0000:292D D2 4C 29      jp      nc, loc_0_294C
0000:2930          loc_0_2930:          ; CODE XREF: sub_0_2913+14↑j
0000:2930 FD 7E 03      ld      a, (iy+3)
0000:2933 DD 96 03      sub     (ix+3)
0000:2936 D2 3B 29      jp      nc, loc_0_293B
0000:2939 ED 44          neg
0000:293B          loc_0_293B:          ; CODE XREF: sub_0_2913+23↑j
0000:293B 94          sub     h
0000:293C DA 45 29      jp      c, loc_0_2945
0000:293F DD 96 09      sub     (ix+9)
0000:2942 D2 4C 29      jp      nc, loc_0_294C
0000:2945          loc_0_2945:          ; CODE XREF: sub_0_2913+29↑j
0000:2945 3E 01          ld      a, 1
0000:2947 DD E1          pop     ix
0000:2949 33          inc     sp
0000:294A 33          inc     sp
0000:294B C9          ret
0000:294C          ;
0000:294C
0000:294C          loc_0_294C:          ; CODE XREF: sub_0_2913+6↑j
0000:294C DD 19          ; sub_0_2913+1A↑j ...
0000:294C          add     ix, de
0000:294E 10 C5      djnz   loc_0_2915
0000:2950 AF          xor     a
0000:2951 DD E1          pop     ix
0000:2953 C9          ret

```

```

0000:2953          ; End of function sub_0_2913
0000:2953
0000:2954          ; SUBROUTINE
0000:2954
0000:2954          sub_0_2954:                                ; CODE XREF: sub_0_1AC3+171|p
0000:2954 3E 0B          ld      a, 0Bh
0000:2956 F7          rst      30h          ; return if level bit not set
0000:2957 CD 74 29     call    sub_0_2974
0000:295A 32 18 62     ld      (unk_0_6218), a
0000:295D 0F          rrca
0000:295E 0F          rrca
0000:295F 32 85 60     ld      (digital_snd_tmr_barrel_jump_priz), a
0000:2962 78          ld      a, b
0000:2963 A7          and     a
0000:2964 C8          ret     z
0000:2965 FE 01          cp     1
0000:2967 CA 6F 29     jp     z, loc_0_296F
0000:296A DD 36 01 01   ld      (ix+1), 1
0000:296E C9          ret
0000:296F          ;
0000:296F
0000:296F          loc_0_296F:                                ; CODE XREF: sub_0_2954+13|j
0000:296F DD 36 11 01   ld      (ix+11h), 1
0000:2973 C9          ret
0000:2973          ; End of function sub_0_2954
0000:2973
0000:2974          ; SUBROUTINE
0000:2974
0000:2974          sub_0_2974:                                ; CODE XREF: sub_0_2954+3|p
0000:2974 FD 21 00 62     ld      iy, mario_alive_flag
0000:2978 3A 05 62     ld      a, (mario_x_coord)
0000:297B 4F          ld      c, a
0000:297C 21 08 04     ld      hl, 408h
0000:297F 06 02     ld      b, 2
0000:2981 11 10 00     ld      de, 10h
0000:2984 DD 21 80 66     ld      ix, unk_0_6680          ; hammer character data
0000:2988 CD 13 29     call    sub_0_2913
0000:298B C9          ret
0000:298B          ; End of function sub_0_2974
0000:298B
0000:298C          ; SUBROUTINE
0000:298C
0000:298C          sub_0_298C:                                ; CODE XREF: sub_0_3202+3C|p
0000:298C 2A C8 63     ld      hl, (unk_0_63C8)
0000:298F 7D          ld      a, l
0000:2990 C6 0E          add    a, 0Eh
0000:2992 6F          ld      l, a
0000:2993 56          ld      d, (hl)
0000:2994 2C          inc    l
0000:2995 7E          ld      a, (hl)
0000:2996 C6 0C          add    a, 0Ch
0000:2998 5F          ld      e, a
0000:2999 EB          ex     de, hl
0000:299A CD F0 2F     call    sub_0_2FF0
0000:299D 7E          ld      a, (hl)
0000:299E FE B0          cp     0B0h          ;
0000:29A0 DA AC 29     jp     c, loc_0_29AC
0000:29A3 E6 0F          and    0Fh
0000:29A5 FE 08          cp     8
0000:29A7 D2 AC 29     jp     nc, loc_0_29AC
0000:29AA AF          xor     a
0000:29AB C9          ret
0000:29AC          ;
0000:29AC
0000:29AC          loc_0_29AC:                                ; CODE XREF: sub_0_298C+14|j
0000:29AC 3E 01          ld      a, 1          ; sub_0_298C+1B|j
0000:29AE C9          ret
0000:29AE          ; End of function sub_0_298C
0000:29AE
0000:29AF          ; SUBROUTINE
0000:29AF
0000:29AF          sub_0_29AF:                                ; CODE XREF: sub_0_2B1C+7|p
0000:29AF 3E 04          ld      a, 4
0000:29B1 F7          rst      30h          ; return if level bit not set
0000:29B2 FD 21 00 62     ld      iy, mario_alive_flag
0000:29B6 3A 05 62     ld      a, (mario_x_coord)
0000:29B9 4F          ld      c, a
0000:29BA 21 08 04     ld      hl, 408h
0000:29BD CD 22 2A     call    sub_0_2A22
0000:29C0 A7          and     a
0000:29C1 CA 20 2A     jp     z, loc_0_2A20
0000:29C4 3E 06          ld      a, 6
0000:29C6 90          sub     b
0000:29C7          ;
0000:29C7          loc_0_29C7:                                ; CODE XREF: sub_0_29AF+1E|j
0000:29C7 CA D0 29     jp     z, loc_0_29D0
0000:29CA DD 19          add    ix, de
0000:29CC 3D          dec    a
0000:29CD C3 C7 29     jp     loc_0_29C7
0000:29D0          ;
0000:29D0          loc_0_29D0:                                ; CODE XREF: sub_0_29AF+18|j
0000:29D0 DD 7E 05     ld      a, (ix+5)
0000:29D3 D6 04          sub     4
0000:29D5 57          ld      d, a
0000:29D6 3A 0C 62     ld      a, (mario_y_before_jump)
0000:29D9 C6 05          add    a, 5
0000:29DB BA          cp     d
0000:29DC D2 EE 29     jp     nc, loc_0_29EE          ; check if on or below elevator
0000:29DF 7A          ld      a, d
0000:29E0 D6 08          sub     8
0000:29E2 32 05 62     ld      (mario_x_coord), a
0000:29E5 3E 01          ld      a, 1          ; flag on elevator
0000:29E7 47          ld      b, a
0000:29E8 32 98 63     ld      (mario_on_elevator), a
0000:29EB 33          inc    sp
0000:29EC 33          inc    sp

```

```

0000:29ED C9 ; _____ ret
0000:29EE
0000:29EE
0000:29EE
loc_0_29EE: ; CODE XREF: sub_0_29AF+2D↑j
; collide with side of elevator
0000:29EE 3A 0C 62 ld a, (mario_y_before_jump)
0000:29F1 D6 0E sub 0Eh
0000:29F3 BA cp d
0000:29F4 D2 1B 2A jp nc, loc_0_2A1B
0000:29F7 3A 10 62 ld a, (unk_0_6210)
0000:29FA A7 and a
0000:29FB 3A 03 62 ld a, (mario_y_coord)
0000:29FE CA 08 2A jp z, loc_0_2A08
0000:2A01 F6 07 or 7
0000:2A03 D6 04 sub 4
0000:2A05 C3 0E 2A jp loc_0_2A0E
0000:2A08
0000:2A08
0000:2A08
loc_0_2A08: ; CODE XREF: sub_0_29AF+4F↑j
0000:2A08 D6 08 sub 8
0000:2A0A F6 07 or 7
0000:2A0C C6 04 add a, 4
0000:2A0E
loc_0_2A0E: ; CODE XREF: sub_0_29AF+56↑j
0000:2A0E ld (mario_y_coord), a
0000:2A11 32 03 62 ld (soft_sprite_ram+4Ch), a
0000:2A14 3E 01 ld a, 1
0000:2A16 06 00 ld b, 0
0000:2A18 33 inc sp
0000:2A19 33 inc sp
0000:2A1A C9 ret
0000:2A1B
0000:2A1B
0000:2A1B
loc_0_2A1B: ; CODE XREF: sub_0_29AF+45↑j
0000:2A1B AF xor a
0000:2A1C 32 00 62 ld (mario_alive_flag), a
0000:2A1F C9 ret
0000:2A20
0000:2A20
0000:2A20
loc_0_2A20: ; CODE XREF: sub_0_29AF+12↑j
0000:2A20 47 ld b, a
0000:2A21 C9 ret
0000:2A21 ; End of function sub_0_29AF
0000:2A22
0000:2A22 ; ██████████ SUBROUTINE ██████████
0000:2A22
0000:2A22
sub_0_2A22: ; CODE XREF: sub_0_29AF+E↑p
0000:2A22 06 06 ld b, 6
0000:2A24 11 10 00 ld de, 10h
0000:2A27 DD 21 00 66 ld ix, unk_0_6600
0000:2A2B CD 13 29 call sub_0_2913
0000:2A2E C9 ret
0000:2A2E ; End of function sub_0_2A22
0000:2A2E
0000:2A2E
0000:2A2F
0000:2A2F ; ██████████ SUBROUTINE ██████████
0000:2A2F
0000:2A2F
sub_0_2A2F: ; CODE XREF: sub_0_1F72+E5↑p
; sub_0_1F72+188↑p
0000:2A2F DD 7E 03 ld a, (ix+3)
0000:2A32 67 ld h, a
0000:2A33 DD 7E 05 ld a, (ix+5)
0000:2A36 C6 04 add a, 4
0000:2A38 6F ld l, a
0000:2A39 E5 push hl
0000:2A3A CD F0 2F call sub_0_2FF0
0000:2A3D D1 pop de
0000:2A3E 7E ld a, (hl)
0000:2A3F FE B0 cp 0B0h ; '0'
0000:2A41 DA 7B 2A jp c, loc_0_2A7B
0000:2A44 E6 0F and 0Fh
0000:2A46 FE 08 cp 8
0000:2A48 D2 7B 2A jp nc, loc_0_2A7B
0000:2A4B 7E ld a, (hl)
0000:2A4C FE C0 cp 0C0h ; 'L'
0000:2A4E CA 7B 2A jp z, loc_0_2A7B
0000:2A51 DA 69 2A jp c, loc_0_2A69
0000:2A54 FE D0 cp 0D0h ; '0'
0000:2A56 DA 6E 2A jp c, loc_0_2A6E
0000:2A59 FE E0 cp 0E0h ; '0'
0000:2A5B DA 63 2A jp c, loc_0_2A63
0000:2A5E FE F0 cp 0F0h ; '-'
0000:2A60 DA 6E 2A jp c, loc_0_2A6E
0000:2A63
loc_0_2A63: ; CODE XREF: sub_0_2A2F+2C↑j
0000:2A63 E6 0F and 0Fh
0000:2A65 3D dec a
0000:2A66 C3 72 2A jp loc_0_2A72
0000:2A69
0000:2A69
loc_0_2A69: ; CODE XREF: sub_0_2A2F+22↑j
0000:2A69 3E FF ld a, 0FFh
0000:2A6B C3 72 2A jp loc_0_2A72
0000:2A6E
0000:2A6E
loc_0_2A6E: ; CODE XREF: sub_0_2A2F+27↑j
; sub_0_2A2F+31↑j
0000:2A6E E6 0F and 0Fh
0000:2A70 D6 09 sub 9
0000:2A72
loc_0_2A72: ; CODE XREF: sub_0_2A2F+37↑j
; sub_0_2A2F+3C↑j
0000:2A72 4F ld c, a
0000:2A73 7B ld a, e
0000:2A74 E6 F8 and 0F8h ; 'o'
0000:2A76 81 add a, c
0000:2A77 BB cp e
0000:2A78 DA 7D 2A jp c, loc_0_2A7D
0000:2A7B
loc_0_2A7B: ; CODE XREF: sub_0_2A2F+12↑j
; sub_0_2A2F+19↑j ...
0000:2A7B AF xor a
0000:2A7C C9 ret
0000:2A7D
; _____

```



```

0000:2A7D      loc_0_2A7D:                                ; CODE XREF: sub_0_2A2F+49↑j
0000:2A7D      sub      4
0000:2A7D D6 04      ld      (ix+5), a
0000:2A7F DD 77 05      ld      a, 1
0000:2A82 3E 01      ret
0000:2A84 C9          ; End of function sub_0_2A2F
0000:2A84
0000:2A84
0000:2A85      ; ██████████ SUBROUTINE ██████████
0000:2A85
0000:2A85      sub_0_2A85:                                ; CODE XREF: 0000:19A1↑p
0000:2A85 3A 15 62      ld      a, (mario_climbing)
0000:2A88 A7          and     a
0000:2A89 C0          ret     nz
0000:2A8A 3A 16 62      ld      a, (mario_jumping)
0000:2A8D A7          and     a
0000:2A8E C0          ret     nz
0000:2A8F 3A 98 63      ld      a, (mario_on_elevator)
0000:2A92 FE 01      cp      1
0000:2A94 C8          ret     z
0000:2A95 3A 03 62      ld      a, (mario_y_coord)
0000:2A98 D6 03      sub     3
0000:2A9A 67          ld      h, a
0000:2A9B 3A 05 62      ld      a, (mario_x_coord)
0000:2A9E C6 0C      add     a, 0Ch
0000:2AA0 6F          ld      l, a
0000:2AA1 E5          push    hl
0000:2AA2 CD F0 2F      call    sub_0_2FF0
0000:2AA5 D1          pop     de
0000:2AA6 7E          ld      a, (hl)
0000:2AA7 FE B0      cp      0B0h ; '0'
0000:2AA9 DA B4 2A      jp      c, loc_0_2AB4
0000:2AAC E6 0F      and     0Fh
0000:2AAE FE 08      cp      8
0000:2AB0 D2 B4 2A      jp      nc, loc_0_2AB4
0000:2AB3 C9          ret
0000:2AB4
0000:2AB4      loc_0_2AB4:                                ; CODE XREF: sub_0_2A85+24↑j
0000:2AB4      ; sub_0_2A85+2B↑j
0000:2AB4 7A          ld      a, d
0000:2AB5 E6 07      and     7
0000:2AB7 CA CD 2A      jp      z, loc_0_2ACD
0000:2ABA 01 20 00      ld      bc, 20h ; ' '
0000:2ABD ED 42      sbc     hl, bc
0000:2ABF 7E          ld      a, (hl)
0000:2AC0 FE B0      cp      0B0h ; '0'
0000:2AC2 DA CD 2A      jp      c, loc_0_2ACD
0000:2AC5 E6 0F      and     0Fh
0000:2AC7 FE 08      cp      8
0000:2AC9 D2 CD 2A      jp      nc, loc_0_2ACD
0000:2ACC C9          ret
0000:2ACD
0000:2ACD      loc_0_2ACD:                                ; CODE XREF: sub_0_2A85+32↑j
0000:2ACD      ; sub_0_2A85+3D↑j ...
0000:2ACD      ld      a, 1
0000:2ACF 32 21 62      ld      (unk_0_6221), a
0000:2AD2 C9          ret
0000:2AD2      ; End of function sub_0_2A85
0000:2AD2
0000:2AD3      ; ██████████ SUBROUTINE ██████████
0000:2AD3
0000:2AD3      sub_0_2AD3:                                ; CODE XREF: sub_0_25F2+C↑p
0000:2AD3 3A 03 62      ld      a, (mario_y_coord)
0000:2AD6 47          ld      b, a
0000:2AD7 3A 05 62      ld      a, (mario_x_coord)
0000:2ADA FE 50      cp      50h ; 'P'
0000:2ADC CA EA 2A      jp      z, loc_0_2AEA
0000:2ADF FE 78      cp      78h ; 'x'
0000:2AE1 CA F6 2A      jp      z, loc_0_2AF6
0000:2AE4 FE C8      cp      0C8h ; 'L'
0000:2AE6 CA F0 2A      jp      z, loc_0_2AF0
0000:2AE9 C9          ret
0000:2AEA
0000:2AEA      loc_0_2AEA:                                ; CODE XREF: sub_0_2AD3+9↑j
0000:2AEA 3A A3 63      ld      a, (unk_0_63A3)
0000:2AED C3 02 2B      jp      loc_0_2B02
0000:2AF0
0000:2AF0      loc_0_2AF0:                                ; CODE XREF: sub_0_2AD3+13↑j
0000:2AF0 3A A6 63      ld      a, (unk_0_63A6)
0000:2AF3 C3 02 2B      jp      loc_0_2B02
0000:2AF6
0000:2AF6      loc_0_2AF6:                                ; CODE XREF: sub_0_2AD3+E↑j
0000:2AF6 78          ld      a, b
0000:2AF7 FE 80      cp      80h ; 'Ç'
0000:2AF9 3A A5 63      ld      a, (unk_0_63A5)
0000:2AFC D2 02 2B      jp      nc, loc_0_2B02
0000:2AFF 3A A4 63      ld      a, (unk_0_63A4)
0000:2B02
0000:2B02      loc_0_2B02:                                ; CODE XREF: sub_0_2AD3+1A↑j
0000:2B02      ; sub_0_2AD3+20↑j ...
0000:2B02      add     a, b
0000:2B03 32 03 62      ld      (mario_y_coord), a
0000:2B06 32 4C 69      ld      (soft_sprite_ram+4Ch), a
0000:2B09 CD 1F 24      call    sub_0_241F
0000:2B0C 21 03 62      ld      hl, mario_y_coord
0000:2B0F 1D          dec     e
0000:2B10 CA 18 2B      jp      z, loc_0_2B18
0000:2B13 15          dec     d
0000:2B14 CA 1A 2B      jp      z, loc_0_2B1A
0000:2B17 C9          ret
0000:2B18
0000:2B18      loc_0_2B18:                                ; CODE XREF: sub_0_2AD3+3D↑j
0000:2B18 35          dec     (hl)
0000:2B19 C9          ret
0000:2B1A
0000:2B1A

```

```

0000:2B1A      loc_0_2B1A:      ; CODE XREF: sub_0_2AD3+41↑j
0000:2B1A 34      inc     (hl)
0000:2B1B C9      ret
0000:2B1B      ; End of function sub_0_2AD3
0000:2B1B
0000:2B1C      ; ██████████ SUBROUTINE ██████████
0000:2B1C
0000:2B1C      sub_0_2B1C:      ; CODE XREF: sub_0_1AC3+142↑p
0000:2B1C DD 21 00 62      ld     ix, mario_alive_flag
0000:2B20 CD 29 2B      call  sub_0_2B29
0000:2B23 CD AF 29      call  sub_0_29AF
0000:2B26 AF      xor     a
0000:2B27 47      ld     b, a
0000:2B28 C9      ret
0000:2B28      ; End of function sub_0_2B1C
0000:2B29
0000:2B29      ; ██████████ SUBROUTINE ██████████
0000:2B29
0000:2B29      sub_0_2B29:      ; CODE XREF: sub_0_2B1C+4↑p
0000:2B29 3A 27 62      ld     a, (level_type)
0000:2B2C 3D      dec     a
0000:2B2D C2 53 2B      jp     nz, loc_0_2B53
0000:2B30 3A 03 62      ld     a, (mario_y_coord)
0000:2B33 67      ld     h, a
0000:2B34 3A 05 62      ld     a, (mario_x_coord)
0000:2B37 C6 07      add     a, 7
0000:2B39 6F      ld     l, a
0000:2B3A CD 9B 2B      call  sub_0_2B9B
0000:2B3D A7      and     a
0000:2B3E CA 51 2B      jp     z, loc_0_2B51
0000:2B41 7B      ld     a, e
0000:2B42 91      sub     c
0000:2B43 FE 04      cp     4
0000:2B45 D2 74 2B      jp     nc, loc_0_2B74
0000:2B48 79      ld     a, c
0000:2B49 D6 07      sub     7
0000:2B4B 32 05 62      ld     (mario_x_coord), a
0000:2B4E 3E 01      ld     a, 1
0000:2B50 47      ld     b, a
0000:2B51
0000:2B51      loc_0_2B51:      ; CODE XREF: sub_0_2B29+15↑j
0000:2B51 E1      pop     hl
0000:2B52 C9      ret
0000:2B53
0000:2B53      ;
0000:2B53      loc_0_2B53:      ; CODE XREF: sub_0_2B29+4↑j
0000:2B53 3A 03 62      ld     a, (mario_y_coord)
0000:2B56 D6 03      sub     3
0000:2B58 67      ld     h, a
0000:2B59 3A 05 62      ld     a, (mario_x_coord)
0000:2B5C C6 07      add     a, 7
0000:2B5E 6F      ld     l, a
0000:2B5F CD 9B 2B      call  sub_0_2B9B
0000:2B62 FE 02      cp     2
0000:2B64 CA 7A 2B      jp     z, loc_0_2B7A
0000:2B67 7A      ld     a, d
0000:2B68 C6 07      add     a, 7
0000:2B6A 67      ld     h, a
0000:2B6B 6B      ld     l, e
0000:2B6C CD 9B 2B      call  sub_0_2B9B
0000:2B6F A7      and     a
0000:2B70 C8      ret     z
0000:2B71 C3 7A 2B      jp     loc_0_2B7A
0000:2B74
0000:2B74      ;
0000:2B74      loc_0_2B74:      ; CODE XREF: sub_0_2B29+1C↑j
0000:2B74 3E 00      ld     a, 0
0000:2B76 06 00      ld     b, 0
0000:2B78 E1      pop     hl
0000:2B79 C9      ret
0000:2B7A
0000:2B7A      ;
0000:2B7A      loc_0_2B7A:      ; CODE XREF: sub_0_2B29+3B↑j
0000:2B7A 3A 10 62      ; sub_0_2B29+48↑j
0000:2B7A      ld     a, (unk_0_6210)
0000:2B7D A7      and     a
0000:2B7E 3A 03 62      ld     a, (mario_y_coord)
0000:2B81 CA 8B 2B      jp     z, loc_0_2B8B
0000:2B84 F6 07      or     7
0000:2B86 D6 04      sub     4
0000:2B88 C3 91 2B      jp     loc_0_2B91
0000:2B8B
0000:2B8B      ;
0000:2B8B      loc_0_2B8B:      ; CODE XREF: sub_0_2B29+58↑j
0000:2B8B D6 08      sub     8
0000:2B8D F6 07      or     7
0000:2B8F C6 04      add     a, 4
0000:2B91
0000:2B91      loc_0_2B91:      ; CODE XREF: sub_0_2B29+5F↑j
0000:2B91 32 03 62      ld     (mario_y_coord), a
0000:2B94 32 4C 69      ld     (soft_sprite_ram+4Ch), a
0000:2B97 3E 01      ld     a, 1
0000:2B99 E1      pop     hl
0000:2B9A C9      ret
0000:2B9A      ; End of function sub_0_2B29
0000:2B9A
0000:2B9B      ; ██████████ SUBROUTINE ██████████
0000:2B9B
0000:2B9B      sub_0_2B9B:      ; CODE XREF: sub_0_2B29+11↑p
0000:2B9B E5      ; sub_0_2B29+36↑p ...
0000:2B9B      push    hl
0000:2B9C CD F0 2F      call  sub_0_2FF0
0000:2B9F D1      pop     de
0000:2BA0 7E      ld     a, (hl)
0000:2BA1 FE B0      cp     0B0h ; 0
0000:2BA3 DA D9 2B      jp     c, loc_0_2BD9
0000:2BA6 E6 0F      and     0Fh
0000:2BA8 FE 08      cp     8
0000:2BAA D2 D9 2B      jp     nc, loc_0_2BD9
0000:2BAD 7E      ld     a, (hl)

```

```

0000:2BAE FE C0      cp      0C0h ; 'L'
0000:2BB0 CA D9 2B      jp      z, loc_0_2BD9
0000:2BB3 DA DC 2B      jp      c, loc_0_2BDC
0000:2BB6 FE D0      cp      0D0h ; '3'
0000:2BB8 DA CB 2B      jp      c, loc_0_2BCB
0000:2BBB FE E0      cp      0E0h ; '0'
0000:2BBD DA C5 2B      jp      c, loc_0_2BC5
0000:2BC0 FE F0      cp      0F0h ; '-'
0000:2BC2 DA CB 2B      jp      c, loc_0_2BCB
0000:2BC5
0000:2BC5      loc_0_2BC5:                                ; CODE XREF: sub_0_2B9B+22↑j
0000:2BC5 E6 0F      and     0Fh
0000:2BC7 3D      dec     a
0000:2BC8 C3 CF 2B      jp      loc_0_2BCF
0000:2BCB
0000:2CB      loc_0_2CB:                                ; CODE XREF: sub_0_2B9B+1D↑j
0000:2CB E6 0F      and     0Fh
0000:2CB      sub     9
0000:2CD D6 09      sub     9
0000:2CF      loc_0_2CF:                                ; CODE XREF: sub_0_2B9B+2D↑j
0000:2CF 4F      ld      c, a
0000:2D0 7B      ld      a, e
0000:2D1 E6 F8      and     0F8h ; 'o'
0000:2D3 81      add     a, c
0000:2D4 4F      ld      c, a
0000:2D5 BB      cp      e
0000:2D6 DA E1 2B      jp      c, loc_0_2BE1
0000:2D9
0000:2D9      loc_0_2D9:                                ; CODE XREF: sub_0_2B9B+8↑j
0000:2D9 AF      xor     a
0000:2D9      ld      b, a
0000:2DA 47      ret
0000:2DB C9
0000:2DDC
0000:2DDC      loc_0_2DDC:                                ; CODE XREF: sub_0_2B9B+18↑j
0000:2DDC 7B      ld      a, e
0000:2DD E6 F8      and     0F8h ; 'o'
0000:2DDF 3D      dec     a
0000:2DE0 4F      ld      c, a
0000:2DE1
0000:2DE1      loc_0_2DE1:                                ; CODE XREF: sub_0_2B9B+3B↑j
0000:2DE1 3A 0C 62      ld      a, (mario_y_before_jump)
0000:2DE4 DD 96 05      sub     (ix+5)
0000:2DE7 83      add     a, e
0000:2DE8 B9      cp      c
0000:2DE9 CA EF 2B      jp      z, loc_0_2BEF
0000:2DEC D2 F8 2B      jp      nc, loc_0_2BF8
0000:2BEF
0000:2BEF      loc_0_2BEF:                                ; CODE XREF: sub_0_2B9B+4E↑j
0000:2BEF 79      ld      a, c
0000:2BF0 D6 07      sub     7
0000:2BF2 32 05 62      ld      (mario_x_coord), a
0000:2BF5 C3 FD 2B      jp      loc_0_2BFD
0000:2BF8
0000:2BF8      loc_0_2BF8:                                ; CODE XREF: sub_0_2B9B+51↑j
0000:2BF8 3E 02      ld      a, 2
0000:2BFA 06 00      ld      b, 0
0000:2BFC C9      ret
0000:2BFD
0000:2BFD      loc_0_2BFD:                                ; CODE XREF: sub_0_2B9B+5A↑j
0000:2BFD 3E 01      ld      a, 1
0000:2BFF 47      ld      b, a
0000:2C00 E1      pop     hl
0000:2C01 E1      pop     hl
0000:2C02 C9      ret
0000:2C02      ; End of function sub_0_2B9B
0000:2C02
0000:2C03
0000:2C03      ; ██████████ S U B R O U T I N E ██████████
0000:2C03
0000:2C03      sub_0_2C03:                                ; CODE XREF: 0000:1989↑p
0000:2C03 3E 01      ld      a, 1
0000:2C05 F7      rst     30h
0000:2C06 D7      rst     10h
0000:2C07 3A 93 63      ld      a, (barrel_deployment)
0000:2C0A 0F      rrca
0000:2C0B D8      ret     c
0000:2C0C 3A B1 62      ld      a, (unk_0_62B1)
0000:2C0F A7      and     a
0000:2C10 C8      ret     z
0000:2C11 4F      ld      c, a
0000:2C12 3A B0 62      ld      a, (unk_0_62B0)
0000:2C15 D6 02      sub     2
0000:2C17 B9      cp      c
0000:2C18 DA 7B 2C      jp      c, loc_0_2C7B
0000:2C1B 3A 82 63      ld      a, (unk_0_6382)
0000:2C1E CB 4F      bit     1, a
0000:2C20 C2 86 2C      jp      nz, loc_0_2C86
0000:2C23 3A 80 63      ld      a, (unk_0_6380)
0000:2C26 47      ld      b, a
0000:2C27 3A 1A 60      ld      a, (gen_purpose_timer)
0000:2C2A E6 1F      and     1Fh
0000:2C2C
0000:2C2C      loc_0_2C2C:                                ; CODE XREF: sub_0_2C03+2D↑j
0000:2C2C B8      cp      b
0000:2C2D CA 33 2C      jp      z, loc_0_2C33
0000:2C30 10 FA      djnz   loc_0_2C2C
0000:2C32 C9      ret
0000:2C33
0000:2C33      loc_0_2C33:                                ; CODE XREF: sub_0_2C03+2A↑j
0000:2C33 3A B0 62      ld      a, (unk_0_62B0)
0000:2C36 CB 3F      srl     a
0000:2C38 B9      cp      c
0000:2C39 DA 41 2C      jp      c, loc_0_2C41
0000:2C3C 3A 19 60      ld      a, (random_no+1)
0000:2C3F 0F      rrca
0000:2C40 D0      ret     nc
0000:2C41
0000:2C41      loc_0_2C41:                                ; CODE XREF: sub_0_2C03+36↑j

```

```

0000:2C41 CD 57 00      call    rand
0000:2C44 E6 0F          and     0Fh
0000:2C46 C2 86 2C          jp      nz, loc_0_2C86
0000:2C49
0000:2C49      loc_0_2C49:      ld      a, 1                      ; CODE XREF: sub_0_2C03+7B|j
0000:2C49 3E 01
0000:2C4B      loc_0_2C4B:      ld      (unk_0_6382), a          ; CODE XREF: sub_0_2C03+80|j
0000:2C4B 32 82 63
0000:2C4E 3C          inc     a
0000:2C4F
0000:2C4F      loc_0_2C4F:      ld      (unk_0_638F), a          ; CODE XREF: sub_0_2C03+89|j
0000:2C4F 32 8F 63
0000:2C52 3E 01      ld      a, 1
0000:2C54 32 92 63      ld      (unk_0_6392), a
0000:2C57 3A B2 62      ld      a, (unk_0_62B2)
0000:2C5A B9          cp      c
0000:2C5B C0          ret     nz
0000:2C5C D6 08      sub     8
0000:2C5E 32 B2 62      ld      (unk_0_62B2), a
0000:2C61 11 20 00      ld      de, 20h ; ' '
0000:2C64 21 00 64      ld      hl, unk_0_6400          ; fireball character data
0000:2C67 06 05      ld      b, 5
0000:2C69
0000:2C69      loc_0_2C69:      ld      a, (hl)                  ; CODE XREF: sub_0_2C03+6C|j
0000:2C69 7E          and     a
0000:2C6A A7          jp      z, loc_0_2C72
0000:2C6E 19          add     hl, de
0000:2C6F 10 F8      djnz   loc_0_2C69
0000:2C71 C9          ret
0000:2C72
0000:2C72      loc_0_2C72:      ld      a, (unk_0_6382)          ; CODE XREF: sub_0_2C03+68|j
0000:2C72 3A 82 63      or      80h ; 'C'
0000:2C75 F6 80      ld      (unk_0_6382), a
0000:2C77 32 82 63      ret
0000:2C7A C9
0000:2C7B
0000:2C7B      loc_0_2C7B:      add     a, 2                      ; CODE XREF: sub_0_2C03+15|j
0000:2C7B C6 02      cp      c
0000:2C7D B9          jp      z, loc_0_2C49
0000:2C7E CA 49 2C      ld      a, 2
0000:2C81 3E 02      jp      loc_0_2C4B
0000:2C83 C3 4B 2C
0000:2C86
0000:2C86      loc_0_2C86:      xor     a                        ; CODE XREF: sub_0_2C03+1D|j
0000:2C86 AF          ld      (unk_0_6382), a          ; sub_0_2C03+43|j
0000:2C87 32 82 63      ld      a, 3
0000:2C8A 3E 03      jp      loc_0_2C4F
0000:2C8C C3 4F 2C      ; End of function sub_0_2C03
0000:2C8C
0000:2C8F
0000:2C8F      ; SUBROUTINE
0000:2C8F
0000:2C8F      sub_0_2C8F:      ld      a, 1                      ; CODE XREF: 0000:1986|p
0000:2C8F 3E 01      rst     30h                      ; return if level bit not set
0000:2C91 F7          rst     10h                      ; return if mario not alive
0000:2C92 D7          ld      a, (barrel_deployment)
0000:2C93 3A 93 63      rrca
0000:2C96 0F          jp      c, loc_0_2D15
0000:2C9A 3A 92 63      ld      a, (unk_0_6392)
0000:2C9D 0F          rrca
0000:2C9E D0          ret     nc
0000:2C9F DD 21 00 67      ld      ix, unk_0_6700
0000:2CA3 11 20 00      ld      de, 20h ; ' '
0000:2CA6 06 0A      ld      b, 0Ah
0000:2CA8
0000:2CA8      loc_0_2CA8:      ld      a, (ix+0)                  ; CODE XREF: sub_0_2C8F+26|j
0000:2CA8 DD 7E 00      rrca
0000:2CAB 0F          jp      c, loc_0_2CB3
0000:2CAC DA B3 2C      rrca
0000:2CAF 0F          jp      nc, loc_0_2CB8
0000:2CB0 D2 B8 2C
0000:2CB3      loc_0_2CB3:      add     ix, de                      ; CODE XREF: sub_0_2C8F+1D|j
0000:2CB3 DD 19      djnz   loc_0_2CA8
0000:2CB5 10 F1      ret
0000:2CB7 C9
0000:2CB8
0000:2CB8      loc_0_2CB8:      ld      (unk_0_62AA), ix          ; CODE XREF: sub_0_2C8F+21|j
0000:2CB8 DD 22 AA 62      ld      (ix+0), 2
0000:2CBC DD 36 00 02      ld      d, 0
0000:2CC0 16 00      ld      a, 0Ah
0000:2CC2 3E 0A      sub     b
0000:2CC4 90          add     a, a
0000:2CC5 87          add     a, a
0000:2CC6 87          ld      e, a
0000:2CC7 5F          ld      hl, soft_sprite_ram+80h
0000:2CC8 21 80 69      add     hl, de
0000:2CCB 19          ld      (unk_0_62AC), hl
0000:2CCC 22 AC 62      ld      a, 1
0000:2CCF 3E 01      ld      (barrel_deployment), a
0000:2CD1 32 93 63      ld      de, 501h
0000:2CD4 11 01 05      call   sub_0_309F
0000:2CD7 CD 9F 30      ld      hl, unk_0_62B1
0000:2CDA 21 B1 62      dec     (hl)
0000:2CDD 35          jp      nz, loc_0_2CE6
0000:2CDE C2 E6 2C      ld      a, 1
0000:2CE1 3E 01      ld      (unk_0_6386), a
0000:2CE3 32 86 63
0000:2CE6      loc_0_2CE6:      ld      a, (hl)                  ; CODE XREF: sub_0_2C8F+4F|j
0000:2CE6 7E          cp      4
0000:2CE7 FE 04      jp      nc, loc_0_2CF6
0000:2CE9 D2 F6 2C      ld      hl, soft_sprite_ram+0A8h
0000:2CEC 21 A8 69      add     a, a
0000:2CEF 87          add     a, a
0000:2CF0 87

```

```

0000:2CF1 5F          ld      e, a
0000:2CF2 16 00          ld      d, 0
0000:2CF4 19          add     hl, de
0000:2CF5 72          ld      (hl), d
0000:2CF6
0000:2CF6          loc_0_2CF6:
0000:2CF6 DD 36 07 15      ld      (ix+7), 15h          ; CODE XREF: sub_0_2C8F+5A↑j
0000:2CFA DD 36 08 0B      ld      (ix+8), 0Bh          ; sideways barrel sprite tile
0000:2CFE DD 36 15 00      ld      (ix+15h), 0
0000:2D02 3A 82 63      ld      a, (unk_0_6382)
0000:2D05 07          rlca
0000:2D06 D2 15 2D      jp      nc, loc_0_2D15
0000:2D09 DD 36 07 19      ld      (ix+7), 19h          ; sideways blue barrel sprite tile
0000:2D0D DD 36 08 0C      ld      (ix+8), 0Ch          ; set blue palette for barrel
0000:2D11 DD 36 15 01      ld      (ix+15h), 1
0000:2D15
0000:2D15          loc_0_2D15:
0000:2D15 21 AF 62          ld      hl, unk_0_62AF          ; CODE XREF: sub_0_2C8F+8↑j
0000:2D15          ; sub_0_2C8F+77↑j
0000:2D18 35          dec     (hl)
0000:2D19 C0          ret     nz
0000:2D1A 36 18          ld      (hl), 18h
0000:2D1C 3A 8F 63      ld      a, (unk_0_638F)
0000:2D1F A7          and     a
0000:2D20 CA 51 2D      jp      z, loc_0_2D51
0000:2D23 4F          ld      c, a
0000:2D24 21 32 39      ld      hl, dk_throw_barrel_spr
0000:2D27 3A 82 63      ld      a, (unk_0_6382)
0000:2D2A 0F          rrca
0000:2D2B DA 2F 2D      jp      c, loc_0_2D2F
0000:2D2E 0D          dec     c
0000:2D2F
0000:2D2F          loc_0_2D2F:
0000:2D2F 79          ld      a, c          ; CODE XREF: sub_0_2C8F+9C↑j
0000:2D30 87          add     a, a
0000:2D31 87          add     a, a
0000:2D32 87          add     a, a
0000:2D33 4F          ld      c, a
0000:2D34 87          add     a, a
0000:2D35 87          add     a, a
0000:2D36 81          add     a, c
0000:2D37 5F          ld      e, a
0000:2D38 16 00      ld      d, 0
0000:2D3A 19          add     hl, de
0000:2D3B CD 4E 00      call    copy_sprites_2_11_data
0000:2D3E 21 8F 63      ld      hl, unk_0_638F
0000:2D41 35          dec     (hl)
0000:2D42 C2 51 2D      jp      nz, loc_0_2D51
0000:2D45 3E 01      ld      a, 1
0000:2D47 32 AF 62      ld      (unk_0_62AF), a
0000:2D4A 3A 82 63      ld      a, (unk_0_6382)
0000:2D4D 0F          rrca
0000:2D4E DA 83 2D      jp      c, loc_0_2D83
0000:2D51
0000:2D51          loc_0_2D51:
0000:2D51 2A A8 62          ld      hl, (unk_0_62A8)          ; CODE XREF: sub_0_2C8F+91↑j
0000:2D51          ; sub_0_2C8F+B3↑j
0000:2D54
0000:2D54          loc_0_2D54:
0000:2D54 7E          ld      a, (hl)          ; CODE XREF: sub_0_2C8F+FA↑j
0000:2D55 DD 2A AA 62      ld      ix, (unk_0_62AA)
0000:2D59 ED 5B AC 62      ld      de, (unk_0_62AC)
0000:2D5D FE 7F          cp      7Fh ; ' '
0000:2D5F CA 8C 2D      jp      z, loc_0_2D8C
0000:2D62 4F          ld      c, a
0000:2D63 E6 7F          and     7Fh ; ' '
0000:2D65 12          ld      (de), a          ; sprite data X coord
0000:2D66 DD 7E 07      ld      a, (ix+7)          ; sprite tile #
0000:2D69 CB 79          bit     7, c
0000:2D6B CA 70 2D      jp      z, loc_0_2D70
0000:2D6E EE 03          xor     3
0000:2D70
0000:2D70          loc_0_2D70:
0000:2D70 13          inc     de          ; CODE XREF: sub_0_2C8F+DC↑j
0000:2D71 12          ld      (de), a          ; sprite tile # (barrel)
0000:2D72 DD 77 07      ld      (ix+7), a          ; sprite tile #
0000:2D75 DD 7E 08      ld      a, (ix+8)
0000:2D78 13          inc     de
0000:2D79 12          ld      (de), a
0000:2D7A 23          inc     hl
0000:2D7B 7E          ld      a, (hl)
0000:2D7C 13          inc     de
0000:2D7D 12          ld      (de), a
0000:2D7E 23          inc     hl
0000:2D7F 22 A8 62      ld      (unk_0_62A8), hl
0000:2D82 C9          ret
0000:2D83
0000:2D83          loc_0_2D83:
0000:2D83          ; CODE XREF: sub_0_2C8F+BF↑j
0000:2D83 21 CC 39      ld      hl, barrel_falling_data
0000:2D86 22 A8 62      ld      (unk_0_62A8), hl
0000:2D89 C3 54 2D      jp      loc_0_2D54
0000:2D8C
0000:2D8C          loc_0_2D8C:
0000:2D8C          ; CODE XREF: sub_0_2C8F+D0↑j
0000:2D8C 21 C3 39      ld      hl, barrel_rolling_data
0000:2D8F 22 A8 62      ld      (unk_0_62A8), hl
0000:2D92 DD 36 01 01      ld      (ix+1), 1
0000:2D96 3A 82 63      ld      a, (unk_0_6382)
0000:2D99 0F          rrca
0000:2D9A DA A5 2D      jp      c, loc_0_2DA5
0000:2D9D DD 36 01 00      ld      (ix+1), 0
0000:2DA1 DD 36 02 02      ld      (ix+2), 2
0000:2DA5
0000:2DA5          loc_0_2DA5:
0000:2DA5          ; CODE XREF: sub_0_2C8F+10B↑j
0000:2DA5 DD 36 00 01      ld      (ix+0), 1
0000:2DA9 DD 36 0F 01      ld      (ix+0Fh), 1
0000:2DAD AF          xor     a
0000:2DAE DD 77 10      ld      (ix+10h), a
0000:2DB1 DD 77 11      ld      (ix+11h), a
0000:2DB4 DD 77 12      ld      (ix+12h), a
0000:2DB7 DD 77 13      ld      (ix+13h), a
0000:2DBA DD 77 14      ld      (ix+14h), a
0000:2DBD 32 93 63      ld      (barrel_deployment), a
0000:2DC0 32 92 63      ld      (unk_0_6392), a
0000:2DC3 1A          ld      a, (de)

```



```

0000:2E7F FD 19          add    iy, de                ; next sprite data
0000:2E81 10 8F          djnz   loc_0_2E12
0000:2E83 C9                  ret
0000:2E84                ;
0000:2E84                ;
0000:2E84                ;
0000:2E84 3E 03          loc_0_2E84: ld    a, 3                ; CODE XREF: sub_0_2E04+2A|j
0000:2E86 DD 86 05          add    a, (ix+5)
0000:2E89 DD 77 05          ld     (ix+5), a
0000:2E8C FE F8          cp     0F8h ; 'o'
0000:2E8E DA 6C 2E          jp     c, loc_0_2E6C
0000:2E91 DD 36 03 00          ld     (ix+3), 0
0000:2E95 DD 36 00 00          ld     (ix+0), 0
0000:2E99 C3 6C 2E          jp     loc_0_2E6C
0000:2E9C                ;
0000:2E9C                ;
0000:2E9C 21 AA 39          loc_0_2E9C: ld     hl, bouncing_spring_data ; CODE XREF: sub_0_2E04+3D|j
0000:2E9F 3E 03          ld     a, 3                ; tmr=3
0000:2EA1 32 83 60          ld     (digital_snd_tmr_coin_spring), a
0000:2EA4 C3 4B 2E          jp     loc_0_2E4B
0000:2EA7                ;
0000:2EA7                ;
0000:2EA7 3A 96 63          loc_0_2EA7: ld     a, (unk_0_6396)                ; CODE XREF: sub_0_2E04+12|j
0000:2EAA 0F          rrca
0000:2EAB D2 78 2E          jp     nc, loc_0_2E78
0000:2EAE AF          xor    a
0000:2EAF 32 96 63          ld     (unk_0_6396), a
0000:2EB2 DD 36 05 50          ld     (ix+5), 50h ; 'P'
0000:2EB6 DD 36 0D 01          ld     (ix+0Dh), 1
0000:2EBA CD 57 00          call   rand
0000:2EBD E6 0F          and    0Fh
0000:2EBF C6 F8          add    a, 0F8h ; 'o'
0000:2EC1 DD 77 03          ld     (ix+3), a
0000:2EC4 DD 36 00 01          ld     (ix+0), 1
0000:2EC8 21 AA 39          ld     hl, bouncing_spring_data
0000:2ECB DD 75 0E          ld     (ix+0Eh), 1
0000:2ECE DD 74 0F          ld     (ix+0Fh), h
0000:2ED1 C3 78 2E          jp     loc_0_2E78                ; end of spring routine
0000:2ED1                ; End of function sub_0_2E04
0000:2ED1                ;
0000:2ED4                ; SUBROUTINE
0000:2ED4                ;
0000:2ED4                ;
0000:2ED4 3E 0B          sub_0_2ED4: ld     a, 0Bh                ; CODE XREF: 0000:1998|p
0000:2ED6 F7          rst     30h                ; return if level bit not set
0000:2ED7 D7          rst     10h                ; return if mario not alive
0000:2ED8 11 18 6A          ld     de, soft_sprite_ram+118h ; hammers in sprite ram
0000:2EDB DD 21 80 66          ld     ix, unk_0_6680        ; hammer character data
0000:2EDF DD 7B 01          ld     a, (ix+1)
0000:2EE2 0F          rrca
0000:2EE3 DA ED 2E          jp     c, loc_0_2EED
0000:2EE6 11 1C 6A          ld     de, soft_sprite_ram+11Ch
0000:2EE9 DD 21 90 66          ld     ix, unk_0_6690
0000:2EED                ;
0000:2EED 2EED DD 36 0E 00          loc_0_2EED: ld     (ix+0Eh), 0                ; CODE XREF: sub_0_2ED4+F|j
0000:2EF1 DD 36 0F F0          ld     (ix+0Fh), 0F0h ; '-'
0000:2EF5 3A 17 62          ld     a, (unk_0_6217)
0000:2EF8 0F          rrca
0000:2EF9 D2 97 2F          jp     nc, loc_0_2F97
0000:2EFC AF          xor    a
0000:2EFD 32 18 62          ld     (unk_0_6218), a
0000:2F00 21 89 60          ld     hl, unk_0_6089
0000:2F03 36 04          ld     (hl), 4
0000:2F05 DD 36 09 06          ld     (ix+9), 6
0000:2F09 DD 36 0A 03          ld     (ix+0Ah), 3
0000:2F0D 06 1E          ld     b, 1Eh
0000:2F0F 3A 07 62          ld     a, (mario_flipy_sprite_tile)
0000:2F12 CB 27          sla    a
0000:2F14 D2 1B 2F          jp     nc, loc_0_2F1B
0000:2F17 F6 80          or     80h ; 'Ç'
0000:2F19 CB F8          set    7, b
0000:2F1B                ;
0000:2F1B 2F1B DD 36 0E 10          loc_0_2F1B: ld     (ix+0Eh), 10h                ; CODE XREF: sub_0_2ED4+40|j
0000:2F1B F6 08          or     8
0000:2F1D 4F          ld     c, a
0000:2F1E 3A 94 63          ld     a, (unk_0_6394)
0000:2F21 CB 5F          bit    3, a
0000:2F23 CA 43 2F          jp     z, loc_0_2F43
0000:2F26 CB C0          set    0, b
0000:2F28 CB C1          set    0, c
0000:2F2A DD 36 09 05          ld     (ix+9), 5
0000:2F2E DD 36 0A 06          ld     (ix+0Ah), 6
0000:2F32 DD 36 0F 00          ld     (ix+0Fh), 0
0000:2F36 DD 36 0E F0          ld     (ix+0Eh), 0F0h ; '-'
0000:2F3A CB 79          bit    7, c
0000:2F3C CA 43 2F          jp     z, loc_0_2F43
0000:2F3F DD 36 0E 10          ld     (ix+0Eh), 10h
0000:2F43                ;
0000:2F43 2F43 DD 36 0E 10          loc_0_2F43: ld     (ix+0Eh), 10h                ; CODE XREF: sub_0_2ED4+4F|j
0000:2F43 79                ; sub_0_2ED4+68|j
0000:2F43                ;
0000:2F43 32 4D 69          ld     a, c
0000:2F47 0E 07          ld     (soft_sprite_ram+4Dh), a
0000:2F49 21 94 63          ld     hl, unk_0_6394
0000:2F4C 34          inc    (hl)
0000:2F4D C2 B7 2F          jp     nz, loc_0_2FB7
0000:2F50 21 95 63          ld     hl, unk_0_6395
0000:2F53 34          inc    (hl)
0000:2F54 7E          ld     a, (hl)
0000:2F55 FE 02          cp     2
0000:2F57 C2 BE 2F          jp     nz, loc_0_2FBE
0000:2F5A AF          xor    a
0000:2F5B 32 95 63          ld     (unk_0_6395), a
0000:2F5E 32 17 62          ld     (unk_0_6217), a
0000:2F61 DD 77 01          ld     (ix+1), a
0000:2F64 3A 03 62          ld     a, (mario_y_coord)
0000:2F67 ED 44          neg
0000:2F69 DD 77 0E          ld     (ix+0Eh), a
0000:2F6C 3A 07 62          ld     a, (mario_flipy_sprite_tile)
0000:2F6F 32 4D 69          ld     (soft_sprite_ram+4Dh), a
0000:2F72 DD 36 00 00          ld     (ix+0), 0

```

```

0000:2F76 3A 89 63      ld      a, (unk_0_6389)
0000:2F79 32 89 60      ld      (unk_0_6089), a
0000:2F7C
0000:2F7C      loc_0_2F7C:
0000:2F7C EB          ; CODE XREF: sub_0_2ED4+E0↑j
0000:2F7C          ; sub_0_2ED4+E7↑j ...
0000:2F7D 3A 03 62      ex      de, hl
0000:2F80 DD 86 0E      ld      a, (mario_y_coord)
0000:2F83 77          add     a, (ix+0Eh)
0000:2F84 DD 77 03      ld      (hl), a
0000:2F87 23          ld      (ix+3), a
0000:2F88 70          inc     hl
0000:2F89 23          ld      (hl), b
0000:2F8A 71          inc     hl
0000:2F8B 23          ld      (hl), c
0000:2F8C 3A 05 62      ld      a, (mario_x_coord)
0000:2F8F DD 86 0F      add     a, (ix+0Fh)
0000:2F92 77          ld      (hl), a
0000:2F93 DD 77 05      ld      (ix+5), a
0000:2F96 C9          ret
0000:2F97
0000:2F97      ;
0000:2F97      loc_0_2F97:
0000:2F97          ; CODE XREF: sub_0_2ED4+25↑j
0000:2F97 3A 18 62      ld      a, (unk_0_6218)
0000:2F9A 0F          rrca
0000:2F9B D0          ret
0000:2F9C DD 36 09 06      nc
0000:2FA0 DD 36 0A 03      ld      (ix+9), 6
0000:2FA4 3A 07 62      ld      (ix+0Ah), 3
0000:2FA7 07          ld      a, (mario_flip_sprite_tile)
0000:2FA8 3E 3C      rlca
0000:2FAA 1F      ld      a, 3Ch ; '<'
0000:2FAB 47          rra
0000:2FAC 0E 07      ld      b, a
0000:2FAE 3A 89 60      ld      c, 7
0000:2FB1 32 89 63      ld      a, (unk_0_6089)
0000:2FB4 C3 7C 2F      ld      (unk_0_6389), a
0000:2FB7          jp      loc_0_2F7C
0000:2FB7      ;
0000:2FB7      loc_0_2FB7:
0000:2FB7          ; CODE XREF: sub_0_2ED4+79↑j
0000:2FB7 3A 95 63      ld      a, (unk_0_6395)
0000:2FBA A7          and     a
0000:2FBB CA 7C 2F      jp      z, loc_0_2F7C
0000:2FBE          ;
0000:2FBE      loc_0_2FBE:
0000:2FBE          ; CODE XREF: sub_0_2ED4+83↑j
0000:2FBE 3A 1A 60      ld      a, (gen_purpose_timer)
0000:2FC1 CB 5F      bit     3, a
0000:2FC3 CA 7C 2F      jp      z, loc_0_2F7C
0000:2FC6 0E 01      ld      c, 1
0000:2FC8 C3 7C 2F      jp      loc_0_2F7C
0000:2FC8      ; End of function sub_0_2ED4
0000:2FC8
0000:2FCB
0000:2FCB      ; SUBROUTINE
0000:2FCB
0000:2FCB      sub_0_2FCB:
0000:2FCB          ; CODE XREF: 0000:19BF↑p
0000:2FCB 3E 0E      ld      a, 0Eh
0000:2FCD F7          rst     30h
0000:2FCE 21 B4 62      ld      hl, unk_0_62B4
0000:2FD1 35          dec     (hl)
0000:2FD2 C0          ret     nz
0000:2FD3 3E 03      ld      a, 3
0000:2FD5 32 B9 62      ld      (unk_0_62B9), a
0000:2FD8 32 96 63      ld      (unk_0_6396), a
0000:2FDB 11 01 05      ld      de, 501h
0000:2FDE CD 9F 30      call    sub_0_309F
0000:2FE1 3A B3 62      ld      a, (unk_0_62B3)
0000:2FE4 77          ld      (hl), a
0000:2FE5 21 B1 62      ld      hl, unk_0_62B1
0000:2FE8 35          dec     (hl)
0000:2FE9 C0          ret
0000:2FEA 3E 01      ld      nz, a, 1
0000:2FEC 32 86 63      ld      a, 1
0000:2FEF C9          ld      (unk_0_6386), a
0000:2FEF      ret
0000:2FEF      ; End of function sub_0_2FCB
0000:2FEF
0000:2FF0
0000:2FF0      ; SUBROUTINE
0000:2FF0
0000:2FF0      sub_0_2FF0:
0000:2FF0          ; CODE XREF: sub_0_DA7+10↑p
0000:2FF0          ; sub_0_DA7+3D↑p ...
0000:2FF0          ; byte2
0000:2FF0 7D          ld      a, 1
0000:2FF1 0F          rrca
0000:2FF2 0F          rrca
0000:2FF3 0F          rrca
0000:2FF4 E6 1F      and     31
0000:2FF6 6F      ld      l, a
0000:2FF7 7C          ld      a, h
0000:2FF8 2F      cpl
0000:2FF9 E6 F8      and     0F8h ; '0'
0000:2FFB 5F      ld      e, a
0000:2FFC AF      xor     a
0000:2FFD 67      ld      h, a
0000:2FFE CB 13      rl      e
0000:3000 17          rla
0000:3001 CB 13      rl      e
0000:3003 17          rla
0000:3004 C6 74      add     a, 74h ; 't'
0000:3006 57          ld      d, a
0000:3007 19      add     hl, de
0000:3008 C9          ret
0000:3008      ; End of function sub_0_2FF0
0000:3008
0000:3009
0000:3009      ; SUBROUTINE
0000:3009
0000:3009      sub_0_3009:
0000:3009          ; CODE XREF: 0000:18DF↑p
0000:3009          ; sub_0_1AC3+1DB↑p ...
0000:3009 57          ld      d, a
0000:3009          rrca
0000:300A 0F          jp      c, loc_0_3022
0000:300B DA 22 30      ld      c, 93h ; '6'
0000:300E 0E 93      rrca
0000:3010 0F

```



```

0000:3011 0F          rrca
0000:3012 D2 17 30      jp      nc, loc_0_3017
0000:3015 0E 6C          ld      c, 6Ch ; 'l'
0000:3017
0000:3017          loc_0_3017:                                ; CODE XREF: sub_0_3009+9↑j
0000:3017 07          rlca
0000:3018 DA 31 30      jp      c, loc_0_3031
0000:301B 79          ld      a, c
0000:301C E6 F0          and     0F0h ; '-'
0000:301E 4F          ld      c, a
0000:301F C3 31 30      jp      loc_0_3031
0000:3022          ;
0000:3022          loc_0_3022:                                ; CODE XREF: sub_0_3009+2↑j
0000:3022 0E B4          ld      c, 0B4h ; 'l'
0000:3024 0F          rrca
0000:3025 0F          rrca
0000:3026 D2 2B 30      jp      nc, loc_0_302B
0000:3029 0E 1E          ld      c, 1Eh
0000:302B
0000:302B          loc_0_302B:                                ; CODE XREF: sub_0_3009+1D↑j
0000:302B CB 50          bit     2, b
0000:302D CA 31 30      jp      z, loc_0_3031
0000:3030 05          dec     b
0000:3031
0000:3031          loc_0_3031:                                ; CODE XREF: sub_0_3009+F↑j
0000:3031 79          ; sub_0_3009+16↑j ...
0000:3031          ld      a, c
0000:3032 0F          rrca
0000:3033 0F          rrca
0000:3034 4F          ld      c, a
0000:3035 E6 03          and     3
0000:3037 B8          cp      b
0000:3038 C2 31 30      jp      nz, loc_0_3031
0000:303B 79          ld      a, c
0000:303C 0F          rrca
0000:303D 0F          rrca
0000:303E E6 03          and     3
0000:3040 FE 03          cp      3
0000:3042 C0          ret     nz
0000:3043 CB 92          res     2, d
0000:3045 15          dec     d
0000:3046 C0          ret     nz
0000:3047 3E 04          ld      a, 4
0000:3049 C9          ret
0000:3049          ; End of function sub_0_3009
0000:3049
0000:304A          ; SUBROUTINE
0000:304A
0000:304A          sub_0_304A:                                ; CODE XREF: 0000:0AF0↑p
0000:304A 11 E0 FF          ; 0000:0B38↑p
0000:304A          ld      de, 0FFE0h
0000:304D 3A 8E 63      ld      a, (unk_0_638E)
0000:3050 4F          ld      c, a
0000:3051 06 00          ld      b, 0
0000:3053 21 00 76      ld      hl, VRAM_start+200h
0000:3056 CD 64 30      call    sub_0_3064
0000:3059 21 C0 75      ld      hl, VRAM_start+1C0h
0000:305C CD 64 30      call    sub_0_3064
0000:305F 21 8E 63      ld      hl, unk_0_638E
0000:3062 35          dec     (hl)
0000:3063 C9          ret
0000:3063          ; End of function sub_0_304A
0000:3063
0000:3064          ; SUBROUTINE
0000:3064
0000:3064          sub_0_3064:                                ; CODE XREF: sub_0_304A+C↑p
0000:3064 09          ; sub_0_304A+12↑p
0000:3064          add     hl, bc
0000:3065 7E          ld      a, (hl)
0000:3066 19          add     hl, de
0000:3067 77          ld      (hl), a
0000:3068 C9          ret
0000:3068          ; End of function sub_0_3064
0000:3068
0000:3069          ;
0000:3069          loc_0_3069:                                ; DATA XREF: 0000:0A80↑o
0000:3069 DF          ; 0000:0A84↑o ...
0000:3069          rst     18h
0000:306A 2A C0 63      ld      hl, (unk_0_63C0)
0000:306D 34          inc     (hl)
0000:306E C9          ret
0000:306F          ; SUBROUTINE
0000:306F
0000:306F          sub_0_306F:                                ; CODE XREF: 0000:0AE8↑p
0000:306F 21 AF 62          ; 0000:1732↑p ...
0000:306F          ld      hl, unk_0_62AF
0000:3072 34          inc     (hl)
0000:3073 7E          ld      a, (hl)
0000:3074 E6 07          and     7
0000:3076 C0          ret     nz
0000:3077 21 0B 69      ld      hl, soft_sprite_ram+0Bh
0000:307A 0E FC          ld      c, 0FCh ; '3'
0000:307C FF          rst     38h
0000:307D 0E 81          ld      c, 81h ; 'ü'
0000:307F 21 09 69      ld      hl, soft_sprite_ram+9
0000:3082 CD 96 30      call    sub_0_3096
0000:3085 21 1D 69      ld      hl, soft_sprite_ram+1Dh
0000:3088 CD 96 30      call    sub_0_3096
0000:308B CD 57 00      call    rand
0000:308E E6 80          and     80h ; 'Ç'
0000:3090 21 2D 69      ld      hl, soft_sprite_ram+2Dh
0000:3093 AE          xor     (hl)
0000:3094 77          ld      (hl), a
0000:3095 C9          ret
0000:3095          ; End of function sub_0_306F
0000:3095
0000:3096          ; SUBROUTINE
0000:3096
0000:3096          ;

```

```

0000:3096
0000:3096
0000:3096      sub_0_3096:                                ; CODE XREF: sub_0_306F+13↑p
0000:3096 06 02                                ; sub_0_306F+19↑p
0000:3096      ld      b, 2
0000:3098
0000:3098      loc_0_3098:                                ; CODE XREF: sub_0_3096+6↑j
0000:3098      ld      a, c
0000:3099 AE      xor      (hl)
0000:309A 77      ld      (hl), a
0000:309B 19      add     hl, de
0000:309C 10 FA      djnz    loc_0_3098
0000:309E C9      ret
0000:309E      ; End of function sub_0_3096
0000:309E
0000:309F      ; ██████████ SUBROUTINE ██████████
0000:309F
0000:309F      sub_0_309F:                                ; CODE XREF: sub_0_17B+3B↑p
0000:309F E5                                ; 0000:01F7↑p ...
0000:309F      push     hl
0000:30A0 21 C0 60      ld      hl, fg_vector_fn_params
0000:30A3 3A B0 60      ld      a, (unk_0_60B0)
0000:30A6 6F      ld      l, a
0000:30A7 CB 7E      bit     7, (hl)
0000:30A9 CA BB 30      jp      z, loc_0_30BB
0000:30AC 72      ld      (hl), d
0000:30AD 2C      inc     l
0000:30AE 73      ld      (hl), e
0000:30AF 2C      inc     l
0000:30B0 7D      ld      a, l
0000:30B1 FE C0      cp      0C0h ; 'L'
0000:30B3 D2 B8 30      jp      nc, loc_0_30B8
0000:30B6 3E C0      ld      a, 0C0h ; 'L'
0000:30B8
0000:30B8      loc_0_30B8:                                ; CODE XREF: sub_0_309F+14↑j
0000:30B8 32 B0 60      ld      (unk_0_60B0), a
0000:30BB
0000:30BB      loc_0_30BB:                                ; CODE XREF: sub_0_309F+A↑j
0000:30BB E1      pop     hl
0000:30BC C9      ret
0000:30BC      ; End of function sub_0_309F
0000:30BC
0000:30BD      ; ██████████ SUBROUTINE ██████████
0000:30BD
0000:30BD      sub_0_30BD:                                ; CODE XREF: 0000:12A3↑p
0000:30BD 21 50 69                                ; 0000:1615↑p
0000:30BD      ld      hl, soft_sprite_ram+50h
0000:30C0 06 02      ld      b, 2
0000:30C2 CD E4 30      call    sub_0_30E4
0000:30C5 2E 80      ld      l, 80h ; 'Ç'
0000:30C7 06 0A      ld      b, 0Ah
0000:30C9 CD E4 30      call    sub_0_30E4
0000:30CC 2E B8      ld      l, 0B8h ; '@'
0000:30CE 06 0B      ld      b, 0Bh
0000:30D0 CD E4 30      call    sub_0_30E4
0000:30D3 21 0C 6A      ld      hl, soft_sprite_ram+10Ch
0000:30D6 06 05      ld      b, 5
0000:30D8 C3 E4 30      jp      sub_0_30E4
0000:30D8      ; End of function sub_0_30BD
0000:30D8
0000:30DB      ; ██████████ SUBROUTINE ██████████
0000:30DB
0000:30DB      sub_0_30DB:                                ; CODE XREF: 0000:12DF↑p
0000:30DB 21 4C 69      ld      hl, soft_sprite_ram+4Ch
0000:30DE 36 00      ld      (hl), 0
0000:30E0 2E 58      ld      l, 58h ; 'X'
0000:30E2 06 06      ld      b, 6
0000:30E2      ; End of function sub_0_30DB
0000:30E2
0000:30E4      ; ██████████ SUBROUTINE ██████████
0000:30E4
0000:30E4      sub_0_30E4:                                ; CODE XREF: sub_0_30BD+5↑p
0000:30E4 7D                                ; sub_0_30BD+C↑p ...
0000:30E4      ld      a, 1
0000:30E5
0000:30E5      loc_0_30E5:                                ; CODE XREF: sub_0_30E4+6↑j
0000:30E5 36 00      ld      (hl), 0
0000:30E7 C6 04      add     a, 4
0000:30E9 6F      ld      l, a
0000:30EA 10 F9      djnz    loc_0_30E5
0000:30EC C9      ret
0000:30EC      ; End of function sub_0_30E4
0000:30EC
0000:30ED      ; ██████████ SUBROUTINE ██████████
0000:30ED
0000:30ED      sub_0_30ED:                                ; CODE XREF: 0000:198C↑p
0000:30ED CD FA 30      call    sub_0_30FA
0000:30F0 CD 3C 31      call    sub_0_313C
0000:30F3 CD B1 31      call    sub_0_31B1
0000:30F6 CD F3 34      call    sub_0_34F3
0000:30F9 C9      ret
0000:30F9      ; End of function sub_0_30ED
0000:30F9
0000:30FA      ; ██████████ SUBROUTINE ██████████
0000:30FA
0000:30FA      sub_0_30FA:                                ; CODE XREF: sub_0_30ED↑p
0000:30FA 3A 80 63      ld      a, (unk_0_6380)
0000:30FD FE 06      cp      6
0000:30FF 38 02      jr      c, loc_0_3103
0000:3101 3E 05      ld      a, 5
0000:3103
0000:3103      loc_0_3103:                                ; CODE XREF: sub_0_30FA+5↑j
0000:3103 EF      rst     28h
0000:3103      ; _____

```

```

0000:3104 10 31      .dw loc_0_3110      ; Jump table
0000:3106 10 31      .dw loc_0_3110
0000:3108 1B 31      .dw loc_0_311B
0000:310A 26 31      .dw loc_0_3126
0000:310C 26 31      .dw loc_0_3126
0000:310E 31 31      .dw loc_0_3131
0000:3110
0000:3110
0000:3110      loc_0_3110:      ; DATA XREF: sub_0_30FA+A|o
0000:3110 3A 1A 60      ; sub_0_30FA+C|o
0000:3110      ld      a, (gen_purpose_timer)
0000:3113 E6 01      and     1
0000:3115 FE 01      cp      1
0000:3117 C8      ret     z
0000:3118 33      inc     sp
0000:3119 33      inc     sp
0000:311A C9      ret
0000:311B
0000:311B
0000:311B      loc_0_311B:      ; DATA XREF: sub_0_30FA+E|o
0000:311B 3A 1A 60      ld      a, (gen_purpose_timer)
0000:311E E6 07      and     7
0000:3120 FE 05      cp      5
0000:3122 F8      ret     m
0000:3123 33      inc     sp
0000:3124 33      inc     sp
0000:3125 C9      ret
0000:3126
0000:3126
0000:3126      loc_0_3126:      ; DATA XREF: sub_0_30FA+10|o
0000:3126 3A 1A 60      ; sub_0_30FA+12|o
0000:3126      ld      a, (gen_purpose_timer)
0000:3129 E6 03      and     3
0000:312B FE 03      cp      3
0000:312D F8      ret     m
0000:312E 33      inc     sp
0000:312F 33      inc     sp
0000:3130 C9      ret
0000:3131
0000:3131
0000:3131      loc_0_3131:      ; DATA XREF: sub_0_30FA+14|o
0000:3131 3A 1A 60      ld      a, (gen_purpose_timer)
0000:3134 E6 07      and     7
0000:3136 FE 07      cp      7
0000:3138 F8      ret     m
0000:3139 33      inc     sp
0000:313A 33      inc     sp
0000:313B C9      ret
0000:313B      ; End of function sub_0_30FA
0000:313B
0000:313C      ; SUBROUTINE
0000:313C
0000:313C
0000:313C      sub_0_313C:      ; CODE XREF: sub_0_30ED+3|p
0000:313C DD 21 00 64      ; fireball character data
0000:3140 AF      ld      ix, unk_0_6400
0000:3141 32 A1 63      xor     a, (unk_0_63A1), a
0000:3144 06 05      ld      b, 5
0000:3146 11 20 00      ld      de, 20h ; ' '
0000:3149
0000:3149      loc_0_3149:      ; CODE XREF: sub_0_313C+30|j
0000:3149 DD 7E 00      ld      a, (ix+0)
0000:314C FE 00      cp      0
0000:314E CA 7C 31      jp      z, loc_0_317C
0000:3151 3A A1 63      ld      a, (unk_0_63A1)
0000:3154 3C      inc     a
0000:3155 32 A1 63      ld      (unk_0_63A1), a
0000:3158 3E 01      ld      a, 1
0000:315A DD 77 08      ld      (ix+8), a
0000:315D 3A 17 62      ld      a, (unk_0_6217)
0000:3160 FE 01      cp      1
0000:3162 C2 6A 31      jp      nz, loc_0_316A
0000:3165 3E 00      ld      a, 0
0000:3167 DD 77 08      ld      (ix+8), a
0000:316A
0000:316A      loc_0_316A:      ; CODE XREF: sub_0_313C+26|j
0000:316A DD 19      ; sub_0_313C+45|j ...
0000:316A
0000:316C 10 DB      add     ix, de
0000:316E 21 A0 63      djnz   loc_0_3149
0000:3171 36 00      ld      hl, unk_0_63A0
0000:3173 3A A1 63      ld      (hl), 0
0000:3176 FE 00      ld      a, (unk_0_63A1)
0000:3178 C0      cp      0
0000:3179 33      ret     nz
0000:317A 33      inc     sp
0000:317B C9      inc     sp
0000:317C      ret
0000:317C
0000:317C      loc_0_317C:      ; CODE XREF: sub_0_313C+12|j
0000:317C 3A A1 63      ld      a, (unk_0_63A1)
0000:317F FE 05      cp      5
0000:3181 CA 6A 31      jp      z, loc_0_316A
0000:3184 3A 27 62      ld      a, (level_type)
0000:3187 FE 02      cp      2
0000:3189 C2 95 31      jp      nz, loc_0_3195
0000:318C 3A A1 63      ld      a, (unk_0_63A1)
0000:318F 4F      ld      c, a
0000:3190 3A 80 63      ld      a, (unk_0_6380)
0000:3193 B9      cp      c
0000:3194 C8      ret     z
0000:3195
0000:3195      loc_0_3195:      ; CODE XREF: sub_0_313C+4D|j
0000:3195 3A A0 63      ; spawn a fireball
0000:3198 FE 01      ld      a, (unk_0_63A0)
0000:319A C2 6A 31      cp      1
0000:319D DD 77 00      jp      nz, loc_0_316A
0000:31A0 DD 77 18      ld      (ix+0), a
0000:31A3 AF      ld      (ix+18h), a
0000:31A4 32 A0 63      xor     a, (unk_0_63A0), a
0000:31A7 3A A1 63      ld      a, (unk_0_63A1)
0000:31AA 3C      inc     a
0000:31AB 32 A1 63      ld      (unk_0_63A1), a
0000:31AE C3 6A 31      jp      loc_0_316A
0000:31AE      ; End of function sub_0_313C

```

```

0000:31AE
0000:31B1
0000:31B1 ; SUBROUTINE
0000:31B1
0000:31B1
0000:31B1
0000:31B1 sub_0_31B1: ; CODE XREF: sub_0_30ED+61p
0000:31B1 CD DD 31 call sub_0_31DD
0000:31B4 AF xor a
0000:31B5 32 A2 63 ld (unk_0_63A2), a
0000:31B8 21 E0 63 ld hl, unk_0_63E0
0000:31BB 22 C8 63 ld (unk_0_63C8), hl
0000:31BE
0000:31BE loc_0_31BE: ; CODE XREF: sub_0_31B1+281j
0000:31BE 2A C8 63 ld hl, (unk_0_63C8)
0000:31C1 01 20 00 ld bc, 20h ; ' '
0000:31C4 09 add hl, bc
0000:31C5 22 C8 63 ld (unk_0_63C8), hl
0000:31C8 7E ld a, (hl)
0000:31C9 A7 and a
0000:31CA CA D0 31 jp z, loc_0_31D0
0000:31CD CD 02 32 call sub_0_3202
0000:31D0
0000:31D0 loc_0_31D0: ; CODE XREF: sub_0_31B1+191j
0000:31D0 3A A2 63 ld a, (unk_0_63A2)
0000:31D3 3C inc a
0000:31D4 32 A2 63 ld (unk_0_63A2), a
0000:31D7 FE 05 cp 5
0000:31D9 C2 BE 31 jp nz, loc_0_31BE
0000:31DC C9 ret
0000:31DC ; End of function sub_0_31B1
0000:31DD
0000:31DD ; SUBROUTINE
0000:31DD
0000:31DD sub_0_31DD: ; CODE XREF: sub_0_31B11p
0000:31DD 3A 80 63 ld a, (unk_0_6380)
0000:31E0 FE 03 cp 3
0000:31E2 F8 ret m
0000:31E3 CD F6 31 call sub_0_31F6
0000:31E6 FE 01 cp 1
0000:31E8 C0 ret nz
0000:31E9 21 39 64 ld hl, unk_0_6439
0000:31EC 3E 02 ld a, 2
0000:31EE 77 ld (hl), a
0000:31EF 21 79 64 ld hl, unk_0_6479
0000:31F2 3E 02 ld a, 2
0000:31F4 77 ld (hl), a
0000:31F5 C9 ret
0000:31F5 ; End of function sub_0_31DD
0000:31F6
0000:31F6 ; SUBROUTINE
0000:31F6
0000:31F6 sub_0_31F6: ; CODE XREF: sub_0_31DD+61p
0000:31F6 3A 18 60 ld a, (random_no)
0000:31F9 E6 03 and 3
0000:31FB FE 01 cp 1
0000:31FD C0 ret nz
0000:31FE 3A 1A 60 ld a, (gen_purpose_timer)
0000:3201 C9 ret
0000:3201 ; End of function sub_0_31F6
0000:3202
0000:3202 ; SUBROUTINE
0000:3202
0000:3202 sub_0_3202: ; CODE XREF: sub_0_31B1+1C1p
0000:3202 DD 2A C8 63 ld ix, (unk_0_63C8)
0000:3206 DD 7E 18 ld a, (ix+18h)
0000:3209 FE 01 cp 1
0000:320B CA 7A 32 jp z, loc_0_327A
0000:320E DD 7E 0D ld a, (ix+0Dh)
0000:3211 FE 04 cp 4
0000:3213 F2 30 32 jp p, loc_0_3230
0000:3216 DD 7E 19 ld a, (ix+19h)
0000:3219 FE 02 cp 2
0000:321B CA 7E 32 jp z, loc_0_327E
0000:321E CD 0F 33 call sub_0_330F
0000:3221 3A 18 60 ld a, (random_no)
0000:3224 E6 03 and 3
0000:3226 C2 33 32 jp nz, loc_0_3233
0000:3229
0000:3229 loc_0_3229: ; CODE XREF: sub_0_3202+7F1j
0000:3229 DD 7E 0D ld a, (ix+0Dh)
0000:322C A7 and a
0000:322D CA 57 32 jp z, loc_0_3257
0000:3230
0000:3230 loc_0_3230: ; CODE XREF: sub_0_3202+111j
0000:3230 CD 3D 33 call sub_0_333D
0000:3233
0000:3233 loc_0_3233: ; CODE XREF: sub_0_3202+241j
0000:3233 DD 7E 0D ld a, (ix+0Dh)
0000:3236 FE 04 cp 4
0000:3238 F2 91 32 jp p, loc_0_3291
0000:323B CD AD 33 call sub_0_33AD
0000:323E CD 8C 29 call sub_0_298C
0000:3241 FE 01 cp 1
0000:3243 CA 97 32 jp z, loc_0_3297
0000:3246 DD 2A C8 63 ld ix, (unk_0_63C8)
0000:324A DD 7E 0E ld a, (ix+0Eh)
0000:324D FE 10 cp 10h
0000:324F DA 8C 32 jp c, loc_0_328C
0000:3252 FE F0 cp 0F0h ; '-'
0000:3254 D2 84 32 jp nc, loc_0_3284
0000:3257
0000:3257 loc_0_3257: ; CODE XREF: sub_0_3202+2B1j
0000:3257 DD 7E 13 ; sub_0_3202+871j ...
0000:3257 ld a, (ix+13h)
0000:325A FE 00 cp 0
0000:325C C2 B9 32 jp nz, loc_0_32B9
0000:325F 3E 11 ld a, 11h
0000:3261
0000:3261 loc_0_3261: ; CODE XREF: sub_0_3202+B81j
0000:3261 DD 77 13 ld (ix+13h), a

```

```

0000:3264 16 00      ld      d, 0
0000:3266 5F        ld      e, a
0000:3267 21 7A 3A     ld      hl, fireball_bouncing_data
0000:326A 19        add     hl, de
0000:326B 7E        ld      a, (hl)
0000:326C DD 46 0E     ld      b, (ix+0Eh)
0000:326F DD 70 03     ld      (ix+3), b
0000:3272 DD 4E 0F     ld      c, (ix+0Fh)
0000:3275 81        add     a, c
0000:3276 DD 77 05     ld      (ix+5), a
0000:3279 C9        ret

;
0000:327A
0000:327A
0000:327A      loc_0_327A:                                ; CODE XREF: sub_0_3202+9↑j
0000:327A CD BD 32      call    sub_0_32BD
0000:327D C9        ret

;
0000:327E
0000:327E      loc_0_327E:                                ; CODE XREF: sub_0_3202+19↑j
0000:327E CD D6 32      call    sub_0_32D6
0000:3281 C3 29 32      jp      loc_0_3229

;
0000:3284
0000:3284      loc_0_3284:                                ; CODE XREF: sub_0_3202+52↑j
0000:3284 3E 02        ld      a, 2
0000:3286
0000:3286      loc_0_3286:                                ; CODE XREF: sub_0_3202+8C↑j
0000:3286 DD 77 0D     ld      (ix+0Dh), a
0000:3289 C3 57 32      jp      loc_0_3257

;
0000:328C
0000:328C      loc_0_328C:                                ; CODE XREF: sub_0_3202+4D↑j
0000:328C 3E 01        ld      a, 1
0000:328E C3 86 32      jp      loc_0_3286

;
0000:3291
0000:3291      loc_0_3291:                                ; CODE XREF: sub_0_3202+36↑j
0000:3291 CD E7 33      call    sub_0_33E7
0000:3294 C3 57 32      jp      loc_0_3257

;
0000:3297
0000:3297      loc_0_3297:                                ; CODE XREF: sub_0_3202+41↑j
0000:3297 DD 2A C8 63   ld      ix, (unk_0_63C8)
0000:329B DD 7E 0D     ld      a, (ix+0Dh)
0000:329E FE 01        cp      1
0000:32A0 C2 B1 32      jp      nz, loc_0_32B1
0000:32A3 3E 02        ld      a, 2
0000:32A5 DD 35 0E     dec     (ix+0Eh)
0000:32A8
0000:32A8      loc_0_32A8:                                ; CODE XREF: sub_0_3202+B4↑j
0000:32A8 DD 77 0D     ld      (ix+0Dh), a
0000:32AB CD C3 33      call    sub_0_33C3
0000:32AE C3 57 32      jp      loc_0_3257

;
0000:32B1
0000:32B1      loc_0_32B1:                                ; CODE XREF: sub_0_3202+9E↑j
0000:32B1 3E 01        ld      a, 1
0000:32B3 DD 34 0E     inc     (ix+0Eh)
0000:32B6 C3 A8 32      jp      loc_0_32A8

;
0000:32B9
0000:32B9      loc_0_32B9:                                ; CODE XREF: sub_0_3202+5A↑j
0000:32B9 3D        dec     a
0000:32BA C3 61 32      jp      loc_0_3261
0000:32BA      ; End of function sub_0_3202
0000:32BA
0000:32BD
0000:32BD      ; SUBROUTINE
0000:32BD
0000:32BD      sub_0_32BD:                                ; CODE XREF: sub_0_3202+78↑p
0000:32BD 3A 27 62     ld      a, (level_type)
0000:32C0 FE 01        cp      1
0000:32C2 CA CE 32      jp      z, loc_0_32CE
0000:32C5 FE 02        cp      2
0000:32C7 CA D2 32      jp      z, loc_0_32D2
0000:32CA CD B9 34      call    sub_0_34B9
0000:32CD C9        ret

;
0000:32CE
0000:32CE      loc_0_32CE:                                ; CODE XREF: sub_0_32BD+5↑j
0000:32CE CD 2C 34      call    sub_0_342C
0000:32D1 C9        ret

;
0000:32D2
0000:32D2      loc_0_32D2:                                ; CODE XREF: sub_0_32BD+A↑j
0000:32D2 CD 78 34      call    sub_0_3478
0000:32D5 C9        ret
0000:32D5      ; End of function sub_0_32BD
0000:32D5
0000:32D6
0000:32D6      ; SUBROUTINE
0000:32D6
0000:32D6      sub_0_32D6:                                ; CODE XREF: sub_0_3202+7C↑p
0000:32D6 DD 7E 1C     ld      a, (ix+1Ch)
0000:32D9 FE 00        cp      0
0000:32DB C2 FD 32      jp      nz, loc_0_32FD
0000:32DE DD 7E 1D     ld      a, (ix+1Dh)
0000:32E1 FE 01        cp      1
0000:32E3 C2 0B 33      jp      nz, loc_0_330B
0000:32E6 DD 36 1D 00   ld      (ix+1Dh), 0
0000:32EA 3A 05 62     ld      a, (mario_x_coord)
0000:32ED DD 46 0F     ld      b, (ix+0Fh)
0000:32F0 90        sub     b
0000:32F1 DA 03 33      jp      c, loc_0_3303
0000:32F4 DD 36 1C FF     ld      (ix+1Ch), 0FFh
0000:32F8
0000:32F8      loc_0_32F8:                                ; CODE XREF: sub_0_32D6+2A↑j
0000:32F8 DD 36 0D 00     ld      (ix+0Dh), 0
0000:32FC C9        ret

;
0000:32FD
0000:32FD      loc_0_32FD:                                ; CODE XREF: sub_0_32D6+5↑j
0000:32FD DD 35 1C     dec     (ix+1Ch)
0000:3300 C2 F8 32      jp      nz, loc_0_32F8

```

```

0000:3303      loc_0_3303:                                ; CODE XREF: sub_0_32D6+1B↑j
0000:3303      ld      (ix+19h), 0
0000:3307      dd      36 19 00      ld      (ix+1Ch), 0
0000:330B      loc_0_330B:                                ; CODE XREF: sub_0_32D6+D↑j
0000:330B      cd      0F 33      call     sub_0_330F
0000:330E      C9                                ret
0000:330E      ; End of function sub_0_32D6
0000:330F      ; ██████████ SUBROUTINE ██████████
0000:330F      sub_0_330F:                                ; CODE XREF: sub_0_3202+1C↑p
0000:330F      dd      7E 16                                ; sub_0_32D6+35↑p
0000:3312      FE 00      ld      a, (ix+16h)
0000:3314      C2 32 33      cp      0
0000:3317      DD 36 16 2B      jp      nz, loc_0_3332
0000:331B      DD 36 0D 00      ld      (ix+16h), 2Bh ; '+'
0000:331F      3A 18 60      ld      (ix+0Dh), 0
0000:3322      0F      ld      a, (random_no)
0000:3323      D2 32 33      rrca
0000:3326      DD 7E 0D      jp      nc, loc_0_3332
0000:3329      FE 01      ld      a, (ix+0Dh)
0000:332B      CA 36 33      cp      1
0000:332E      DD 36 0D 01      jp      z, loc_0_3336
0000:3332      ld      (ix+0Dh), 1
0000:3332      loc_0_3332:                                ; CODE XREF: sub_0_330F+5↑j
0000:3332      dd      35 16                                ; sub_0_330F+14↑j ...
0000:3335      C9                                dec      (ix+16h)
0000:3336      ret
0000:3336      ;
0000:3336      loc_0_3336:                                ; CODE XREF: sub_0_330F+1C↑j
0000:3336      dd      36 0D 02      ld      (ix+0Dh), 2
0000:333A      C3 32 33      C3 32 33      jp      loc_0_3332
0000:333A      ; End of function sub_0_330F
0000:333D      ; ██████████ SUBROUTINE ██████████
0000:333D      sub_0_333D:                                ; CODE XREF: sub_0_3202+2E↑p
0000:333D      dd      7E 0D      ld      a, (ix+0Dh)
0000:3340      FE 08      cp      8
0000:3342      CA 71 33      jp      z, loc_0_3371
0000:3345      FE 04      cp      4
0000:3347      CA 8A 33      jp      z, loc_0_338A
0000:334A      CD A1 33      call    sub_0_33A1
0000:334D      DD 7E 0F      ld      a, (ix+0Fh)
0000:3350      C6 08      add     a, 8
0000:3352      57      ld      d, a
0000:3353      DD 7E 0E      ld      a, (ix+0Eh)
0000:3356      01 15 00      ld      bc, 15h
0000:3359      CD 6E 23      call    sub_0_236E
0000:335C      A7      and     a
0000:335D      CA 99 33      jp      z, loc_0_3399
0000:3360      DD 70 1F      ld      (ix+1Fh), b
0000:3363      3A 05 62      ld      a, (mario_x_coord)
0000:3366      47      ld      b, a
0000:3367      DD 7E 0F      ld      a, (ix+0Fh)
0000:336A      90      sub     b
0000:336B      D0      ret     nc
0000:336C      DD 36 0D 04      ld      (ix+0Dh), 4
0000:3370      C9                                ret
0000:3371      ;
0000:3371      loc_0_3371:                                ; CODE XREF: sub_0_333D+5↑j
0000:3371      dd      7E 0F      ld      a, (ix+0Fh)
0000:3374      C6 08      add     a, 8
0000:3376      DD 46 1F      ld      b, (ix+1Fh)
0000:3379      B8      cp      b
0000:337A      C0      ret     nz
0000:337B      DD 36 0D 00      ld      (ix+0Dh), 0
0000:337F      DD 7E 19      ld      a, (ix+19h)
0000:3382      FE 02      cp      2
0000:3384      C0      ret     nz
0000:3385      DD 36 1D 01      ld      (ix+1Dh), 1
0000:3389      C9                                ret
0000:338A      ;
0000:338A      loc_0_338A:                                ; CODE XREF: sub_0_333D+A↑j
0000:338A      dd      7E 0F      ld      a, (ix+0Fh)
0000:338D      C6 08      add     a, 8
0000:338F      DD 46 1F      ld      b, (ix+1Fh)
0000:3392      B8      cp      b
0000:3393      C0      ret     nz
0000:3394      DD 36 0D 00      ld      (ix+0Dh), 0
0000:3398      C9                                ret
0000:3399      ;
0000:3399      loc_0_3399:                                ; CODE XREF: sub_0_333D+20↑j
0000:3399      dd      70 1F      ld      (ix+1Fh), b
0000:339C      DD 36 0D 08      ld      (ix+0Dh), 8
0000:33A0      C9                                ret
0000:33A0      ; End of function sub_0_333D
0000:33A1      ; ██████████ SUBROUTINE ██████████
0000:33A1      sub_0_33A1:                                ; CODE XREF: sub_0_333D+D↑p
0000:33A1      3E 07      ld      a, 7
0000:33A3      F7      rst     30h
0000:33A4      DD 7E 0F      ld      a, (ix+0Fh)
0000:33A7      FE 59      cp      59h ; 'Y'
0000:33A9      D0      ret     nc
0000:33AA      33      inc     sp
0000:33AB      33      inc     sp
0000:33AC      C9                                ret
0000:33AC      ; End of function sub_0_33A1
0000:33AD      ; ██████████ SUBROUTINE ██████████

```

```

0000:33AD
0000:33AD
0000:33AD      sub_0_33AD:                                ; CODE XREF: sub_0_3202+39↑p
0000:33AD DD 7E 0D      ld      a, (ix+0Dh)
0000:33B0 FE 01      cp      1
0000:33B2 CA D9 33      jp      z, loc_0_33D9
0000:33B5 DD 7E 07      ld      a, (ix+7)
0000:33B8 E6 7F      and     7Fh ; ' '
0000:33BA DD 77 07      ld      (ix+7), a
0000:33BD DD 35 0E      dec     (ix+0Eh)
0000:33C0
0000:33C0      loc_0_33C0:                                ; CODE XREF: 0000:33E4↑j
0000:33C0 CD 09 34      call    sub_0_3409
0000:33C0      ; End of function sub_0_33AD
0000:33C0
0000:33C3      ; ██████████ S U B R O U T I N E ██████████
0000:33C3
0000:33C3      sub_0_33C3:                                ; CODE XREF: sub_0_3202+A9↑p
0000:33C3 3A 27 62      ld      a, (level_type)
0000:33C6 FE 01      cp      1
0000:33C8 C0          ret      nz
0000:33C9 DD 66 0E      ld      h, (ix+0Eh)
0000:33CC DD 6E 0F      ld      l, (ix+0Fh)
0000:33CF DD 46 0D      ld      b, (ix+0Dh)
0000:33D2 CD 33 23      call    sub_0_2333
0000:33D5 DD 75 0F      ld      (ix+0Fh), l
0000:33D8 C9          ret
0000:33D8      ; End of function sub_0_33C3
0000:33D8
0000:33D9
0000:33D9      loc_0_33D9:                                ; CODE XREF: sub_0_33AD+5↑j
0000:33D9 DD 7E 07      ld      a, (ix+7)
0000:33DC F6 80      or      80h ; 'C'
0000:33DE DD 77 07      ld      (ix+7), a
0000:33E1 DD 34 0E      inc     (ix+0Eh)
0000:33E4 C3 C0 33      jp      loc_0_33C0
0000:33E7
0000:33E7      ; ██████████ S U B R O U T I N E ██████████
0000:33E7
0000:33E7      sub_0_33E7:                                ; CODE XREF: sub_0_3202+8F↑p
0000:33E7 CD 09 34      call    sub_0_3409
0000:33EA DD 7E 0D      ld      a, (ix+0Dh)
0000:33ED FE 08      cp      8
0000:33EF C2 05 34      jp      nz, loc_0_3405
0000:33F2 DD 7E 14      ld      a, (ix+14h)
0000:33F5 A7          and     a
0000:33F6 C2 01 34      jp      nz, loc_0_3401
0000:33F9 DD 36 14 02      ld      (ix+14h), 2
0000:33FD DD 35 0F      dec     (ix+0Fh)
0000:3400 C9          ret
0000:3401
0000:3401      loc_0_3401:                                ; CODE XREF: sub_0_33E7+F↑j
0000:3401 DD 35 14      dec     (ix+14h)
0000:3404 C9          ret
0000:3405
0000:3405      loc_0_3405:                                ; CODE XREF: sub_0_33E7+8↑j
0000:3405 DD 34 0F      inc     (ix+0Fh)
0000:3408 C9          ret
0000:3408      ; End of function sub_0_33E7
0000:3408
0000:3409      ; ██████████ S U B R O U T I N E ██████████
0000:3409
0000:3409      sub_0_3409:                                ; CODE XREF: sub_0_33AD+13↑p
0000:3409 DD 7E 15      ld      a, (ix+15h)
0000:3409      ; sub_0_33E7↑p
0000:340C A7          and     a
0000:340D C2 28 34      jp      nz, loc_0_3428
0000:3410 DD 36 15 02      ld      (ix+15h), 2
0000:3414 DD 34 07      inc     (ix+7)
0000:3417 DD 7E 07      ld      a, (ix+7)
0000:341A E6 0F      and     0Fh
0000:341C FE 0F      cp      0Fh
0000:341E C0          ret      nz
0000:341F DD 7E 07      ld      a, (ix+7)
0000:3422 EE 02      xor     2
0000:3424 DD 77 07      ld      (ix+7), a
0000:3427 C9          ret
0000:3428
0000:3428      loc_0_3428:                                ; CODE XREF: sub_0_3409+4↑j
0000:3428 DD 35 15      dec     (ix+15h)
0000:342B C9          ret
0000:342B      ; End of function sub_0_3409
0000:342B
0000:342C      ; ██████████ S U B R O U T I N E ██████████
0000:342C
0000:342C      sub_0_342C:                                ; CODE XREF: sub_0_32BD+11↑p
0000:342C DD 6E 1A      ld      l, (ix+1Ah)
0000:342F DD 66 1B      ld      h, (ix+1Bh)
0000:3432 AF          xor     a
0000:3433 01 00 00      ld      bc, 0
0000:3436 ED 4A      adc     hl, bc
0000:3438 C2 42 34      jp      nz, loc_0_3442
0000:343B 21 8C 3A      ld      hl, fireball_bounce_data
0000:343E DD 36 03 26      ld      (ix+3), 26h ; '&'
0000:3442
0000:3442      loc_0_3442:                                ; CODE XREF: sub_0_342C+C↑j
0000:3442 DD 34 03      inc     (ix+3)
0000:3445
0000:3445      loc_0_3445:                                ; CODE XREF: sub_0_3478+2D↑j
0000:3445      ; sub_0_3478+3E↑j
0000:3445 7E          ld      a, (hl)
0000:3446 FE AA      cp      0AAh ; '¬'
0000:3448 CA 56 34      jp      z, loc_0_3456
0000:344B DD 77 05      ld      (ix+5), a
0000:344E 23          inc     hl

```

```

0000:344F DD 75 1A      ld      (ix+1Ah), l
0000:3452 DD 74 1B      ld      (ix+1Bh), h
0000:3455 C9              ret
0000:3456
0000:3456
0000:3456      loc_0_3456:                                ; CODE XREF: sub_0_342C+1C1j
0000:3456 AF              xor      a
0000:3457 DD 77 13      ld      (ix+13h), a
0000:345A DD 77 18      ld      (ix+18h), a
0000:345D DD 77 0D      ld      (ix+0Dh), a
0000:3460 DD 77 1C      ld      (ix+1Ch), a
0000:3463 DD 7E 03      ld      a, (ix+3)
0000:3466 DD 77 0E      ld      (ix+0Eh), a
0000:3469 DD 7E 05      ld      a, (ix+5)
0000:346C DD 77 0F      ld      (ix+0Fh), a
0000:346F DD 36 1A 00      ld      (ix+1Ah), 0
0000:3473 DD 36 1B 00      ld      (ix+1Bh), 0
0000:3477 C9              ret
0000:3477      ; End of function sub_0_342C
0000:3477
0000:3478      ; ██████████ SUBROUTINE ██████████
0000:3478
0000:3478      sub_0_3478:                                ; CODE XREF: sub_0_32BD+151p
0000:3478 DD 6E 1A      ld      l, (ix+1Ah)
0000:347B DD 66 1B      ld      h, (ix+1Bh)
0000:347E AF              xor      a
0000:347F 01 00 00      ld      bc, 0
0000:3482 ED 4A      adc     hl, bc
0000:3484 C2 9A 34      jp      nz, loc_0_349A
0000:3487 21 AC 3A      ld      hl, cement_fireball_data
0000:348A 3A 03 62      ld      a, (mario_y_coord)
0000:348D CB 7F      bit     7, a
0000:348F CA A8 34      jp      z, loc_0_34A8
0000:3492 DD 36 0D 01      ld      (ix+0Dh), 1
0000:3496 DD 36 03 7E      ld      (ix+3), 7Eh ; '~'
0000:349A
0000:349A      loc_0_349A:                                ; CODE XREF: sub_0_3478+C1j
0000:349A DD 7E 0D                                ; sub_0_3478+381j
0000:349A      ld      a, (ix+0Dh)
0000:349D FE 01      cp      1
0000:349F C2 B3 34      jp      nz, loc_0_34B3
0000:34A2 DD 34 03      inc     (ix+3)
0000:34A5 C3 45 34      jp      loc_0_3445
0000:34A8
0000:34A8
0000:34A8      loc_0_34A8:                                ; CODE XREF: sub_0_3478+171j
0000:34A8 DD 36 0D 02      ld      (ix+0Dh), 2
0000:34AC DD 36 03 80      ld      (ix+3), 80h ; 'Q'
0000:34B0 C3 9A 34      jp      loc_0_349A
0000:34B3
0000:34B3
0000:34B3      loc_0_34B3:                                ; CODE XREF: sub_0_3478+271j
0000:34B3 DD 35 03      dec     (ix+3)
0000:34B6 C3 45 34      jp      loc_0_3445
0000:34B6      ; End of function sub_0_3478
0000:34B6
0000:34B9      ; ██████████ SUBROUTINE ██████████
0000:34B9
0000:34B9      sub_0_34B9:                                ; CODE XREF: sub_0_32BD+D1p
0000:34B9 3A 27 62      ld      a, (level_type)
0000:34BC FE 03      cp      3
0000:34BE C8              ret     z
0000:34BF 3A 03 62      ld      a, (mario_y_coord)
0000:34C2 CB 7F      bit     7, a
0000:34C4 C2 ED 34      jp      nz, loc_0_34ED
0000:34C7 21 C4 3A      ld      hl, rivet_fireball_data
0000:34CA
0000:34CA      loc_0_34CA:                                ; CODE XREF: sub_0_34B9+371j
0000:34CA 06 00      ld      b, 0
0000:34CC 3A 19 60      ld      a, (random_no+1)
0000:34CF E6 06      and     a, 6
0000:34D1 4F      ld      c, a
0000:34D2 09      add     hl, bc
0000:34D3 7E      ld      a, (hl)
0000:34D4 DD 77 03      ld      (ix+3), a
0000:34D7 DD 77 0E      ld      (ix+0Eh), a
0000:34DA 23      inc     hl
0000:34DB 7E      ld      a, (hl)
0000:34DC DD 77 05      ld      (ix+5), a
0000:34DF DD 77 0F      ld      (ix+0Fh), a
0000:34E2 AF              xor      a
0000:34E3 DD 77 0D      ld      (ix+0Dh), a
0000:34E6 DD 77 18      ld      (ix+18h), a
0000:34E9 DD 77 1C      ld      (ix+1Ch), a
0000:34EC C9              ret
0000:34ED
0000:34ED
0000:34ED      loc_0_34ED:                                ; CODE XREF: sub_0_34B9+B1j
0000:34ED 21 D4 3A      ld      hl, rivet_fireball_start_points
0000:34F0 C3 CA 34      jp      loc_0_34CA
0000:34F0      ; End of function sub_0_34B9
0000:34F0
0000:34F3      ; ██████████ SUBROUTINE ██████████
0000:34F3
0000:34F3      sub_0_34F3:                                ; CODE XREF: sub_0_30ED+91p
0000:34F3 21 00 64      ld      hl, unk_0_6400                ; fireball character data
0000:34F6 11 D0 69      ld      de, soft_sprite_ram+0D0h      ; fireballs in sprite ram
0000:34F9 06 05      ld      b, 5                          ; 5 fireballs (max)
0000:34FB
0000:34FB      loc_0_34FB:                                ; CODE XREF: sub_0_34F3+281j
0000:34FB 7E      ld      a, (hl)
0000:34FC A7      and     a
0000:34FD CA 1E 35      jp      z, loc_0_351E
0000:3500 2C      inc     l
0000:3501 2C      inc     l
0000:3502 2C      inc     l
0000:3503 7E      ld      a, (hl)                      ; fireball X coordinate
0000:3504 12      ld      (de), a
0000:3505 3E 04      ld      a, 4
0000:3507 85      add     a, 1

```



```

0000:3508 6F      ld      l, a
0000:3509 1C      inc     e
0000:350A 7E      ld      a, (hl)
0000:350B 12      ld      (de), a
0000:350C 2C      inc     l
0000:350D 1C      inc     e
0000:350E 7E      ld      a, (hl)
0000:350F 12      ld      (de), a
0000:3510 2D      dec     l
0000:3511 2D      dec     l
0000:3512 2D      dec     l
0000:3513 1C      inc     e
0000:3514 7E      ld      a, (hl)
0000:3515 12      ld      (de), a
0000:3516 13      inc     de
0000:3517      loc_0_3517:
0000:3517      ; CODE XREF: sub_0_34F3+33|j
0000:3517 3E 1B      ld      a, 1Bh
0000:3519 85      add     a, l
0000:351A 6F      ld      l, a
0000:351B 10 DE      djnz    loc_0_34FB
0000:351D C9      ret
0000:351E      ;
0000:351E      loc_0_351E:
0000:351E      ; CODE XREF: sub_0_34F3+A|j
0000:351E 3E 05      ld      a, 5
0000:3520 85      add     a, l
0000:3521 6F      ld      l, a
0000:3522 3E 04      ld      a, 4
0000:3524 83      add     a, e
0000:3525 5F      ld      e, a
0000:3526 C3 17 35      jp      loc_0_3517
0000:3526      ; End of function sub_0_34F3
0000:3526      ;
0000:3529 00 00 00      Bonus_Points_Tbl:.db 0, 0, 0
0000:3529      ; DATA XREF: add_bonus_and_update_high_score+9|o
0000:3529      ; 0 pts
0000:352C 00 01 00      .db 0, 1, 0
0000:352C      ; 100 pts
0000:352F 00 02 00      .db 0, 2, 0
0000:352F      ; 200 pts
0000:3532 00 03 00      .db 0, 3, 0
0000:3532      ; 300 pts
0000:3535 00 04 00      .db 0, 4, 0
0000:3535      ; 400 pts
0000:3538 00 05 00      .db 0, 5, 0
0000:3538      ; 500 pts
0000:353B 00 06 00      .db 0, 6, 0
0000:353B      ; 600 pts
0000:353E 00 07 00      .db 0, 7, 0
0000:353E      ; 700 pts
0000:3541 00 08 00      .db 0, 8, 0
0000:3541      ; 800 pts
0000:3544 00 09 00      .db 0, 9, 0
0000:3544      ; 900 pts
0000:3547 00 00 00      .db 0, 0, 0
0000:3547      ; 0 pts
0000:354A 00 10 00      .db 0, 10h, 0
0000:354A      ; 1000 pts
0000:354D 00 20 00      .db 0, 20h, 0
0000:354D      ; 2000 pts
0000:3550 00 30 00      .db 0, 30h, 0
0000:3550      ; 3000 pts
0000:3553 00 40 00      .db 0, 40h, 0
0000:3553      ; 4000 pts
0000:3556 00 50 00      .db 0, 50h, 0
0000:3556      ; 5000 pts
0000:3559 00 60 00      .db 0, 60h, 0
0000:3559      ; 6000 pts
0000:355C 00 70 00      .db 0, 70h, 0
0000:355C      ; 7000 pts
0000:355F 00 80 00      .db 0, 80h, 0
0000:355F      ; 8000 pts
0000:3562 00 90 00      .db 0, 90h, 0
0000:3562      ; 9000 pts
0000:3565 94 77      high_score_tbl:.dw VRAM_start+394h
0000:3565      ; DATA XREF: read_dips_and_high_score_tbl+53|o
0000:3567 01 23 24 10+      .db 1, 23h, 24h, 10h, 10h, 0, 0, 7, 6, 5, 0, 10h, 10h
0000:3567 10 00 00 07+      .db 10h, 10h, 10h, 10h, 10h, 10h, 10h, 10h, 10h, 10h
0000:3567 06 05 00 10+      .db 10h, 3Fh, 0, 50h, 76h, 0
0000:3568 F4 76      .dw VRAM_start+2F4h
0000:3568      .dw VRAM_start+396h
0000:3569 02 1E 14 10+      .db 2, 1Eh, 14h, 10h, 10h, 0, 0, 6, 1, 0, 0, 10h, 10h
0000:3569 10 00 00 06+      .db 10h, 10h, 10h, 10h, 10h, 10h, 10h, 10h, 10h, 10h
0000:3569 01 00 00 10+      .db 10h, 3Fh, 0, 0, 61h, 0
0000:3569 F6 76      .dw VRAM_start+2F6h
0000:3569 98 77      .dw VRAM_start+398h
0000:3569 03 22 14 10+      .db 3, 22h, 14h, 10h, 10h, 0, 0, 5, 9, 5, 0, 10h, 10h
0000:3569 10 00 00 05+      .db 10h, 10h, 10h, 10h, 10h, 10h, 10h, 10h, 10h, 10h
0000:3569 09 05 00 10+      .db 10h, 3Fh, 0, 50h, 59h, 0
0000:3569 F8 76      .dw VRAM_start+2F8h
0000:3569 9A 77      .dw VRAM_start+39Ah
0000:3569 04 24 18 10+      .db 4, 24h, 18h, 10h, 10h, 0, 0, 5, 0, 5, 0, 10h, 10h
0000:3569 10 00 00 05+      .db 10h, 10h, 10h, 10h, 10h, 10h, 10h, 10h, 10h, 10h
0000:3569 00 05 00 10+      .db 10h, 3Fh, 0, 50h, 50h, 0
0000:3569 FA 76      .dw VRAM_start+2FAh
0000:3569 9C 77      .dw VRAM_start+39Ch
0000:3569 05 24 18 10+      .db 5, 24h, 18h, 10h, 10h, 0, 0, 4, 3, 0, 0, 10h, 10h
0000:3569 10 00 00 04+      .db 10h, 10h, 10h, 10h, 10h, 10h, 10h, 10h, 10h, 10h
0000:3569 03 00 00 10+      .db 10h, 3Fh, 0, 0, 43h, 0
0000:3569 FC 76      .dw VRAM_start+2FCh
0000:3569 3B 5C 4B 5C+letter_coords: .db 3Bh, 5Ch, 4Bh, 5Ch, 5Bh, 5Ch, 6Bh, 5Ch, 7Bh, 5Ch, 8Bh
0000:3569 5B 5C 6B 5C+      ; DATA XREF: sub_0_15FA+4|o
0000:3569 7B 5C 8B 5C+      .db 5Ch, 9Bh, 5Ch, 0ABh, 5Ch, 0BBh, 5Ch, 0CBh, 5Ch, 3Bh
0000:3569 9B 5C AB 5C+      .db 6Ch, 4Bh, 6Ch, 5Bh, 6Ch, 6Bh, 6Ch, 7Bh, 6Ch, 8Bh, 6Ch
0000:3569 BB 5C CB 5C+      .db 9Bh, 6Ch, 0ABh, 6Ch, 0BBh, 6Ch, 0CBh, 6Ch, 3Bh, 7Ch
0000:3569 3B 6C 4B 6C+      .db 4Bh, 7Ch, 5Bh, 7Ch, 6Bh, 7Ch, 7Bh, 7Ch, 8Bh, 7Ch, 9Bh
0000:3569 5B 6C 6B 6C+      .db 7Ch, 0ABh, 7Ch, 0BBh, 7Ch, 0CBh, 7Ch
0000:3569 8B 36      message_table: .dw aGAME_OVER
0000:3569 01 00      .dw 1
0000:3569 98 36      .dw aPLAYER
0000:3569 A5 36      .dw aPLAYER_
0000:3569 B2 36      .dw aHIGH_SCORE
0000:3569 BF 36      .dw aCREDIT
0000:3569 06 00      .dw 6
0000:3569 CC 36      .dw aHOW_HIGH_CAN_YOU_GET
0000:3569 08 00      .dw 8
0000:3569 E6 36      .dw aONLY_1_PLAYER_BUTTON
0000:3569 FD 36      .dw a1_OR_2_PLAYERS
0000:3569 08 00      .dw 0Bh
0000:3569 15 37      .dw aPUSH
0000:3569 1C 37      .dw aNAME_REGISTRATION
0000:3569 30 37      .dw aNAME
0000:3569 38 37      .dw a???
0000:3569 47 37      .dw aA_B_C_D_E_F_G_H_I_J
0000:3569 5D 37      .dw aK_L_M_N_O_P_Q_R_S_T
0000:3569 73 37      .dw aU_V_W_X_Y_Z_rub_end
0000:3569 8B 37      .dw aREGI_TIME
0000:3569 00 61      .dw 6100h
0000:3569 22 61      .dw 6122h
0000:3569 44 61      .dw 6144h
0000:3569 66 61      .dw 6166h
0000:3569 88 61      .dw 6188h
0000:3569 9E 37      .dw aRANK_SCORE_NAME
0000:3569 B6 37      .dw aYOUR_NAME_WAS_REGISTERED
0000:3569 D2 37      .dw aINSERT_COIN

```

```

0000:3683 E1 37 .dw aPLAYER_COIN
0000:3685 1D 00 .dw 1Dh
0000:3687 00 3F .dw byte_0_3EFF+1
0000:3689 09 3F .dw byte_0_3F09
0000:368B 96 76 17 11+aGAME_OVER: .db 96h, 76h, 17h, 11h, 1Dh, 15h, 10h, 10h, 1Fh, 26h, 15h
0000:368B 1D 15 10 10+ ; DATA XREF: 0000:364B|o
0000:368B 1F 26 15 22+ .db 22h, 3Fh
0000:3698 94 76 20 1C+aPLAYER: .db 94h, 76h, 20h, 1Ch, 11h, 29h, 15h, 22h, 10h, 30h, 32h
0000:3698 11 29 15 22+ ; DATA XREF: 0000:364F|o
0000:3698 10 30 32 31+ .db 31h, 3Fh
0000:36A5 94 76 20 1C+aPLAYER_: .db 94h, 76h, 20h, 1Ch, 11h, 29h, 15h, 22h, 10h, 30h, 33h
0000:36A5 11 29 15 22+ ; DATA XREF: 0000:3651|o
0000:36A5 10 30 33 31+ .db 31h, 3Fh
0000:36B2 80 76 18 19+aHIGH_SCORE: .db 80h, 76h, 18h, 19h, 17h, 18h, 10h, 23h, 13h, 1Fh, 22h
0000:36B2 17 18 10 23+ ; DATA XREF: 0000:3653|o
0000:36B2 13 1F 22 15+ .db 15h, 3Fh
0000:36BF 9F 75 13 22+aCREDIT: .db 9Fh, 75h, 13h, 22h, 15h, 14h, 19h, 24h, 10h, 10h, 10h
0000:36BF 15 14 19 24+ ; DATA XREF: 0000:3655|o
0000:36BF 10 10 10 10+ .db 10h, 3Fh
0000:36CC 5E 77 18 1F+aHOW_HIGH_CAN_YOU_GET: .db 5Eh, 77h, 18h, 1Fh, 27h, 10h, 18h, 19h, 17h, 18h, 10h
0000:36CC 27 10 18 19+ ; DATA XREF: 0000:3659|o
0000:36CC 17 18 10 13+ .db 13h, 11h, 1Eh, 10h, 29h, 1Fh, 25h, 10h, 17h, 15h, 24h
0000:36CC 11 1E 10 29+ .db 10h, 0Fh, 10h, 3Fh
0000:36E6 29 77 1F 1E+aONLY_1_PLAYER_BUTTON: .db 29h, 77h, 1Fh, 1Eh, 1Ch, 29h, 10h, 1, 10h, 20h, 1Ch
0000:36E6 1C 29 10 01+ ; DATA XREF: 0000:365D|o
0000:36E6 10 20 1C 11+ .db 11h, 29h, 15h, 22h, 10h, 12h, 25h, 24h, 24h, 1Fh, 1Eh
0000:36E6 29 15 22 10+ .db 3Fh
0000:36FD 29 77 01 10+a1_OR_2_PLAYERS: .db 29h, 77h, 1, 10h, 1Fh, 22h, 10h, 2, 10h, 20h, 1Ch
0000:36FD 1F 22 10 02+ ; DATA XREF: 0000:365F|o
0000:36FD 10 20 1C 11+ .db 11h, 29h, 15h, 22h, 23h, 10h, 12h, 25h, 24h, 24h, 1Fh
0000:36FD 29 15 22 23+ .db 1Eh, 3Fh
0000:3715 27 76 20 25+aPUSH: .db 27h, 76h, 20h, 25h, 23h, 18h, 3Fh
0000:3715 27 76 20 25+aPUSH: ; DATA XREF: 0000:3663|o
0000:371C 06 77 1E 11+aNAME_REGISTRATION: .db 6, 77h, 1Eh, 11h, 1Dh, 15h, 10h, 22h, 15h, 17h, 19h
0000:371C 1D 15 10 22+ ; DATA XREF: 0000:3665|o
0000:371C 15 17 19 23+ .db 23h, 24h, 22h, 11h, 24h, 19h, 1Fh, 1Eh, 3Fh
0000:3730 88 76 1E 11+aNAME: .db 88h, 76h, 1Eh, 11h, 1Dh, 15h, 2Eh, 3Fh
0000:3730 88 76 1E 11+aNAME: ; DATA XREF: 0000:3667|o
0000:3738 E9 75 2D 2D+a??? : .db 0E9h, 75h, 2Dh, 2Dh, 2Dh, 10h, 10h, 10h, 10h, 10h
0000:3738 2D 10 10 10+ ; DATA XREF: 0000:3669|o
0000:3738 10 10 10 10+ .db 10h, 10h, 10h, 10h, 3Fh
0000:3747 08 77 11 10+aA_B_C_D_E_F_G_H_I_J: .db 08h, 77h, 11h, 10h, 12h, 10h, 13h, 10h, 14h, 10h, 15h
0000:3747 12 10 13 10+ ; DATA XREF: 0000:366B|o
0000:3747 14 10 15 10+ .db 10h, 16h, 10h, 17h, 10h, 18h, 10h, 19h, 10h, 1Ah, 3Fh
0000:375D 0D 77 aK_L_M_N_O_P_Q_R_S_T: .dw VRAM_start+30Dh
0000:375F 1B 10 1C 10+ .db 1Bh, 10h, 1Ch, 10h, 1Dh, 10h, 1Eh, 10h, 1Fh, 10h, 20h
0000:375F 1D 10 1E 10+ .db 10h, 21h, 10h, 22h, 10h, 23h, 10h, 24h, 3Fh
0000:3773 0F 77 aU_V_W_X_Y_Z_rub_end: .dw VRAM_start+30Fh
0000:3775 25 10 26 10+ .db 25h, 10h, 26h, 10h, 27h, 10h, 28h, 10h, 29h, 10h, 2Ah
0000:3775 27 10 28 10+ .db 10h, 2Bh, 10h, 2Ch, 44h, 45h, 46h, 47h, 48h, 10h, 3Fh
0000:378B F2 76 aREGI_TIME: .dw VRAM_start+2F2h
0000:378D 22 15 17 19+ .db 22h, 15h, 17h, 19h, 10h, 24h, 19h, 1Dh, 15h, 10h, 10h
0000:378D 10 24 19 1D+ .db 30h, 3, 0, 31h, 10h, 3Fh
0000:379E 92 77 aRANK_SCORE_NAME: .dw VRAM_start+392h
0000:379E 92 77 aRANK_SCORE_NAME: ; DATA XREF: 0000:367D|o
0000:37A0 22 11 1E 1B+ .db 22h, 11h, 1Eh, 1Bh, 10h, 10h, 23h, 13h, 1Fh, 22h, 15h
0000:37A0 10 23 13+ .db 10h, 10h, 1Eh, 11h, 1Dh, 15h, 10h, 10h, 10h, 10h, 3Fh
0000:37B6 72 77 aYOUR_NAME_WAS_REGISTERED: .dw VRAM_start+372h
0000:37B6 72 77 aYOUR_NAME_WAS_REGISTERED: ; DATA XREF: 0000:367F|o
0000:37B8 29 1F 25 22+ .db 29h, 1Fh, 25h, 22h, 10h, 1Eh, 11h, 1Dh, 15h, 10h, 27h
0000:37B8 10 1E 11 1D+ .db 11h, 23h, 10h, 22h, 15h, 17h, 19h, 23h, 24h, 15h, 22h
0000:37B8 15 10 27 11+ .db 15h, 14h, 42h, 3Fh
0000:37D2 A7 76 aINSERT_COIN: .dw VRAM_start+2A7h
0000:37D4 19 1E 23 15+ .db 19h, 1Eh, 23h, 15h, 22h, 24h, 10h, 13h, 1Fh, 19h, 1Eh
0000:37D4 22 24 10 13+ .db 10h, 3Fh
0000:37E1 0A 77 aPLAYER_COIN: .dw VRAM_start+30Ah
0000:37E1 0A 77 aPLAYER_COIN: ; DATA XREF: 0000:3683|o
0000:37E3 10 10 20 1C+ .db 10h, 10h, 20h, 1Ch, 11h, 29h, 15h, 22h, 10h, 10h, 10h
0000:37E3 11 29 15 22+ .db 10h, 13h, 1Fh, 19h, 1Eh, 3Fh
0000:37F4 FC 76 .dw VRAM_start+2FCCh
0000:37F6 49 4A 10 1E+a_NINTENDO: .db 49h, 4Ah, 10h, 1Eh, 19h, 1Eh, 24h, 15h, 1Eh, 14h, 1Fh
0000:37F6 19 1E 24 15+ .db 10h, 10h, 10h, 10h, 3Fh
0000:3806 7C 75 .dw VRAM_start+17Ch
0000:3808 01 09 08 01+ .db 1, 9, 8, 1, 3Fh
0000:380D 02 97 38 68+draw_data_climb: .db 2, 97h, 38h, 68h, 38h, 2, 0DFh, 54h, 10h, 54h, 2, 0EFh
0000:380D 38 02 DF 54+ ; DATA XREF: 0000:0A92|o
0000:380D 10 54 02 EF+ .db 6Dh, 20h, 6Dh, 2, 0DFh, 8Eh, 10h, 8Eh, 2, 0EFh, 0AFh
0000:380D 6D 20 6D 02+ .db 20h, 0AFh, 2, 0DFh, 0D0h, 10h, 0D0h, 2, 0EFh, 0F1h
0000:380D DF 8E 10 8E+ .db 10h, 0F1h, 0, 53h, 18h, 53h, 54h, 0, 63h, 18h, 63h
0000:380D 02 EF AF 20+ .db 54h, 0, 93h, 38h, 93h, 54h, 0, 83h, 54h, 83h, 0F1h
0000:380D AF 02 DF D0+ .db 0, 93h, 54h, 93h, 0F1h, 0AAh
0000:384A 8D 7D 8C bonus_graphic_tiles: .db 8Dh, 7Dh, 8Ch
0000:384D 6F 00 7C .db 6Fh, 0, 7Ch
0000:3850 6E 00 7C .db 6Eh, 0, 7Ch
0000:3853 6D 00 7C .db 6Dh, 0, 7Ch
0000:3856 6C 00 7C .db 6Ch, 0, 7Ch
0000:3859 8F 7F 8E .db 8Fh, 7Fh, 8Eh
0000:385C 47 27 08 50+kong_normal_spr: .db 47h, 27h, 8, 50h
0000:385C 2F A7 08 50+ ; DATA XREF: animate_kong_and_pauline+74|o
0000:385C 3B 25 08 50+ ; 0000:0B1E|o ...
0000:385C 00 70 08 48+ .db 2Fh, 0A7h, 8, 50h
0000:385C 3B 23 07 40+ .db 3Bh, 25h, 8, 50h
0000:385C 46 A9 08 44+ .db 0, 70h, 8, 48h
0000:385C 00 70 08 48+ .db 3Bh, 23h, 7, 40h
0000:385C 30 29 08 44+ .db 46h, 0A9h, 8, 44h
0000:385C 00 70 08 48+ .db 0, 70h, 8, 48h
0000:385C 00 70 08 48+ .db 30h, 29h, 8, 44h
0000:385C 00 70 0A 48 .db 0, 70h, 8, 48h
0000:385C 00 70 0A 48 .db 0, 70h, 0Ah, 48h
0000:3884 6F 10 09 23+pauline_spr: .db 6Fh, 10h, 9, 23h
0000:3884 6F 11 0A 33 .db 6Fh, 11h, 0Ah, 33h
0000:388C 50 34 08 3C kong_climbing_spr: .db 50h, 34h, 8, 3Ch
0000:388C ; DATA XREF: 0000:0AC0|o
0000:388C ; 0000:168B|o ...
0000:3890 00 35 08 3C .db 0, 35h, 8, 3Ch
0000:3894 53 32 08 40 .db 53h, 32h, 8, 40h
0000:3898 63 33 08 40 .db 63h, 33h, 8, 40h
0000:389C 00 70 08 48 .db 0, 70h, 8, 48h
0000:38A0 53 36 08 50 .db 53h, 36h, 8, 50h
0000:38A4 63 37 08 50 .db 63h, 37h, 8, 50h
0000:38A8 6B 31 08 41 .db 6Bh, 31h, 8, 41h
0000:38AC 00 70 08 48 .db 0, 70h, 8, 48h
0000:38B0 6A 14 0A 48 .db 6Ah, 14h, 0Ah, 48h
0000:38B4 FD FD FD FD+dk_intro_jump_up_data: .db 0FDh, 0FDh, 0FDh, 0FDh, 0FDh, 0FDh, 0FEh, 0FEh
0000:38B4 FD FD FE+ ; DATA XREF: 0000:0AA9|o
0000:38B4 FE FE FE+ .db 0FEh, 0FEh, 0FEh, 0FEh, 0FFh, 0FFh, 0FFh, 0
0000:38B4 FE FF FF+ .db 0, 1, 1, 1, 7Fh
0000:38CB FF FF FF+dk_intro_jump_across_data: .db 0FFh, 0FFh, 0FFh, 0FFh, 0FFh, 0, 0FFh, 0, 0, 1, 0
0000:38CB FF 00 FF 00+ ; DATA XREF: 0000:0AAF|o
0000:38CB 00 01 00 01+ ; 0000:0B86|o
0000:38CB 01 01 01 01+ .db 1, 1, 1, 1, 1, 7Fh
0000:38DC 04 7F F0 10+draw_data_bend_girders_2: .db 4, 7Fh, 0F0h, 10h, 0F0h, 2, 0DFh, 0F2h, 70h, 0F8h
0000:38DC F0 02 DF F2+ ; DATA XREF: 0000:0B91|o

```

```

0000:38DC 70 F8 02 6F+ .db 2, 6Fh, 0F8h, 10h, 0F8h, 0AAh, 4, 0DFh, 0D0h, 90h
0000:38DC F8 10 F8 AA+ .db 0D0h, 2, 0DFh, 0DCh, 20h, 0D1h, 0AAh, 0FFh, 0FFh, 0FFh
0000:38DC 04 DF D0 90+ .db 0FFh, 0FFh, 4, 0DFh, 0A8h, 20h, 0A8h, 4, 5Fh, 0B0h
0000:38DC D0 02 DF DC+ .db 20h, 0B0h, 2, 0DFh, 0B0h, 20h, 0BBh, 0AAh, 4, 0DFh
0000:38DC 20 D1 AA FF+ .db 88h, 30h, 88h, 4, 0DFh, 90h, 0B0h, 90h, 2, 0DFh, 9Ah
0000:38DC FF FF FF FF+ .db 20h, 8Fh, 0AAh, 4, 0BFh, 68h, 20h, 68h, 4, 3Fh, 70h
0000:38DC 04 DF A8 20+ .db 20h, 70h, 2, 0DFh, 6Eh, 20h, 79h, 0AAh
0000:392C 02 DF 58 A0+draw_data_bend_girders_1: .db 2, 0DFh, 58h, 0A0h, 55h, 0AAh ; DATA XREF: 0000:0B48|o
0000:3932 00 70 08 44+dk_throw_barrel_spr: .db 0, 70h, 8, 44h, 2Bh, 0ACh, 8, 4Ch, 3Bh, 0AEh, 8, 4Ch ; DATA XREF: 0000:1671|o
0000:3932 2B AC 08 4C+ ; DATA XREF: 0000:1671|o
0000:3932 3B AE 08 4C+ ; sub_0_2C8F+95|o
0000:3932 3B AF 08 3C+ .db 3Bh, 0AFh, 8, 3Ch, 4Bh, 0B0h, 7, 3Ch, 4Bh, 0ADh, 8
0000:3932 4B B0 07 3C+ .db 4Ch, 0, 70h, 8, 44h, 0, 70h, 8, 44h, 0, 70h, 8, 44h
0000:3932 4B AD 08 4C+ .db 0, 70h, 0Ah, 44h, 47h, 27h, 8, 4Ch, 2Fh, 0A7h, 8, 4Ch
0000:3932 00 70 08 44+ .db 3Bh, 25h, 8, 4Ch, 0, 70h, 8, 44h, 3Bh, 23h, 7, 3Ch
0000:3932 00 70 08 44+ .db 4Bh, 2Ah, 8, 3Ch, 4Bh, 2Bh, 8, 4Ch, 2Bh, 0AAh, 8, 3Ch
0000:3932 00 70 08 44+ .db 2Bh, 0ABh, 8, 4Ch, 0, 70h, 0Ah, 44h, 0, 70h, 8, 44h
0000:3932 00 70 0A 44+ .db 4Bh, 2Ch, 8, 4Ch, 3Bh, 2Eh, 8, 4Ch, 3Bh, 2Fh, 8, 3Ch
0000:3932 47 27 08 4C+ .db 2Bh, 30h, 7, 3Ch, 2Bh, 2Dh, 8, 4Ch, 0, 70h, 8, 44h
0000:3932 2F A7 08 4C+ .db 0, 70h, 8, 44h, 0, 70h, 8, 44h, 0, 70h, 0Ah, 44h
0000:39AA FD FD FE+ bouncing_spring_data: .db 0FDh, 0FDh, 0FDh, 0FEh, 0FEh, 0FEh, 0FEh, 0FEh, 0FEh, 0FEh ; DATA XREF: sub_0_2E04+98|o
0000:39AA FE FE FE FF+ ; DATA XREF: sub_0_2E04+C4|o
0000:39AA FF 00 FF 00+ ; DATA XREF: sub_0_2E04+C4|o
0000:39AA 00 01 00 01+ .db 0, 0FFh, 0, 0, 1, 0, 1, 1, 2, 2, 2, 2, 3, 3, 3, 7Fh
0000:39C3 1E 4E BB 4C+ barrel_rolling_data: .db 1Eh, 4Eh, 0BBh, 4Ch, 0D8h, 4Eh, 59h, 4Eh, 7Fh
0000:39C3 D8 4E 59 4E+ ; DATA XREF: sub_0_2C8F+FD|o
0000:39CC BB 4D 7F barrel_falling_data: .db 0BBh, 4Dh, 7Fh ; DATA XREF: sub_0_2C8F+F4|o
0000:39CF 47 27 08 50 dk_thrash_right_spr: .db 47h, 27h, 8, 50h ; DATA XREF: animate_kong_and_pauline+43|o
0000:39CF ; 0000:0816|o
0000:39D3 2D 26 08 50 .db 2Dh, 26h, 8, 50h
0000:39D7 3B 25 08 50 .db 3Bh, 25h, 8, 50h
0000:39DB 00 70 08 48 .db 0, 70h, 8, 48h
0000:39DF 3B 24 07 40 .db 3Bh, 24h, 7, 40h
0000:39E3 4B 28 08 40 .db 4Bh, 28h, 8, 40h
0000:39E7 00 70 08 48 .db 0, 70h, 8, 48h
0000:39EB 30 29 08 44 .db 30h, 29h, 8, 44h
0000:39EF 00 70 08 48 .db 0, 70h, 8, 48h
0000:39F3 00 70 0A 48 .db 0, 70h, 0Ah, 48h
0000:39F7 49 A6 08 50 dk_thrash_left_spr: .db 49h, 0A6h, 8, 50h ; DATA XREF: animate_kong_and_pauline+4A|o
0000:39FB 2F A7 08 50 .db 2Fh, 0A7h, 8, 50h
0000:39FF 3B 25 08 50 .db 3Bh, 25h, 8, 50h
0000:3A03 00 70 08 48 .db 0, 70h, 8, 48h
0000:3A07 3B 24 07 40 .db 3Bh, 24h, 7, 40h
0000:3A0B 46 A9 08 44 .db 46h, 0A9h, 8, 44h
0000:3A0F 00 70 08 48 .db 0, 70h, 8, 48h
0000:3A13 2B A8 08 40 .db 2Bh, 0A8h, 8, 40h
0000:3A17 00 70 08 48 .db 0, 70h, 8, 48h
0000:3A1B 00 70 0A 48 .db 0, 70h, 0Ah, 48h
0000:3A1F 73 A7 88 60 fk_falling_spr: .db 73h, 0A7h, 88h, 60h ; DATA XREF: 0000:1870|o
0000:3A23 8B 27 88 60 .db 8Bh, 27h, 88h, 60h
0000:3A27 7F 25 88 60 .db 7Fh, 25h, 88h, 60h
0000:3A2B 00 70 88 68 .db 0, 70h, 88h, 68h
0000:3A2F 7F 24 87 70 .db 7Fh, 24h, 87h, 70h
0000:3A33 74 29 88 6C .db 74h, 29h, 88h, 6Ch
0000:3A37 00 70 88 68 .db 0, 70h, 88h, 68h
0000:3A3B 8A A9 88 6C .db 8Ah, 0A9h, 88h, 6Ch
0000:3A3F 00 70 88 68 .db 0, 70h, 88h, 68h
0000:3A43 00 70 8A 68 .db 0, 70h, 8Ah, 68h
0000:3A47 05 AF F0 50+draw_data_rivet_end1: .db 5, 0AFh, 0F0h, 50h, 0F0h, 0AAh ; DATA XREF: 0000:17D9|o
0000:3A4D 05 AF E8 50+draw_data_rivet_end2: .db 5, 0AFh, 0E8h, 50h, 0E8h, 0AAh ; DATA XREF: 0000:17E5|o
0000:3A53 05 AF E0 50+draw_data_rivet_end3: .db 5, 0AFh, 0E0h, 50h, 0E0h, 0AAh ; DATA XREF: 0000:17F1|o
0000:3A59 05 AF D8 50+draw_data_rivet_end4: .db 5, 0AFh, 0D8h, 50h, 0D8h, 0AAh ; DATA XREF: 0000:17FD|o
0000:3A5F 05 B7 58 48+draw_data_rivet_end5: .db 5, 0B7h, 58h, 48h, 58h, 0AAh ; DATA XREF: 0000:18A5|o
0000:3A65 01 04 01 03+level_seq1: .db 1, 4, 1, 3, 4, 1, 2, 3, 4, 1, 2, 1, 3, 4 ; DATA XREF: 0000:095F|o
0000:3A73 01 02 01 03+level_seq2: .db 1, 2, 1, 3, 1, 4, 7Fh ; DATA XREF: 0000:1799|o
0000:3A73 01 04 7F ; 0000:1947|o
0000:3A7A FF 00 FF FF+fireball_bouncing_data: .db 0FFh, 0, 0FFh, 0FFh, 0FEh, 0FEh, 0FEh, 0FEh, 0FEh, 0FEh ; DATA XREF: sub_0_3202+65|o
0000:3A7A FE FE FE FE+ ; DATA XREF: sub_0_3202+65|o
0000:3A7A FE FE FE FE+ .db 0FEh, 0FEh, 0FEh, 0FEh, 0FEh, 0FEh, 0FFh, 0FFh, 0
0000:3A8C E8 E5 E3 E2+fireball_bounce_data: .db 0E8h, 0E5h, 0E3h, 0E2h, 0E1h, 0E0h, 0DFh, 0DEh, 0DDh ; DATA XREF: sub_0_342C+F|o
0000:3A8C E1 00 DF DE+ ; DATA XREF: sub_0_342C+F|o
0000:3A8C DD DD DC DC+ .db 0DDh, 0DCh, 0DCh, 0DCh, 0DCh, 0DCh, 0DDh, 0DDh
0000:3A8C DC DC DC DC+ .db 0DEh, 0DFh, 0E0h, 0E1h, 0E2h, 0E3h, 0E4h, 0E5h, 0E7h
0000:3A8C DD DD DE DF+ .db 0E9h, 0EBh, 0EDh, 0F0h, 0AAh
0000:3AAC 80 7B 78 76+cement_fireball_data: .db 80h, 7Bh, 78h, 76h, 74h, 73h, 72h, 71h, 70h, 70h, 6Fh ; DATA XREF: sub_0_3478+F|o
0000:3AAC 74 73 72 71+ ; DATA XREF: sub_0_3478+F|o
0000:3AAC 70 70 6F 6F+ .db 6Fh, 6Fh, 70h, 70h, 71h, 72h, 73h, 74h, 75h, 76h, 77h
0000:3AAC 6F 70 70 71+ .db 78h, 0AAh
0000:3AC4 EE F0 DB A0+rivet_fireball_data: .db 0EEh, 0F0h, 0DBh, 0A0h, 0E6h, 0C8h, 0D6h, 78h, 0EBh ; DATA XREF: sub_0_34B9+E|o
0000:3AC4 E6 C8 D6 78+ ; DATA XREF: sub_0_34B9+E|o
0000:3AC4 EB F0 DB A0+ .db 0F0h, 0DBh, 0A0h, 0E6h, 0C8h, 0E6h, 0C8h
0000:3AD4 1B C8 23 A0+rivet_fireball_start_points: .db 1Bh, 0C8h, 23h, 0A0h, 2Bh, 78h, 12h, 0F0h, 1Bh, 0C8h ; DATA XREF: sub_0_34B9+34|o
0000:3AD4 2B 78 12 F0+ ; DATA XREF: sub_0_34B9+34|o
0000:3AD4 1B C8 23 A0+ .db 23h, 0A0h, 12h, 0F0h, 1Bh, 0C8h
0000:3AE4 02 97 38 68+barrel_level_tilemap_data: .db 2, 97h, 38h, 68h, 38h, 2, 9Fh, 54h, 10h, 54h, 2, 0DFh ; DATA XREF: 0000:0CD4|o
0000:3AE4 38 02 9F 54+ ; DATA XREF: 0000:0CD4|o
0000:3AE4 10 54 02 DF+ ; sub_0_2441+19|o
0000:3AE4 58 A0 55 02+ .db 58h, 0A0h, 55h, 2, 0EFh, 6Dh, 20h, 79h, 2, 0DFh, 9Ah
0000:3AE4 EF 6D 20 79+ .db 10h, 8Eh, 2, 0EFh, 0AFh, 20h, 0BBh, 2, 0DFh, 0DCh
0000:3AE4 02 DF 9A 10+ .db 10h, 0D0h, 2, 0FFh, 0F0h, 80h, 0F7h, 2, 7Fh, 0F8h
0000:3AE4 8E 02 EF AF+ .db 0, 0F8h, 0, 0CBh, 57h, 0CBh, 6Fh, 0, 0CBh, 99h, 0CBh
0000:3AE4 20 BB 02 DF+ .db 0B1h, 0, 0CBh, 0DBh, 0CBh, 0F3h, 0, 63h, 18h, 63h
0000:3AE4 DC 10 D0 02+ .db 54h, 1, 63h, 0D5h, 63h, 0F8h, 0, 33h, 78h, 33h, 90h
0000:3AE4 FF F0 80 F7+ .db 0, 33h, 0BAh, 33h, 0D2h, 0, 53h, 18h, 53h, 54h, 1
0000:3AE4 02 7F FB 00+ .db 53h, 92h, 53h, 0B8h, 0, 5Bh, 76h, 5Bh, 92h, 0, 73h
0000:3AE4 F8 00 CB 57+ .db 0B6h, 73h, 0D6h, 0, 83h, 95h, 83h, 0B5h, 0, 93h, 38h
0000:3AE4 CB 6F 00 CB+ .db 93h, 54h, 1, 0BBh, 70h, 0BBh, 98h, 1, 6Bh, 54h, 6Bh
0000:3AE4 99 CB B1 00+ .db 75h, 0AAh
0000:3B5D 06 8F 90 70+cement_pie_level_tilemap_data: .db 6, 8Fh, 90h, 70h, 90h, 6, 8Fh, 98h, 70h, 98h, 6, 8Fh ; DATA XREF: 0000:0CDF|o
0000:3B5D 90 06 8F 98+ ; DATA XREF: 0000:0CDF|o
0000:3B5D 70 98 06 8F+ ; sub_0_2441+20|o
0000:3B5D A0 70 A0 00+ .db 0A0h, 70h, 0A0h, 0, 63h, 18h, 63h, 58h, 0, 63h, 80h
0000:3B5D 63 18 63 58+ .db 63h, 0A8h, 0, 63h, 0D0h, 63h, 0F8h, 0, 53h, 18h, 53h
0000:3B5D 00 63 80 63+ .db 58h, 0, 53h, 0A8h, 53h, 0D0h, 0, 9Bh, 80h, 9Bh, 0A8h
0000:3B5D A8 00 63 D0+ .db 0, 9Bh, 0D0h, 9Bh, 0F8h, 1, 23h, 58h, 23h, 80h, 1
0000:3B5D 63 F8 00 53+ .db 0DBh, 58h, 0DBh, 80h, 0, 2Bh, 80h, 2Bh, 0A8h, 0, 0D3h
0000:3B5D 18 53 58 00+ .db 80h, 0D3h, 0A8h, 0, 0A3h, 0A8h, 0A3h, 0D0h, 0, 2Bh
0000:3B5D 53 A8 53 D0+ .db 0D0h, 2Bh, 0F8h, 0, 0D3h, 0D0h, 0D3h, 0F8h, 0, 93h
0000:3B5D 00 9B 00 9B+ .db 38h, 93h, 58h, 2, 97h, 38h, 68h, 38h, 3, 0EFh, 58h
0000:3B5D A8 00 9B D0+ .db 10h, 58h, 3, 0F7h, 80h, 88h, 80h, 3, 77h, 80h, 8, 80h
0000:3B5D 9B F8 01 23+ .db 2, 0A7h, 0A8h, 50h, 0A8h, 2, 0E7h, 0A8h, 0B8h, 0A8h
0000:3B5D 58 23 80 01+ .db 2, 3Fh, 0A8h, 18h, 0A8h, 3, 0EFh, 0D0h, 10h, 0D0h
0000:3B5D DB 58 DB 80+ .db 2, 0EFh, 0F8h, 10h, 0F8h, 0AAh
0000:3BE5 00 63 18 63+elevator_level_tilemap_data: .db 0, 63h, 18h, 63h, 58h, 0, 63h, 88h, 63h, 0D0h, 0, 53h ; DATA XREF: 0000:0CFA|o
0000:3BE5 58 00 63 88+ ; DATA XREF: 0000:0CFA|o
0000:3BE5 63 D0 00 53+ ; sub_0_2441+27|o

```

```
0000:3BE5 18 53 58 00+ .db 18h, 53h, 58h, 0, 53h, 88h, 53h, 0D0h, 0, 0E3h, 68h
0000:3BE5 53 88 53 D0+ .db 0E3h, 90h, 0, 0E3h, 0B8h, 0E3h, 0D0h, 0, 0CBh, 90h
0000:3BE5 00 E3 68 E3+ .db 0CBh, 0B0h, 0, 0B3h, 58h, 0B3h, 78h, 0, 9Bh, 80h, 9Bh
0000:3BE5 90 00 E3 B8+ .db 0A0h, 0, 93h, 38h, 93h, 58h, 0, 23h, 88h, 23h, 0C0h
0000:3BE5 E3 D0 00 CB+ .db 0, 1Bh, 0C0h, 1Bh, 0E8h, 2, 97h, 38h, 68h, 38h, 2
0000:3BE5 90 CB B0 00+ .db 0B7h, 58h, 10h, 58h, 2, 0EFh, 68h, 0E0h, 68h, 2, 0D7h
0000:3BE5 B3 58 B3 78+ .db 70h, 0C8h, 70h, 2, 0BFh, 78h, 0B0h, 78h, 2, 0A7h, 80h
0000:3BE5 00 9B 80 9B+ .db 90h, 80h, 2, 67h, 88h, 48h, 88h, 2, 27h, 88h, 10h
0000:3BE5 A0 00 93 38+ .db 88h, 2, 0EFh, 90h, 0C8h, 90h, 2, 0A7h, 0A0h, 98h, 0A0h
0000:3BE5 93 58 00 23+ .db 2, 0BFh, 0A8h, 0B0h, 0A8h, 2, 0D7h, 0B0h, 0C8h, 0B0h
0000:3BE5 88 23 C0 00+ .db 2, 0EFh, 0B8h, 0E0h, 0B8h, 2, 27h, 0C0h, 10h, 0C0h
0000:3BE5 1B C0 1B E8+ .db 2, 0EFh, 0D0h, 0D8h, 0D0h, 2, 67h, 0D0h, 50h, 0D0h
0000:3BE5 02 97 38 68+ .db 2, 0CFh, 0D8h, 0C0h, 0D8h, 2, 0B7h, 0E0h, 0A8h, 0E0h
0000:3BE5 38 02 B7 58+ .db 2, 9Fh, 0E8h, 88h, 0E8h, 2, 27h, 0E8h, 10h, 0E8h, 2
0000:3BE5 10 58 02 EF+ .db 0EFh, 0F8h, 10h, 0F8h, 0AAh
0000:3C8B 00 7B 80 7B+rivet_level_tilemap_data:.db 0, 7Bh, 80h, 7Bh, 0A8h, 0, 7Bh, 0D0h, 7Bh, 0F8h, 0
0000:3C8B A8 00 7B D0+ ; DATA XREF: 0000:0CC3!o
0000:3C8B 7B F8 00 33+ ; sub_0_2441+2D!o
0000:3C8B 58 33 80 00+ .db 33h, 58h, 33h, 80h, 0, 53h, 58h, 53h, 80h, 0, 0ABh
0000:3C8B 53 58 53 80+ .db 58h, 0ABh, 80h, 0, 0CBh, 58h, 0CBh, 80h, 0, 2Bh, 80h
0000:3C8B 00 AB 58 AB+ .db 2Bh, 0A8h, 0, 0D3h, 80h, 0D3h, 0A8h, 0, 23h, 0A8h
0000:3C8B 80 00 CB 58+ .db 23h, 0D0h, 0, 5Bh, 0A8h, 5Bh, 0D0h, 0, 0A3h, 0A8h
0000:3C8B CB 80 00 2B+ .db 0A3h, 0D0h, 0, 0DBh, 0A8h, 0DBh, 0D0h, 0, 1Bh, 0D0h
0000:3C8B 80 2B A8 00+ .db 1Bh, 0F8h, 0, 0E3h, 0D0h, 0E3h, 0F8h, 5, 0B7h, 30h
0000:3C8B D3 80 D3 A8+ .db 48h, 30h, 5, 0CFh, 58h, 30h, 58h, 5, 0D7h, 80h, 28h
0000:3C8B 00 23 A8 23+ .db 80h, 5, 0DFh, 0A8h, 20h, 0A8h, 5, 0E7h, 0D0h, 18h
0000:3C8B D0 00 5B A8+ .db 0D0h, 5, 0EFh, 0F8h, 10h, 0F8h, 0AAh
0000:3CF0 10 82 85 8B+draw_data_how_high:.db 10h, 82h, 85h, 8Bh, 10h, 85h, 80h, 8Bh, 10h, 87h, 85h
0000:3CF0 10 85 80 8B+ ; DATA XREF: 0000:0C50!o
0000:3CF0 10 87 85 8B+ .db 8Bh, 81h, 80h, 80h, 8Bh, 81h, 82h, 85h, 8Bh, 81h, 85h
0000:3CF0 81 80 80 8B+ .db 80h, 8Bh
0000:3D08 05 title_screen: .db 5 ; DATA XREF: 0000:07F7!o
0000:3D08 ; RLE-encoded "DONKEY KONG" title
0000:3D09 88 77 .dw VRAM_start+388h
0000:3D0B 01 .db 1
0000:3D0C 68 77 .dw VRAM_start+368h
0000:3D0E 01 .db 1
0000:3D0F 6C 77 .dw VRAM_start+36Ch
0000:3D11 03 .db 3
0000:3D12 49 77 .dw VRAM_start+349h
0000:3D14 05 .db 5
0000:3D15 08 77 .dw VRAM_start+308h
0000:3D17 01 .db 1
0000:3D18 E8 76 .dw VRAM_start+2E8h
0000:3D1A 01 .db 1
0000:3D1B EC 76 .dw VRAM_start+2ECh
0000:3D1D 05 .db 5
0000:3D1E C8 76 .dw VRAM_start+2C8h
0000:3D20 05 .db 5
0000:3D21 88 76 .dw VRAM_start+288h
0000:3D23 02 .db 2
0000:3D24 69 76 .dw VRAM_start+269h
0000:3D26 02 .db 2
0000:3D27 4A 76 .dw VRAM_start+24Ah
0000:3D29 05 .db 5
0000:3D2A 28 76 .dw VRAM_start+228h
0000:3D2C 05 .db 5
0000:3D2D E8 75 .dw VRAM_start+1E8h
0000:3D2F 01 .db 1
0000:3D30 CA 75 .dw VRAM_start+1CAh
0000:3D32 03 .db 3
0000:3D33 A9 75 .dw VRAM_start+1A9h
0000:3D35 01 .db 1
0000:3D36 88 75 .dw VRAM_start+188h
0000:3D38 01 .db 1
0000:3D39 8C 75 .dw VRAM_start+18Ch
0000:3D3B 05 .db 5
0000:3D3C 48 75 .dw VRAM_start+148h
0000:3D3E 01 .db 1
0000:3D3F 28 75 .dw VRAM_start+128h
0000:3D41 01 .db 1
0000:3D42 2A 75 .dw VRAM_start+12Ah
0000:3D44 01 .db 1
0000:3D45 2C 75 .dw VRAM_start+12Ch
0000:3D47 01 .db 1
0000:3D48 08 75 .dw VRAM_start+108h
0000:3D4A 01 .db 1
0000:3D4B 0A 75 .dw VRAM_start+10Ah
0000:3D4D 01 .db 1
0000:3D4E 0C 75 .dw VRAM_start+10Ch
0000:3D50 03 .db 3
0000:3D51 C8 74 .dw VRAM_start+0C8h
0000:3D53 03 .db 3
0000:3D54 AA 74 .dw VRAM_start+0AAh
0000:3D56 03 .db 3
0000:3D57 88 74 .dw VRAM_start+88h
0000:3D59 05 .db 5
0000:3D5A 2F 77 .dw VRAM_start+32Fh
0000:3D5C 05 .db 5
0000:3D5D 0F 77 .dw VRAM_start+30Fh
0000:3D5F 02 .db 2
0000:3D60 F0 76 .dw VRAM_start+2F0h
0000:3D62 02 .db 2
0000:3D63 CF 76 .dw VRAM_start+2CFh
0000:3D65 02 .db 2
0000:3D66 D2 76 .dw VRAM_start+2D2h
0000:3D68 05 .db 5
0000:3D69 8F 76 .dw VRAM_start+28Fh
0000:3D6B 05 .db 5
0000:3D6C 6F 76 .dw VRAM_start+26Fh
0000:3D6E 01 .db 1
0000:3D6F 4F 76 .dw VRAM_start+24Fh
0000:3D71 01 .db 1
0000:3D72 53 76 .dw VRAM_start+253h
0000:3D74 05 .db 5
0000:3D75 2F 76 .dw VRAM_start+22Fh
0000:3D77 05 .db 5
0000:3D78 EF 75 .dw VRAM_start+1EFh
0000:3D7A 02 .db 2
0000:3D7B D0 75 .dw VRAM_start+1D0h
0000:3D7D 02 .db 2
0000:3D7E B1 75 .dw VRAM_start+1B1h
0000:3D80 05 .db 5
0000:3D81 8F 75 .dw VRAM_start+18Fh
0000:3D83 03 .db 3
0000:3D84 50 75 .dw VRAM_start+150h
```

```

0000:3D86 05 .db 5
0000:3D87 2F 75 .dw VRAM_start+12Fh
0000:3D89 01 .db 1
0000:3D8A 0F 75 .dw VRAM_start+10Fh
0000:3D8C 01 .db 1
0000:3D8D 13 75 .dw VRAM_start+113h
0000:3D8F 01 .db 1
0000:3D90 EF 74 .dw VRAM_start+0EFh
0000:3D92 01 .db 1
0000:3D93 F1 74 .dw VRAM_start+0F1h
0000:3D95 01 .db 1
0000:3D96 F3 74 .dw VRAM_start+0F3h
0000:3D98 02 .db 2
0000:3D99 D1 74 .dw VRAM_start+0D1h
0000:3D9B 00 .db 0
0000:3D9C 00 00 23 68+level_init_data: .db 0, 0, 23h, 68h, 1, 11h, 0, 0, 0, 10h, 0DBh, 68h, 1
0000:3D9C 01 11 00 00+ ; DATA XREF: 0000:0F6F[o
0000:3D9C 00 10 DB 68+ .db 40h, 0, 0, 8, 1, 1, 1, 1, 1, 1, 1, 1, 0, 0, 0, 0
0000:3D9C 01 40 00 00+ .db 0, 0, 80h, 1, 0C0h, 0FFh, 1, 0FFh, 0FFh, 34h, 0C3h
0000:3D9C 08 01 01 01+ .db 39h, 0, 67h, 80h, 69h, 1Ah, 1, 0, 0, 0, 0, 0, 0, 0
0000:3D9C 01 01 01 01+ .db 0, 4, 0, 10h, 0, 0, 0, 0
0000:3DDC 1E 18 0B 4B+top_barrel_spr: .db 1Eh, 18h, 0Bh, 4Bh, 14h, 18h, 0Bh, 4Bh, 1Eh, 18h, 0Bh
0000:3DDC 14 18 0B 4B+ ; DATA XREF: 0000:0FD7[o
0000:3DDC 1E 18 0B 3B+ .db 3Bh, 14h, 18h, 0Bh, 3Bh
0000:3DEC 3D 01 03 02 fireball_spr: .db 3Dh, 1, 3, 2 ; DATA XREF: 0000:0FE2[o
0000:3DEC ; 0000:101F[o ...
0000:3DF0 4D 01 04 01 fireball_rivet_spr: .db 4Dh, 1, 4, 1 ; DATA XREF: 0000:1131[o
0000:3DF4 27 70 01 E0+girders_extra_spr: .db 27h, 70h, 1, 0E0h, 0, 0 ; DATA XREF: 0000:0FEF[o
0000:3DFA 7F 40 01 78+cement_extra_spr: .db 7Fh, 40h, 1, 78h, 2, 0 ; DATA XREF: 0000:1049[o
0000:3E00 27 49 0C F0 girder_oil_barrel_spr: .db 27h, 49h, 0Ch, 0F0h ; DATA XREF: 0000:0FF5[o
0000:3E04 7F 49 0C 88 cement_oil_barrel_spr: .db 7Fh, 49h, 0Ch, 88h ; DATA XREF: 0000:104F[o
0000:3E08 1E 07 03 09 hammer_pickup_spr: .db 1Eh, 7, 3, 9 ; DATA XREF: sub_0_11A6+9[o
0000:3E0C 24 64 BB C0 girder_hammer_locs: .db 24h, 64h, 0BBh, 0C0h ; DATA XREF: 0000:1000[o
0000:3E10 23 8D 7B B4 cement_hammer_locs: .db 23h, 8Dh, 7Bh, 0B4h ; DATA XREF: 0000:1070[o
0000:3E14 1B 8C 7C 64 rivet_hammer_locs: .db 1Bh, 8Ch, 7Ch, 64h ; DATA XREF: 0000:113D[o
0000:3E18 4B 0E 04 02 cement_pie_spr: .db 4Bh, 0Eh, 4, 2 ; DATA XREF: 0000:102E[o
0000:3E1C 23 46 03 68+cement_ladder_spr: .db 23h, 46h, 3, 68h, 0DBh, 46h, 3, 68h ; DATA XREF: 0000:105A[o
0000:3E24 17 50 00 5C+cement_conveyer_spr: .db 17h, 50h, 0, 5Ch, 0E7h, 0D0h, 0, 5Ch, 8Ch, 50h, 0 ; DATA XREF: 0000:1065[o
0000:3E24 E7 D0 00 5C+ ; DATA XREF: 0000:1065[o
0000:3E24 8C 50 00 84+ .db 84h, 73h, 0D0h, 0, 84h, 17h, 50h, 0, 0D4h, 0E7h, 0D0h
0000:3E24 73 D0 00 84+ .db 0, 0D4h
0000:3E3C 53 73 0A A0+cement_obj_spr: .db 53h, 73h, 0Ah, 0A0h, 8Bh, 74h, 0Ah, 0F0h, 0DBh, 75h
0000:3E3C 8B 74 0A F0+ ; DATA XREF: 0000:1076[o
0000:3E3C DB 75 0A A0 .db 0Ah, 0A0h
0000:3E48 5B 73 0A C8+elevator_obj_spr: .db 5Bh, 73h, 0Ah, 0C8h, 0E3h, 74h, 0Ah, 60h, 1Bh, 75h
0000:3E48 E3 74 0A 60+ ; DATA XREF: 0000:10DE[o
0000:3E48 1B 75 0A 80 .db 0Ah, 80h
0000:3E54 DB 73 0A C8+rivit_obj_spr: .db 0DBh, 73h, 0Ah, 0C8h, 93h, 74h, 0Ah, 0F0h, 33h, 75h
0000:3E54 93 74 0A F0+ ; DATA XREF: 0000:1143[o
0000:3E54 33 75 0A 50 .db 0Ah, 50h
0000:3E60 44 03 08 04 elevator_spr: .db 44h, 3, 8, 4 ; DATA XREF: 0000:10C3[o
0000:3E64 37 F4 37 C0+elevator_spr_xy: .db 37h, 0F4h, 37h, 0C0h, 37h, 8Ch, 77h, 70h, 77h, 0A4h
0000:3E64 37 8C 77 70+ ; DATA XREF: 0000:10B7[o
0000:3E64 77 A4 77 D8 .db 77h, 0D8h
0000:3E70 ;
0000:3E70
0000:3E70 loc_0_3E70: ld de, 1 ; CODE XREF: sub_0_1DBD+1A[j
0000:3E70 11 01 00
0000:3E73
0000:3E73 loc_0_3E73: ld b, 7Bh ; '{
0000:3E73 06 7B rra
0000:3E75 1F jp nc, loc_0_1E28
0000:3E76 D2 28 1E ld e, 3
0000:3E79 1E 03 ld b, 7Dh ; '}'
0000:3E7B 06 7D rra
0000:3E7D 1F jp nc, loc_0_1E28
0000:3E7E D2 28 1E ld e, 5
0000:3E81 1E 05 ld b, 7Fh ; ' '
0000:3E83 06 7F jp loc_0_1E28
0000:3E85 C3 28 1E
0000:3E88 ;
0000:3E88 SUBROUTINE
0000:3E88
0000:3E88 sub_0_3E88: ; CODE XREF: sub_0_2853+18[p
0000:3E88 3A 27 62 ld a, (level_type)
0000:3E8B E5 push hl
0000:3E8C EF rst 28h ; go!
0000:3E8C ;
0000:3E8D 00 00 .dw 0 ; Jump table
0000:3E8F 99 3E .dw loc_0_3E99
0000:3E91 B0 28 .dw l2_check_hammer_hit
0000:3E93 E0 28 .dw l3_check_hammer_hit
0000:3E95 01 29 .dw l4_check_hammer_hit
0000:3E97 00 00 .dw 0
0000:3E99 ;
0000:3E99 loc_0_3E99: ; DATA XREF: sub_0_3E88+7[o
0000:3E99 E1 pop hl
0000:3E9A AF xor a
0000:3E9B 32 60 60 ld (unk_0_6060), a
0000:3E9E 06 0A ld b, 0Ah
0000:3EA0 11 20 00 ld de, 20h ; ' '
0000:3EA3 DD 21 00 67 ld ix, unk_0_6700
0000:3EA7 CD C3 3E call sub_0_3EC3
0000:3EAA 06 05 ld b, 5
0000:3EAC DD 21 00 64 ld ix, unk_0_6400 ; fireball character data
0000:3EB0 CD C3 3E call sub_0_3EC3
0000:3EB3 3A 60 60 ld a, (unk_0_6060)
0000:3EB6 A7 and a
0000:3EB7 C8 ret z
0000:3EB8 FE 01 cp 1
0000:3EBA C8 ret z
0000:3EBB FE 03 cp 3
0000:3EBD 3E 03 ld a, 3
0000:3EBF D8 ret c
0000:3EC0 3E 07 ld a, 7
0000:3EC2 C9 ret
0000:3EC2 ; End of function sub_0_3E88
0000:3EC2
0000:3EC3 ;
0000:3EC3 SUBROUTINE
0000:3EC3
0000:3EC3 sub_0_3EC3: ; CODE XREF: sub_0_3E88+1F[p
0000:3EC3 DD CB 00 46 ; sub_0_3E88+28[p ...

```

```

0000:3EC3          bit      0, (ix+0)
0000:3EC7 CA FA 3E          jp      z, loc_0_3EFA
0000:3ECA 79              ld      a, c
0000:3ECB DD 96 05          sub     (ix+5)
0000:3ECE D2 D3 3E          jp      nc, loc_0_3ED3
0000:3ED1 ED 44              neg
0000:3ED3
0000:3ED3          loc_0_3ED3:                                     ; CODE XREF: sub_0_3EC3+B↑j
0000:3ED3 3C              inc     a
0000:3ED4 95              sub     l
0000:3ED5 DA DE 3E          jp      c, loc_0_3EDE
0000:3ED8 DD 96 0A          sub     (ix+0Ah)
0000:3EDB D2 FA 3E          jp      nc, loc_0_3EFA
0000:3EDE
0000:3EDE          loc_0_3EDE:                                     ; CODE XREF: sub_0_3EC3+12↑j
0000:3EDE FD 7E 03          ld      a, (iy+3)
0000:3EE1 DD 96 03          sub     (ix+3)
0000:3EE4 D2 E9 3E          jp      nc, loc_0_3EE9
0000:3EE7 ED 44              neg
0000:3EE9
0000:3EE9          loc_0_3EE9:                                     ; CODE XREF: sub_0_3EC3+21↑j
0000:3EE9 94              sub     h
0000:3EEA DA F3 3E          jp      c, loc_0_3EF3
0000:3EED DD 96 09          sub     (ix+9)
0000:3EF0 D2 FA 3E          jp      nc, loc_0_3EFA
0000:3EF3
0000:3EF3          loc_0_3EF3:                                     ; CODE XREF: sub_0_3EC3+27↑j
0000:3EF3 3A 60 60          ld      a, (unk_0_6060)
0000:3EF6 3C              inc     a
0000:3EF7 32 60 60          ld      (unk_0_6060), a
0000:3EFA
0000:3EFA          loc_0_3EFA:                                     ; CODE XREF: sub_0_3EC3+4↑j
0000:3EFA DD 19                                     ; sub_0_3EC3+18↑j ...
0000:3EFA          add     ix, de
0000:3EFC 10 C5          djnz   sub_0_3EC3
0000:3EFE C9              ret
0000:3EFE          ; End of function sub_0_3EC3
0000:3EFE
0000:3EFE          ;
0000:3EFF 00 5C 76 49+byte_0_3EFF: .db 0, 5Ch, 76h, 49h, 4Ah, 1, 9, 8, 1, 3Fh ; DATA XREF: 0000:3687↑o
0000:3F09 7D 77 1E 19+byte_0_3F09: .db 7Dh, 77h, 1Eh, 19h, 1Eh, 24h, 15h, 1Eh, 14h, 1Fh, 10h ; DATA XREF: 0000:3689↑o
0000:3F09 1E 24 15 1E+          ; sub_0_2441↑o
0000:3F09 14 1F 10 1F+          .db 1Fh, 16h, 10h, 11h, 1Dh, 15h, 22h, 19h, 13h, 11h, 10h
0000:3F09 16 10 11 1D+          .db 19h, 1Eh, 13h, 2Bh, 3Fh
0000:3F09 15 22 19 13+
0000:3F24
0000:3F24          ; SUBROUTINE
0000:3F24
0000:3F24          sub_0_3F24:                                     ; CODE XREF: 0000:081C↑p
0000:3F24 21 AF 74          ld      hl, VRAM_start+0AFh
0000:3F27 11 E0 FF          ld      de, 0FFEOh
0000:3F2A 36 9F          ld      (hl), 9Fh ; 'f'
0000:3F2C 19          add     hl, de
0000:3F2D 36 9E          ld      (hl), 9Eh ; 'x'
0000:3F2F C9              ret
0000:3F2F          ; End of function sub_0_3F24
0000:3F2F
0000:3F2F          ;
0000:3F30 50 52 4F 47+ProgramWeWouldTeachYou_Tel_tokyo:.text "PROGRAM,WE WOULD TEACH YOU.*****TEL,TOKYO-JAPAN 044(244)2"
0000:3F30 52 41 4D 2C+          .text "151 EXTENTION 304 SYSTEM DESIGN IKEGAMI CO. LIM."
0000:3FA0
0000:3FA0          loc_0_3FA0:                                     ; CODE XREF: 0000:0CD1↑j
0000:3FA0 CD A6 3F          call    sub_0_3FA6
0000:3FA3 C3 5F 0D          jp      loc_0_D5F
0000:3FA6
0000:3FA6          ; SUBROUTINE
0000:3FA6
0000:3FA6          sub_0_3FA6:                                     ; CODE XREF: 0000:3FA0↑p
0000:3FA6 3E 02          ld      a, 2
0000:3FA8 F7              rst     30h ; return if level bit not set
0000:3FA9 06 02          ld      b, 2
0000:3FAB 21 6C 77          ld      hl, VRAM_start+36Ch
0000:3FAE
0000:3FAE          loc_0_3FAE:                                     ; CODE XREF: sub_0_3FA6+11↑j
0000:3FAE 36 10          ld      (hl), 10h
0000:3FB0 23          inc     hl
0000:3FB1 23          inc     hl
0000:3FB2 36 C0          ld      (hl), 0C0h ; 'L'
0000:3FB4 21 8C 74          ld      hl, VRAM_start+8Ch
0000:3FB7 10 F5          djnz   loc_0_3FAE
0000:3FB9 C9              ret
0000:3FB9          ; End of function sub_0_3FA6
0000:3FB9
0000:3FB9          ;
0000:3FBA 00 00 00 00+          .db 0, 0, 0, 0, 0, 0
0000:3FC0
0000:3FC0          ; SUBROUTINE
0000:3FC0
0000:3FC0          sub_0_3FC0:                                     ; CODE XREF: 0000:2285↑p
0000:3FC0 21 4D 69          ld      hl, soft_sprite_ram+4Dh
0000:3FC3 36 03          ld      (hl), 3
0000:3FC5 2C          inc     l
0000:3FC6 2C          inc     l
0000:3FC7 C9              ret
0000:3FC7          ; End of function sub_0_3FC0
0000:3FC7
0000:3FC7          ;
0000:3FC8 00 00 41 7F+          .db 0, 0, 41h, 7Fh, 7Fh, 41h, 0, 0, 0, 7Fh, 7Fh, 18h, 3Ch
0000:3FC8 7F 41 00 00+          .db 76h, 63h, 41h, 0, 0, 7Fh, 7Fh, 49h, 49h, 49h, 41h
0000:3FC8 00 7F 7F 18+          .db 0, 1Ch, 3Eh, 63h, 41h, 49h, 79h, 79h, 0, 7Ch, 7Eh
0000:3FC8 3C 76 63 41+          .db 13h, 11h, 13h, 7Eh, 7Ch, 0, 7Fh, 7Fh, 0Eh, 1Ch, 0Eh
0000:3FC8 00 00 7F 7F+          .db 7Fh, 7Fh, 0, 0, 41h, 7Fh, 7Fh, 41h, 0, 0
0000:3FC8 49 49 49 41+; end of 'ROM'
0000:3FC8 00 1C 3E 63+
0000:6000
0000:6000
0000:6000          ; Segment type: Regular
0000:6000          ; segment 'RAM'
0000:6000          .org 6000h
0000:6000 ??          RAM_start: .block 1 ; DATA XREF: 0000:0268↑o
0000:6001 ??          no_of_credits: .block 1 ; DATA XREF: display_credits+5↑o
0000:6001          ; 0000:073F↑r ...

```

```

0000:6002 ??                .block 1
0000:6003 ??                .block 1
0000:6004 ??                .block 1 ; DATA XREF: sub_0_17B+5[o
0000:6005 ??                .block 1 ; DATA XREF: 0000:00C6[r
0000:6005 ??                .block 1 ; sub_0_17B+12[r ...
0000:6006 ??                .block 1
0000:6007 ??                .block 1
0000:6008 ??                .block 1
0000:6009 ??                .block 1 ; DATA XREF: return_NOT_16bit_timeout[o
0000:6009*??                .block 1 ; DATA XREF: return_NOT_8bit_timeout[o
0000:6009*??                .block 1 ; 0000:078E[o ...
0000:600A ??                .block 1 ; DATA XREF: 0000:01EE[w
0000:600A ??                .block 1 ; 0000:06FE[r ...
0000:600B ??                .block 1
0000:600C ??                .block 1
0000:600D ??                .block 1
0000:600E ??                .block 1
0000:600F ??                .block 1
0000:6010 ??                .block 1
0000:6010 ??                .block 1 ; DATA XREF: 0000:00AC[w
0000:6011 ??                .block 1 ; 0000:1502[r ...
0000:6012 ??                .block 1
0000:6013 ??                .block 1
0000:6014 ??                .block 1
0000:6015 ??                .block 1
0000:6016 ??                .block 1
0000:6017 ??                .block 1
0000:6018 ?? ??            .block 2 ; DATA XREF: rand[r
0000:6018 ??                .block 2 ; rand+B[w ...
0000:601A ??                .block 1 ; DATA XREF: rand+3[o
0000:601A ??                .block 1 ; 0000:00B5[o ...
0000:601B ??                .block 1
0000:601C ??                .block 1
0000:601D ??                .block 1
0000:601E ??                .block 1
0000:601F ??                .block 1
0000:6020 ??                .block 1
0000:6020 ??                .block 1 ; DATA XREF: read_dips_and_high_score_tbl+4[o
0000:6021 ??                .block 1 ; 0000:0922[r ...
0000:6021 ??                .block 1 ; DATA XREF: check_and_award_bonus+1E[o
0000:6021 ??                .block 1 ; 7/10/15/20K
0000:6022 ?? ?? ?? ??      .block 4 ; DATA XREF: sub_0_17B+27[o
0000:6026 ??                .block 1 ; DATA XREF: 0000:0087[r
0000:6026 ??                .block 1 ; 0000:099F[r ...
0000:6027 ??                .block 1
0000:6028 ??                .block 1
0000:6029 ??                .block 1
0000:602A ??                .block 1
0000:602B ??                .block 1
0000:602C ??                .block 1
0000:602D ??                .block 1
0000:602E ??                .block 1
0000:602F ??                .block 1
0000:6030 ??                .block 1
0000:6030 ??                .block 1 ; DATA XREF: 0000:1499[o
0000:6031 ??                .block 1 ; 0000:14FC[o
0000:6032 ??                .block 1
0000:6032 ??                .block 1 ; DATA XREF: 0000:158A[o
0000:6032 ??                .block 1 ; 0000:15B2[w
0000:6033 ??                .block 1
0000:6034 ??                .block 1 ; DATA XREF: 0000:14DC[o
0000:6035 ??                .block 1
0000:6036 ??                .block 1
0000:6037 ??                .block 1
0000:6038 ??                .block 1
0000:6039 ??                .block 1
0000:603A ??                .block 1
0000:603B ??                .block 1
0000:603C ??                .block 1
0000:603D ??                .block 1
0000:603E ??                .block 1
0000:603F ??                .block 1
0000:6040 ??                .block 1
0000:6040 ??                .block 1 ; DATA XREF: 0000:093E[o
0000:6040 ??                .block 1 ; 0000:09AB[o ...
0000:6041 ??                .block 1 ; game init data copied here
0000:6042 ?? ??            .block 2 ; ptr sequence data
0000:6044 ??                .block 1
0000:6045 ??                .block 1
0000:6046 ??                .block 1
0000:6047 ??                .block 1
0000:6048 ??                .block 1
0000:6048 ??                .block 1 ; DATA XREF: 0000:0909[o
0000:6048 ??                .block 1 ; 0000:091F[o ...
0000:6049 ??                .block 1
0000:604A ??                .block 1
0000:604B ??                .block 1
0000:604C ??                .block 1
0000:604D ??                .block 1
0000:604E ??                .block 1
0000:604F ??                .block 1
0000:6050 ??                .block 1
0000:6051 ??                .block 1
0000:6052 ??                .block 1
0000:6053 ??                .block 1
0000:6054 ??                .block 1
0000:6055 ??                .block 1
0000:6056 ??                .block 1
0000:6057 ??                .block 1
0000:6058 ??                .block 1
0000:6059 ??                .block 1
0000:605A ??                .block 1
0000:605B ??                .block 1
0000:605C ??                .block 1
0000:605D ??                .block 1
0000:605E ??                .block 1
0000:605F ??                .block 1
0000:6060 ??                .block 1
0000:6061 ??                .block 1
0000:6062 ??                .block 1
0000:6063 ??                .block 1
0000:6064 ??                .block 1
0000:6065 ??                .block 1
0000:6066 ??                .block 1
0000:6067 ??                .block 1
0000:6068 ??                .block 1
0000:6069 ??                .block 1
0000:606A ??                .block 1
0000:606B ??                .block 1
0000:606C ??                .block 1

```



```

0000:606D ??                .block 1
0000:606E ??                .block 1
0000:606F ??                .block 1
0000:6070 ??                .block 1
0000:6071 ??                .block 1
0000:6072 ??                .block 1
0000:6073 ??                .block 1
0000:6074 ??                .block 1
0000:6075 ??                .block 1
0000:6076 ??                .block 1
0000:6077 ??                .block 1
0000:6078 ??                .block 1
0000:6079 ??                .block 1
0000:607A ??                .block 1
0000:607B ??                .block 1
0000:607C ??                .block 1
0000:607D ??                .block 1
0000:607E ??                .block 1
0000:607F ??                .block 1
0000:6080 ??                digital_snd_tmr_walk:.block 1                ; DATA XREF: update_sounds|o
0000:6080 ??                ; stop_sound+6|o ...
0000:6081 ??                digital_snd_tmr_jump:.block 1                ; DATA XREF: sub_0_1AC3+E9|o
0000:6082 ??                digital_snd_tmr_thump:.block 1                ; DATA XREF: animate_kong_and_pauline+52|w
0000:6082 ??                ; 0000:0B45|w ...
0000:6083 ??                digital_snd_tmr_coin_spring:.block 1
0000:6084 ??                digital_snd_tmr_kong_fall:.block 1
0000:6085 ??                digital_snd_tmr_barrel_jump_priz:.block 1                ; DATA XREF: sub_0_1DBD+25|o
0000:6085 ??                ; sub_0_1DBD+87|o ...
0000:6086 ??                digital_snd_tmr_6:.block 1
0000:6087 ??                digital_snd_tmr_7:.block 1
0000:6088 ??                unk_0_6088:                .block 1                ; DATA XREF: update_sounds+2E|o
0000:6088 ??                ; 0000:12A8|w
0000:6089 ??                unk_0_6089:                .block 1                ; DATA XREF: 0000:067A|w
0000:6089 ??                ; 0000:0CC0|w ...
0000:608A ??                unk_0_608A:                .block 1                ; DATA XREF: 0000:0ADB|o
0000:608A ??                ; 0000:0BB3|o ...
0000:608B ??                unk_0_608B:                .block 1                ; DATA XREF: update_sounds+1A|o
0000:608C ??                .block 1
0000:608D ??                .block 1
0000:608E ??                .block 1
0000:608F ??                .block 1
0000:6090 ??                .block 1
0000:6091 ??                .block 1
0000:6092 ??                .block 1
0000:6093 ??                .block 1
0000:6094 ??                .block 1
0000:6095 ??                .block 1
0000:6096 ??                .block 1
0000:6097 ??                .block 1
0000:6098 ??                .block 1
0000:6099 ??                .block 1
0000:609A ??                .block 1
0000:609B ??                .block 1
0000:609C ??                .block 1
0000:609D ??                .block 1
0000:609E ??                .block 1
0000:609F ??                .block 1
0000:60A0 ??                .block 1
0000:60A1 ??                .block 1
0000:60A2 ??                .block 1
0000:60A3 ??                .block 1
0000:60A4 ??                .block 1
0000:60A5 ??                .block 1
0000:60A6 ??                .block 1
0000:60A7 ??                .block 1
0000:60A8 ??                .block 1
0000:60A9 ??                .block 1
0000:60AA ??                .block 1
0000:60AB ??                .block 1
0000:60AC ??                .block 1
0000:60AD ??                .block 1
0000:60AE ??                .block 1
0000:60AF ??                .block 1
0000:60B0 ??                unk_0_60B0:                .block 1
0000:60B1 ??                unk_0_60B1:                .block 1
0000:60B2 ?? ?? ?? ??      p1_score:                .block 3                ; DATA XREF: 0000:01C9|o
0000:60B2 ??                ; current_player_score_DE|o ...
0000:60B5 ?? ?? ?? ??      p2_score:                .block 3                ; DATA XREF: current_player_score_DE+8|o
0000:60B5 ??                ; zero_score_or_high_score+D|o ...
0000:60B8 ?? ?? ?? ??      high_score:                .block 3                ; DATA XREF: add_bonus_and_update_high_score+37|o
0000:60B8 ??                ; zero_score_or_high_score+15|o ...
0000:60BB ??                .block 1
0000:60BC ??                .block 1
0000:60BD ??                .block 1
0000:60BE ??                .block 1
0000:60BF ??                .block 1
0000:60C0 ?? ?? ?? ??      fg_vector_fn_params:.block 40h                ; DATA XREF: 0000:0291|o
0000:60C0 ?? ?? ?? ??      ; sub_0_309F+1|o
0000:6100 ?? ?? ?? ??      high_score_tbl_ram:.block 0AAh                ; DATA XREF: read_dips_and_high_score_tbl+56|o
0000:61AA ??                .block 1
0000:61AB ??                .block 1
0000:61AC ??                .block 1
0000:61AD ??                .block 1
0000:61AE ??                .block 1
0000:61AF ??                .block 1
0000:61B0 ??                .block 1
0000:61B1 ??                unk_0_61B1:                .block 1                ; DATA XREF: sub_0_13CA+D|o
0000:61B2 ??                .block 1
0000:61B3 ??                .block 1
0000:61B4 ??                .block 1
0000:61B5 ??                .block 1
0000:61B6 ??                .block 1
0000:61B7 ??                .block 1
0000:61B8 ??                .block 1
0000:61B9 ??                .block 1
0000:61BA ??                .block 1
0000:61BB ??                .block 1
0000:61BC ??                .block 1
0000:61BD ??                .block 1
0000:61BE ??                .block 1
0000:61BF ??                .block 1
0000:61C0 ??                .block 1
0000:61C1 ??                .block 1
0000:61C2 ??                .block 1
0000:61C3 ??                .block 1
0000:61C4 ??                .block 1

```



```

0000:61C5 ??                .block 1
0000:61C6 ??                .block 1
0000:61C7 ??                .block 1 ; DATA XREF: sub_0_13CA[o
0000:61C8 ??                .block 1 ; DATA XREF: sub_0_13CA+2F[o
0000:61C9 ??                .block 1
0000:61CA ??                .block 1
0000:61CB ??                .block 1
0000:61CC ??                .block 1
0000:61CD ??                .block 1
0000:61CE ??                .block 1
0000:61CF ??                .block 1
0000:61D0 ??                .block 1
0000:61D1 ??                .block 1
0000:61D2 ??                .block 1
0000:61D3 ??                .block 1
0000:61D4 ??                .block 1
0000:61D5 ??                .block 1
0000:61D6 ??                .block 1
0000:61D7 ??                .block 1
0000:61D8 ??                .block 1
0000:61D9 ??                .block 1
0000:61DA ??                .block 1
0000:61DB ??                .block 1
0000:61DC ??                .block 1
0000:61DD ??                .block 1
0000:61DE ??                .block 1
0000:61DF ??                .block 1
0000:61E0 ??                .block 1
0000:61E1 ??                .block 1
0000:61E2 ??                .block 1
0000:61E3 ??                .block 1
0000:61E4 ??                .block 1
0000:61E5 ??                .block 1
0000:61E6 ??                .block 1
0000:61E7 ??                .block 1
0000:61E8 ??                .block 1
0000:61E9 ??                .block 1
0000:61EA ??                .block 1
0000:61EB ??                .block 1
0000:61EC ??                .block 1
0000:61ED ??                .block 1
0000:61EE ??                .block 1
0000:61EF ??                .block 1
0000:61F0 ??                .block 1
0000:61F1 ??                .block 1
0000:61F2 ??                .block 1
0000:61F3 ??                .block 1
0000:61F4 ??                .block 1
0000:61F5 ??                .block 1
0000:61F6 ??                .block 1
0000:61F7 ??                .block 1
0000:61F8 ??                .block 1
0000:61F9 ??                .block 1
0000:61FA ??                .block 1
0000:61FB ??                .block 1
0000:61FC ??                .block 1
0000:61FD ??                .block 1
0000:61FE ??                .block 1
0000:61FF ??                .block 1
0000:6200 ??                mario_alive_flag: .block 1 ; DATA XREF: return_if_mario_not_alive[r
0000:6200 ??                ; 0000:0BE3[r ...
0000:6201 ??                .block 1
0000:6202 ??                unk_0_6202: .block 1
0000:6203 ??                mario_y_coord: .block 1
0000:6203 ??                ; DATA XREF: animate_kong_and_pauline+D6[r
0000:6204 ??                unk_0_6204: .block 1 ; animate_kong_and_pauline+10E[r ...
0000:6205 ??                mario_x_coord: .block 1
0000:6205 ??                ; DATA XREF: sub_0_19DA+13[r
0000:6206 ??                unk_0_6206: .block 1 ; sub_0_1A33+22[r ...
0000:6207 ??                mario_flip_sprite_tile: .block 1 ; DATA XREF: sub_0_1AC3+54[o
0000:6207 ??                ; sub_0_1AC3+9D[o ...
0000:6208 ??                mario_flip_colour: .block 1
0000:6209 ??                unk_0_6209: .block 1 ; DATA XREF: 0000:0FA5[o
0000:620A ??                .block 1
0000:620B ??                mario_x_before_jump: .block 1
0000:620C ??                mario_y_before_jump: .block 1
0000:620D ??                .block 1
0000:620E ??                unk_0_620E: .block 1 ; DATA XREF: sub_0_1AC3+E6[w
0000:620E ??                ; sub_0_1AC3+1B6[o ...
0000:620F ??                unk_0_620F: .block 1 ; DATA XREF: sub_0_1AC3+1CE[r
0000:620F ??                ; sub_0_1AC3+1EA[r ...
0000:6210 ??                unk_0_6210: .block 1 ; mario_???
0000:6210 ??                ; DATA XREF: sub_0_1AC3+B0[o
0000:6210 ??                ; sub_0_1F46+F[w ...
0000:6211 ??                unk_0_6211: .block 1
0000:6212 ??                unk_0_6212: .block 1
0000:6213 ??                unk_0_6213: .block 1
0000:6214 ??                unk_0_6214: .block 1
0000:6215 ??                mario_climbing: .block 1
0000:6216 ??                mario_jumping: .block 1
0000:6217 ??                unk_0_6217: .block 1
0000:6218 ??                unk_0_6218: .block 1
0000:6219 ??                unk_0_6219: .block 1
0000:621A ??                unk_0_621A: .block 1 ; DATA XREF: sub_0_1AC3+5D[o
0000:621A ??                ; sub_0_1AC3+2B3[r ...
0000:621B ??                .block 1
0000:621C ??                unk_0_621C: .block 1 ; DATA XREF: sub_0_1AC3+262[o
0000:621C ??                ; sub_0_1AC3+2BD[r
0000:621D ??                .block 1
0000:621E ??                unk_0_621E: .block 1 ; DATA XREF: sub_0_1AC3+7[r
0000:621E ??                ; sub_0_1AC3+92[o ...
0000:621F ??                unk_0_621F: .block 1
0000:6220 ??                unk_0_6220: .block 1
0000:6221 ??                unk_0_6221: .block 1
0000:6222 ??                unk_0_6222: .block 1
0000:6223 ??                .block 1
0000:6224 ??                unk_0_6224: .block 1
0000:6225 ??                unk_0_6225: .block 1
0000:6226 ??                .block 1
0000:6227 ??                level_type: .block 1 ; DATA XREF: sub_0_30+14[o
0000:6227 ??                ; 0000:01EA[w ...
0000:6228 ??                lives_left: .block 1 ; DATA XREF: 0000:01D9[w
0000:6228 ??                ; check_and_award_bonus+28[o ...
0000:6229 ??                level: .block 1 ; DATA XREF: 0000:01D6[w
0000:6229 ??                ; sub_0_37F+15[r ...
0000:6229 ??                ; keeps incrementing

```

```

0000:622A ?? ?? seq_data: .block 2
0000:622C ?? unk_0_622C: .block 1 ; DATA XREF: 0000:0A6E!o
0000:622C ; 0000:12F6!w ...
0000:622D ?? awarded_bonus_life: .block 1
0000:622E ?? unk_0_622E: .block 1 ; DATA XREF: 0000:0C05!r
0000:622E ; 0000:0C0E!w ...
0000:622F ?? unk_0_622F: .block 1
0000:6230 ?? .block 1
0000:6231 ?? .block 1
0000:6232 ?? .block 1
0000:6233 ?? .block 1
0000:6234 ?? .block 1
0000:6235 ?? .block 1
0000:6236 ?? .block 1
0000:6237 ?? .block 1
0000:6238 ?? .block 1
0000:6239 ?? .block 1
0000:623A ?? .block 1
0000:623B ?? .block 1
0000:623C ?? .block 1
0000:623D ?? .block 1
0000:623E ?? .block 1
0000:623F ?? .block 1
0000:6240 ?? .block 1
0000:6241 ?? .block 1
0000:6242 ?? .block 1
0000:6243 ?? .block 1
0000:6244 ?? .block 1
0000:6245 ?? .block 1
0000:6246 ?? .block 1
0000:6247 ?? .block 1
0000:6248 ?? .block 1
0000:6249 ?? .block 1
0000:624A ?? .block 1
0000:624B ?? .block 1
0000:624C ?? .block 1
0000:624D ?? .block 1
0000:624E ?? .block 1
0000:624F ?? .block 1
0000:6250 ?? .block 1
0000:6251 ?? .block 1
0000:6252 ?? .block 1
0000:6253 ?? .block 1
0000:6254 ?? .block 1
0000:6255 ?? .block 1
0000:6256 ?? .block 1
0000:6257 ?? .block 1
0000:6258 ?? .block 1
0000:6259 ?? .block 1
0000:625A ?? .block 1
0000:625B ?? .block 1
0000:625C ?? .block 1
0000:625D ?? .block 1
0000:625E ?? .block 1
0000:625F ?? .block 1
0000:6260 ?? .block 1
0000:6261 ?? .block 1
0000:6262 ?? .block 1
0000:6263 ?? .block 1
0000:6264 ?? .block 1
0000:6265 ?? .block 1
0000:6266 ?? .block 1
0000:6267 ?? .block 1
0000:6268 ?? .block 1
0000:6269 ?? .block 1
0000:626A ?? .block 1
0000:626B ?? .block 1
0000:626C ?? .block 1
0000:626D ?? .block 1
0000:626E ?? .block 1
0000:626F ?? .block 1
0000:6270 ?? .block 1
0000:6271 ?? .block 1
0000:6272 ?? .block 1
0000:6273 ?? .block 1
0000:6274 ?? .block 1
0000:6275 ?? .block 1
0000:6276 ?? .block 1
0000:6277 ?? .block 1
0000:6278 ?? .block 1
0000:6279 ?? .block 1
0000:627A ?? .block 1
0000:627B ?? .block 1
0000:627C ?? .block 1
0000:627D ?? .block 1
0000:627E ?? .block 1
0000:627F ?? .block 1
0000:6280 ?? unk_0_6280: .block 1 ; DATA XREF: 0000:0F64!o
0000:6280 ; 0000:0F72!o ...
0000:6281 ?? .block 1
0000:6282 ?? .block 1
0000:6283 ?? .block 1
0000:6284 ?? .block 1
0000:6285 ?? .block 1
0000:6286 ?? .block 1
0000:6287 ?? .block 1
0000:6288 ?? unk_0_6288: .block 1 ; DATA XREF: sub_0_2207+E!o
0000:6289 ?? .block 1
0000:628A ?? .block 1
0000:628B ?? .block 1
0000:628C ?? .block 1
0000:628D ?? .block 1
0000:628E ?? .block 1
0000:628F ?? .block 1
0000:6290 ?? unk_0_6290: .block 1 ; DATA XREF: sub_0_1A33+53!o
0000:6290 ; sub_0_1E57+29!r
0000:6291 ?? unk_0_6291: .block 1
0000:6292 ?? unk_0_6292: .block 1 ; DATA XREF: sub_0_1A33+48!o
0000:6293 ?? .block 1
0000:6294 ?? .block 1
0000:6295 ?? .block 1
0000:6296 ?? .block 1
0000:6297 ?? .block 1
0000:6298 ?? .block 1
0000:6299 ?? .block 1
0000:629A ?? .block 1

```

```

0000:629B ??                .block 1
0000:629C ??                .block 1
0000:629D ??                .block 1
0000:629E ??                .block 1
0000:629F ??                .block 1
0000:62A0 ??                .block 1
0000:62A0 ??                unk_0_62A0:
                                ; DATA XREF: 0000:16BC|w
                                ; 0000:16D2|w ...
0000:62A1 ??                unk_0_62A1: .block 1
                                ; DATA XREF: sub_0_2602+14|o
0000:62A2 ??                .block 1
0000:62A3 ??                unk_0_62A3: .block 1
                                ; DATA XREF: sub_0_2523+2E|r
                                ; sub_0_262F|o ...
0000:62A4 ??                .block 1
0000:62A5 ??                unk_0_62A5: .block 1
                                ; DATA XREF: sub_0_2679+7|o
0000:62A6 ??                unk_0_62A6: .block 1
                                ; DATA XREF: sub_0_2523+39|r
                                ; sub_0_2679+14|o
                                ; DATA XREF: sub_0_27DA|o
0000:62A7 ??                unk_0_62A7: .block 1
0000:62A8 ??                unk_0_62A8: .block 1
0000:62A9 ??                .block 1
0000:62AA ??                unk_0_62AA: .block 1
0000:62AB ??                .block 1
0000:62AC ??                unk_0_62AC: .block 1
0000:62AD ??                .block 1
0000:62AE ??                .block 1
0000:62AF ??                unk_0_62AF: .block 1
                                ; DATA XREF: 0000:0AA6|w
                                ; 0000:0AEB|r ...
0000:62AF ??                .block 1
                                ; DATA XREF: 0000:063A|r
                                ; 0000:0F8E|o ...
0000:62B0 ??                unk_0_62B0: .block 1
                                ; DATA XREF: sub_0_2C03+9|r
                                ; sub_0_2C8F+4B|o ...
0000:62B1 ??                unk_0_62B1: .block 1
0000:62B2 ??                unk_0_62B2: .block 1
0000:62B3 ??                unk_0_62B3: .block 1
0000:62B4 ??                unk_0_62B4: .block 1
                                ; DATA XREF: sub_0_2FCB+3|o
0000:62B5 ??                .block 1
0000:62B6 ??                .block 1
0000:62B7 ??                .block 1
0000:62B8 ??                unk_0_62B8: .block 1
                                ; DATA XREF: sub_0_3A2+9|o
0000:62B9 ??                unk_0_62B9: .block 1
0000:62BA ??                unk_0_62BA: .block 1
                                ; DATA XREF: sub_0_3A2+2F|o
                                ; sub_0_3A2+3E|w
0000:62BA ??                .block 1
0000:62BB ??                .block 1
0000:62BC ??                .block 1
0000:62BD ??                .block 1
0000:62BE ??                .block 1
0000:62BF ??                .block 1
0000:62C0 ??                .block 1
0000:62C1 ??                .block 1
0000:62C2 ??                .block 1
0000:62C3 ??                .block 1
0000:62C4 ??                .block 1
0000:62C5 ??                .block 1
0000:62C6 ??                .block 1
0000:62C7 ??                .block 1
0000:62C8 ??                .block 1
0000:62C9 ??                .block 1
0000:62CA ??                .block 1
0000:62CB ??                .block 1
0000:62CC ??                .block 1
0000:62CD ??                .block 1
0000:62CE ??                .block 1
0000:62CF ??                .block 1
0000:62D0 ??                .block 1
0000:62D1 ??                .block 1
0000:62D2 ??                .block 1
0000:62D3 ??                .block 1
0000:62D4 ??                .block 1
0000:62D5 ??                .block 1
0000:62D6 ??                .block 1
0000:62D7 ??                .block 1
0000:62D8 ??                .block 1
0000:62D9 ??                .block 1
0000:62DA ??                .block 1
0000:62DB ??                .block 1
0000:62DC ??                .block 1
0000:62DD ??                .block 1
0000:62DE ??                .block 1
0000:62DF ??                .block 1
0000:62E0 ??                .block 1
0000:62E1 ??                .block 1
0000:62E2 ??                .block 1
0000:62E3 ??                .block 1
0000:62E4 ??                .block 1
0000:62E5 ??                .block 1
0000:62E6 ??                .block 1
0000:62E7 ??                .block 1
0000:62E8 ??                .block 1
0000:62E9 ??                .block 1
0000:62EA ??                .block 1
0000:62EB ??                .block 1
0000:62EC ??                .block 1
0000:62ED ??                .block 1
0000:62EE ??                .block 1
0000:62EF ??                .block 1
0000:62F0 ??                .block 1
0000:62F1 ??                .block 1
0000:62F2 ??                .block 1
0000:62F3 ??                .block 1
0000:62F4 ??                .block 1
0000:62F5 ??                .block 1
0000:62F6 ??                .block 1
0000:62F7 ??                .block 1
0000:62F8 ??                .block 1
0000:62F9 ??                .block 1
0000:62FA ??                .block 1
0000:62FB ??                .block 1
0000:62FC ??                .block 1
0000:62FD ??                .block 1
0000:62FE ??                .block 1
0000:62FF ??                .block 1
0000:6300 ??                unk_0_6300: .block 1
                                ; DATA XREF: sub_0_236E|o
                                ; sub_0_2441+30|o
0000:6300 ??                .block 1
0000:6301 ??                .block 1
0000:6302 ??                .block 1
0000:6303 ??                .block 1
0000:6304 ??                .block 1
0000:6305 ??                .block 1
0000:6306 ??                .block 1

```

```
0000:6307 ?? .block 1
0000:6308 ?? .block 1
0000:6309 ?? .block 1
0000:630A ?? .block 1
0000:630B ?? .block 1
0000:630C ?? .block 1
0000:630D ?? .block 1
0000:630E ?? .block 1
0000:630F ?? .block 1
0000:6310 ?? unk_0_6310: .block 1 ; DATA XREF: sub_0_2441+B|o
0000:6311 ?? .block 1
0000:6312 ?? .block 1
0000:6313 ?? .block 1
0000:6314 ?? .block 1
0000:6315 ?? .block 1
0000:6316 ?? .block 1
0000:6317 ?? .block 1
0000:6318 ?? .block 1
0000:6319 ?? .block 1
0000:631A ?? .block 1
0000:631B ?? .block 1
0000:631C ?? .block 1
0000:631D ?? .block 1
0000:631E ?? .block 1
0000:631F ?? .block 1
0000:6320 ?? .block 1
0000:6321 ?? .block 1
0000:6322 ?? .block 1
0000:6323 ?? .block 1
0000:6324 ?? .block 1
0000:6325 ?? .block 1
0000:6326 ?? .block 1
0000:6327 ?? .block 1
0000:6328 ?? .block 1
0000:6329 ?? .block 1
0000:632A ?? .block 1
0000:632B ?? .block 1
0000:632C ?? .block 1
0000:632D ?? .block 1
0000:632E ?? .block 1
0000:632F ?? .block 1
0000:6330 ?? .block 1
0000:6331 ?? .block 1
0000:6332 ?? .block 1
0000:6333 ?? .block 1
0000:6334 ?? .block 1
0000:6335 ?? .block 1
0000:6336 ?? .block 1
0000:6337 ?? .block 1
0000:6338 ?? .block 1
0000:6339 ?? .block 1
0000:633A ?? .block 1
0000:633B ?? .block 1
0000:633C ?? .block 1
0000:633D ?? .block 1
0000:633E ?? .block 1
0000:633F ?? .block 1
0000:6340 ?? unk_0_6340: .block 1
0000:6341 ?? unk_0_6341: .block 1 ; DATA XREF: sub_0_1DBD+E|w
; 0000:1E4A|o
0000:6342 ?? unk_0_6342: .block 1
0000:6343 ?? unk_0_6343: .block 1
0000:6344 ?? .block 1
0000:6345 ?? unk_0_6345: .block 1 ; DATA XREF: sub_0_1E96|r
; sub_0_1E96+60|o
; DATA XREF: 0000:1F09|o
; 0000:1F23|o
0000:6346 ?? unk_0_6346: .block 1
0000:6347 ?? .block 1
0000:6348 ?? unk_0_6348: .block 1
0000:6349 ?? .block 1
0000:634A ?? .block 1
0000:634B ?? .block 1
0000:634C ?? .block 1
0000:634D ?? .block 1
0000:634E ?? .block 1
0000:634F ?? .block 1
0000:6350 ?? unk_0_6350: .block 1
0000:6351 ?? unk_0_6351: .block 1
0000:6352 ?? unk_0_6352: .block 1
0000:6353 ?? unk_0_6353: .block 1
0000:6354 ?? unk_0_6354: .block 1
0000:6355 ?? .block 1
0000:6356 ?? .block 1
0000:6357 ?? .block 1
0000:6358 ?? .block 1
0000:6359 ?? .block 1
0000:635A ?? .block 1
0000:635B ?? .block 1
0000:635C ?? .block 1
0000:635D ?? .block 1
0000:635E ?? .block 1
0000:635F ?? .block 1
0000:6360 ?? .block 1
0000:6361 ?? .block 1
0000:6362 ?? .block 1
0000:6363 ?? .block 1
0000:6364 ?? .block 1
0000:6365 ?? .block 1
0000:6366 ?? .block 1
0000:6367 ?? .block 1
0000:6368 ?? .block 1
0000:6369 ?? .block 1
0000:636A ?? .block 1
0000:636B ?? .block 1
0000:636C ?? .block 1
0000:636D ?? .block 1
0000:636E ?? .block 1
0000:636F ?? .block 1
0000:6370 ?? .block 1
0000:6371 ?? .block 1
0000:6372 ?? .block 1
0000:6373 ?? .block 1
0000:6374 ?? .block 1
0000:6375 ?? .block 1
0000:6376 ?? .block 1
0000:6377 ?? .block 1
```

```

0000:6378 ??                .block 1
0000:6379 ??                .block 1
0000:637A ??                .block 1
0000:637B ??                .block 1
0000:637C ??                .block 1
0000:637D ??                .block 1
0000:637E ??                .block 1
0000:637F ??                .block 1
0000:6380 ??                .block 1
0000:6381 ??                .block 1
0000:6382 ??                .block 1
0000:6383 ??                .block 1
0000:6384 ??                .block 1
0000:6385 ??                .block 1
0000:6386 ??                .block 1
0000:6387 ??                .block 1
0000:6388 ??                .block 1
0000:6388 ??                .block 1
0000:6389 ??                .block 1
0000:638A ??                .block 1
0000:638A ??                .block 1
0000:638B ??                .block 1
0000:638C ??                .block 1
0000:638D ??                .block 1
0000:638D ??                .block 1
0000:638E ??                .block 1
0000:638E ??                .block 1
0000:638F ??                .block 1
0000:638F ??                .block 1
0000:6390 ??                .block 1
0000:6390 ??                .block 1
0000:6391 ??                .block 1
0000:6392 ??                .block 1
0000:6393 ??                .block 1
0000:6394 ??                .block 1
0000:6394 ??                .block 1
0000:6395 ??                .block 1
0000:6395 ??                .block 1
0000:6396 ??                .block 1
0000:6397 ??                .block 1
0000:6398 ??                .block 1
0000:6399 ??                .block 1
0000:639A ??                .block 1
0000:639B ??                .block 1
0000:639B ??                .block 1
0000:639C ??                .block 1
0000:639D ??                .block 1
0000:639D ??                .block 1
0000:639E ??                .block 1
0000:639E ??                .block 1
0000:639F ??                .block 1
0000:63A0 ??                .block 1
0000:63A0 ??                .block 1
0000:63A0 ??                .block 1
0000:63A1 ??                .block 1
0000:63A2 ??                .block 1
0000:63A3 ??                .block 1
0000:63A4 ??                .block 1
0000:63A5 ??                .block 1
0000:63A6 ??                .block 1
0000:63A7 ??                .block 1
0000:63A7 ??                .block 1
0000:63A8 ??                .block 1
0000:63A9 ??                .block 1
0000:63AA ??                .block 1
0000:63AB ??                .block 1
0000:63AC ??                .block 1
0000:63AD ??                .block 1
0000:63AE ??                .block 1
0000:63AF ??                .block 1
0000:63B0 ??                .block 1
0000:63B1 ??                .block 1
0000:63B2 ??                .block 1
0000:63B3 ??                .block 1
0000:63B4 ??                .block 1
0000:63B5 ??                .block 1
0000:63B6 ??                .block 1
0000:63B7 ??                .block 1
0000:63B8 ??                .block 1
0000:63B8 ??                .block 1
0000:63B9 ??                .block 1
0000:63BA ??                .block 1
0000:63BB ??                .block 1
0000:63BC ??                .block 1
0000:63BD ??                .block 1
0000:63BE ??                .block 1
0000:63BF ??                .block 1
0000:63C0 ??                .block 1
0000:63C1 ??                .block 1
0000:63C2 ??                .block 1
0000:63C3 ??                .block 1
0000:63C4 ??                .block 1
0000:63C5 ??                .block 1
0000:63C6 ??                .block 1
0000:63C7 ??                .block 1
0000:63C8 ??                .block 1
0000:63C9 ??                .block 1
0000:63CA ??                .block 1
0000:63CB ??                .block 1
0000:63CC ??                .block 1
0000:63CD ??                .block 1
0000:63CE ??                .block 1
0000:63CF ??                .block 1
0000:63D0 ??                .block 1
0000:63D1 ??                .block 1
0000:63D2 ??                .block 1
0000:63D3 ??                .block 1
0000:63D4 ??                .block 1
0000:63D5 ??                .block 1
0000:63D6 ??                .block 1
0000:63D7 ??                .block 1
0000:63D8 ??                .block 1
0000:63D9 ??                .block 1
0000:63DA ??                .block 1
0000:63DB ??                .block 1
0000:63DC ??                .block 1

```

```

; DATA XREF: sub_0_37F+7|o

```

```

; DATA XREF: 0000:02D1|o
; DATA XREF: sub_0_37F|o
; DATA XREF: 0000:0A76|r
; 0000:0ABA|o ...

```

```

; DATA XREF: 0000:161F|r
; 0000:1633|r ...

```

```

; DATA XREF: 0000:07CB|r
; 0000:07D5|w ...

```

```

; DATA XREF: 0000:0B58|w
; 0000:0B94|r ...
; DATA XREF: 0000:0AD4|w
; 0000:0B3B|r ...
; DATA XREF: sub_0_2C03+4C|w
; sub_0_2C8F+8D|r ...
; DATA XREF: animate_kong_and_pauline+2B|o
; animate_kong_and_pauline+8B|r ...

```

```

; DATA XREF: sub_0_2ED4+4A|r
; sub_0_2ED4+75|o
; DATA XREF: sub_0_2ED4+7C|o
; sub_0_2ED4+87|w ...

```

```

; DATA XREF: sub_0_2523|o
; sub_0_2523+65|w

```

```

; DATA XREF: 0000:127F|r
; 0000:1295|o ...
; DATA XREF: 0000:129B|w
; 0000:12B2|o

```

```

; DATA XREF: sub_0_3A2+39|w
; 0000:0768|w ...

```

```

; DATA XREF: 0000:0BFA|o
; 0000:0C43|r ...

```

```

; DATA XREF: 0000:0635|r
; 0000:06AC|o

```

```

; DATA XREF: next_attract_action+3|o

```

```
0000:63DD ?? .block 1
0000:63DE ?? .block 1
0000:63DF ?? .block 1
0000:63E0 ?? unk_0_63E0: .block 1 ; DATA XREF: sub_0_31B1+7[o
0000:63E1 ?? .block 1
0000:63E2 ?? .block 1
0000:63E3 ?? .block 1
0000:63E4 ?? .block 1
0000:63E5 ?? .block 1
0000:63E6 ?? .block 1
0000:63E7 ?? .block 1
0000:63E8 ?? .block 1
0000:63E9 ?? .block 1
0000:63EA ?? .block 1
0000:63EB ?? .block 1
0000:63EC ?? .block 1
0000:63ED ?? .block 1
0000:63EE ?? .block 1
0000:63EF ?? .block 1
0000:63F0 ?? .block 1
0000:63F1 ?? .block 1
0000:63F2 ?? .block 1
0000:63F3 ?? .block 1
0000:63F4 ?? .block 1
0000:63F5 ?? .block 1
0000:63F6 ?? .block 1
0000:63F7 ?? .block 1
0000:63F8 ?? .block 1
0000:63F9 ?? .block 1
0000:63FA ?? .block 1
0000:63FB ?? .block 1
0000:63FC ?? .block 1
0000:63FD ?? .block 1
0000:63FE ?? .block 1
0000:63FF ?? .block 1
0000:6400 ?? unk_0_6400: .block 1 ; DATA XREF: 0000:10E9[o
; sub_0_286F+2A[o ...
0000:6401 ?? .block 1
0000:6402 ?? .block 1
0000:6403 ?? .block 1
0000:6404 ?? .block 1
0000:6405 ?? .block 1
0000:6406 ?? .block 1
0000:6407 ?? unk_0_6407: .block 1 ; DATA XREF: 0000:0FE5[o
; 0000:1022[o ...
0000:6408 ?? .block 1
0000:6409 ?? .block 1
0000:640A ?? .block 1
0000:640B ?? .block 1
0000:640C ?? .block 1
0000:640D ?? .block 1
0000:640E ?? .block 1
0000:640F ?? .block 1
0000:6410 ?? .block 1
0000:6411 ?? .block 1
0000:6412 ?? .block 1
0000:6413 ?? .block 1
0000:6414 ?? .block 1
0000:6415 ?? .block 1
0000:6416 ?? .block 1
0000:6417 ?? .block 1
0000:6418 ?? .block 1
0000:6419 ?? .block 1
0000:641A ?? .block 1
0000:641B ?? .block 1
0000:641C ?? .block 1
0000:641D ?? .block 1
0000:641E ?? .block 1
0000:641F ?? .block 1
0000:6420 ?? .block 1
0000:6421 ?? .block 1
0000:6422 ?? .block 1
0000:6423 ?? .block 1
0000:6424 ?? .block 1
0000:6425 ?? .block 1
0000:6426 ?? .block 1
0000:6427 ?? .block 1
0000:6428 ?? .block 1
0000:6429 ?? .block 1
0000:642A ?? .block 1
0000:642B ?? .block 1
0000:642C ?? .block 1
0000:642D ?? .block 1
0000:642E ?? .block 1
0000:642F ?? .block 1
0000:6430 ?? .block 1
0000:6431 ?? .block 1
0000:6432 ?? .block 1
0000:6433 ?? .block 1
0000:6434 ?? .block 1
0000:6435 ?? .block 1
0000:6436 ?? .block 1
0000:6437 ?? .block 1
0000:6438 ?? .block 1
0000:6439 ?? unk_0_6439: .block 1 ; DATA XREF: sub_0_31DD+C[o
0000:643A ?? .block 1
0000:643B ?? .block 1
0000:643C ?? .block 1
0000:643D ?? .block 1
0000:643E ?? .block 1
0000:643F ?? .block 1
0000:6440 ?? .block 1
0000:6441 ?? .block 1
0000:6442 ?? .block 1
0000:6443 ?? .block 1
0000:6444 ?? .block 1
0000:6445 ?? .block 1
0000:6446 ?? .block 1
0000:6447 ?? .block 1
0000:6448 ?? .block 1
0000:6449 ?? .block 1
0000:644A ?? .block 1
0000:644B ?? .block 1
0000:644C ?? .block 1
0000:644D ?? .block 1
0000:644E ?? .block 1
```

```
0000:644F ?? .block 1
0000:6450 ?? .block 1
0000:6451 ?? .block 1
0000:6452 ?? .block 1
0000:6453 ?? .block 1
0000:6454 ?? .block 1
0000:6455 ?? .block 1
0000:6456 ?? .block 1
0000:6457 ?? .block 1
0000:6458 ?? .block 1
0000:6459 ?? .block 1
0000:645A ?? .block 1
0000:645B ?? .block 1
0000:645C ?? .block 1
0000:645D ?? .block 1
0000:645E ?? .block 1
0000:645F ?? .block 1
0000:6460 ?? .block 1
0000:6461 ?? .block 1
0000:6462 ?? .block 1
0000:6463 ?? .block 1
0000:6464 ?? .block 1
0000:6465 ?? .block 1
0000:6466 ?? .block 1
0000:6467 ?? .block 1
0000:6468 ?? .block 1
0000:6469 ?? .block 1
0000:646A ?? .block 1
0000:646B ?? .block 1
0000:646C ?? .block 1
0000:646D ?? .block 1
0000:646E ?? .block 1
0000:646F ?? .block 1
0000:6470 ?? .block 1
0000:6471 ?? .block 1
0000:6472 ?? .block 1
0000:6473 ?? .block 1
0000:6474 ?? .block 1
0000:6475 ?? .block 1
0000:6476 ?? .block 1
0000:6477 ?? .block 1
0000:6478 ?? .block 1
0000:6479 ?? unk_0_6479: ; DATA XREF: sub_0_31DD+12|o
0000:647A ?? .block 1
0000:647B ?? .block 1
0000:647C ?? .block 1
0000:647D ?? .block 1
0000:647E ?? .block 1
0000:647F ?? .block 1
0000:6480 ?? .block 1
0000:6481 ?? .block 1
0000:6482 ?? .block 1
0000:6483 ?? .block 1
0000:6484 ?? .block 1
0000:6485 ?? .block 1
0000:6486 ?? .block 1
0000:6487 ?? .block 1
0000:6488 ?? .block 1
0000:6489 ?? .block 1
0000:648A ?? .block 1
0000:648B ?? .block 1
0000:648C ?? .block 1
0000:648D ?? .block 1
0000:648E ?? .block 1
0000:648F ?? .block 1
0000:6490 ?? .block 1
0000:6491 ?? .block 1
0000:6492 ?? .block 1
0000:6493 ?? .block 1
0000:6494 ?? .block 1
0000:6495 ?? .block 1
0000:6496 ?? .block 1
0000:6497 ?? .block 1
0000:6498 ?? .block 1
0000:6499 ?? .block 1
0000:649A ?? .block 1
0000:649B ?? .block 1
0000:649C ?? .block 1
0000:649D ?? .block 1
0000:649E ?? .block 1
0000:649F ?? .block 1
0000:64A0 ?? unk_0_64A0: ; DATA XREF: 0000:1166|o
0000:64A1 ?? .block 1
0000:64A2 ?? .block 1
0000:64A3 ?? unk_0_64A3: ; DATA XREF: 0000:1151|o
0000:64A4 ?? .block 1
0000:64A5 ?? .block 1
0000:64A6 ?? .block 1
0000:64A7 ?? unk_0_64A7: ; DATA XREF: 0000:115D|o
0000:64A8 ?? .block 1
0000:64A9 ?? .block 1
0000:64AA ?? .block 1
0000:64AB ?? .block 1
0000:64AC ?? .block 1
0000:64AD ?? .block 1
0000:64AE ?? .block 1
0000:64AF ?? .block 1
0000:64B0 ?? .block 1
0000:64B1 ?? .block 1
0000:64B2 ?? .block 1
0000:64B3 ?? .block 1
0000:64B4 ?? .block 1
0000:64B5 ?? .block 1
0000:64B6 ?? .block 1
0000:64B7 ?? .block 1
0000:64B8 ?? .block 1
0000:64B9 ?? .block 1
0000:64BA ?? .block 1
0000:64BB ?? .block 1
0000:64BC ?? .block 1
0000:64BD ?? .block 1
0000:64BE ?? .block 1
0000:64BF ?? .block 1
0000:64C0 ?? .block 1
0000:64C1 ?? .block 1
0000:64C2 ?? .block 1
```

```
0000:64C3 ?? .block 1
0000:64C4 ?? .block 1
0000:64C5 ?? .block 1
0000:64C6 ?? .block 1
0000:64C7 ?? .block 1
0000:64C8 ?? .block 1
0000:64C9 ?? .block 1
0000:64CA ?? .block 1
0000:64CB ?? .block 1
0000:64CC ?? .block 1
0000:64CD ?? .block 1
0000:64CE ?? .block 1
0000:64CF ?? .block 1
0000:64D0 ?? .block 1
0000:64D1 ?? .block 1
0000:64D2 ?? .block 1
0000:64D3 ?? .block 1
0000:64D4 ?? .block 1
0000:64D5 ?? .block 1
0000:64D6 ?? .block 1
0000:64D7 ?? .block 1
0000:64D8 ?? .block 1
0000:64D9 ?? .block 1
0000:64DA ?? .block 1
0000:64DB ?? .block 1
0000:64DC ?? .block 1
0000:64DD ?? .block 1
0000:64DE ?? .block 1
0000:64DF ?? .block 1
0000:64E0 ?? .block 1
0000:64E1 ?? .block 1
0000:64E2 ?? .block 1
0000:64E3 ?? .block 1
0000:64E4 ?? .block 1
0000:64E5 ?? .block 1
0000:64E6 ?? .block 1
0000:64E7 ?? .block 1
0000:64E8 ?? .block 1
0000:64E9 ?? .block 1
0000:64EA ?? .block 1
0000:64EB ?? .block 1
0000:64EC ?? .block 1
0000:64ED ?? .block 1
0000:64EE ?? .block 1
0000:64EF ?? .block 1
0000:64F0 ?? .block 1
0000:64F1 ?? .block 1
0000:64F2 ?? .block 1
0000:64F3 ?? .block 1
0000:64F4 ?? .block 1
0000:64F5 ?? .block 1
0000:64F6 ?? .block 1
0000:64F7 ?? .block 1
0000:64F8 ?? .block 1
0000:64F9 ?? .block 1
0000:64FA ?? .block 1
0000:64FB ?? .block 1
0000:64FC ?? .block 1
0000:64FD ?? .block 1
0000:64FE ?? .block 1
0000:64FF ?? .block 1
0000:6500 ?? unk_0_6500: .block 1 ; DATA XREF: sub_0_1186+C↑o
; 0000:28F9↑o ...
0000:6501 ?? .block 1
0000:6502 ?? .block 1
0000:6503 ?? .block 1
0000:6504 ?? .block 1
0000:6505 ?? .block 1
0000:6506 ?? .block 1
0000:6507 ?? unk_0_6507: .block 1 ; DATA XREF: sub_0_1186+3↑o
0000:6508 ?? .block 1
0000:6509 ?? .block 1
0000:650A ?? .block 1
0000:650B ?? .block 1
0000:650C ?? .block 1
0000:650D ?? .block 1
0000:650E ?? .block 1
0000:650F ?? .block 1
0000:6510 ?? .block 1
0000:6511 ?? .block 1
0000:6512 ?? .block 1
0000:6513 ?? .block 1
0000:6514 ?? .block 1
0000:6515 ?? .block 1
0000:6516 ?? .block 1
0000:6517 ?? .block 1
0000:6518 ?? .block 1
0000:6519 ?? .block 1
0000:651A ?? .block 1
0000:651B ?? .block 1
0000:651C ?? .block 1
0000:651D ?? .block 1
0000:651E ?? .block 1
0000:651F ?? .block 1
0000:6520 ?? .block 1
0000:6521 ?? .block 1
0000:6522 ?? .block 1
0000:6523 ?? .block 1
0000:6524 ?? .block 1
0000:6525 ?? .block 1
0000:6526 ?? .block 1
0000:6527 ?? .block 1
0000:6528 ?? .block 1
0000:6529 ?? .block 1
0000:652A ?? .block 1
0000:652B ?? .block 1
0000:652C ?? .block 1
0000:652D ?? .block 1
0000:652E ?? .block 1
0000:652F ?? .block 1
0000:6530 ?? .block 1
0000:6531 ?? .block 1
0000:6532 ?? .block 1
0000:6533 ?? .block 1
0000:6534 ?? .block 1
0000:6535 ?? .block 1
```



```
0000:6536 ?? .block 1
0000:6537 ?? .block 1
0000:6538 ?? .block 1
0000:6539 ?? .block 1
0000:653A ?? .block 1
0000:653B ?? .block 1
0000:653C ?? .block 1
0000:653D ?? .block 1
0000:653E ?? .block 1
0000:653F ?? .block 1
0000:6540 ?? .block 1
0000:6541 ?? .block 1
0000:6542 ?? .block 1
0000:6543 ?? .block 1
0000:6544 ?? .block 1
0000:6545 ?? .block 1
0000:6546 ?? .block 1
0000:6547 ?? .block 1
0000:6548 ?? .block 1
0000:6549 ?? .block 1
0000:654A ?? .block 1
0000:654B ?? .block 1
0000:654C ?? .block 1
0000:654D ?? .block 1
0000:654E ?? .block 1
0000:654F ?? .block 1
0000:6550 ?? .block 1
0000:6551 ?? .block 1
0000:6552 ?? .block 1
0000:6553 ?? .block 1
0000:6554 ?? .block 1
0000:6555 ?? .block 1
0000:6556 ?? .block 1
0000:6557 ?? .block 1
0000:6558 ?? .block 1
0000:6559 ?? .block 1
0000:655A ?? .block 1
0000:655B ?? .block 1
0000:655C ?? .block 1
0000:655D ?? .block 1
0000:655E ?? .block 1
0000:655F ?? .block 1
0000:6560 ?? .block 1
0000:6561 ?? .block 1
0000:6562 ?? .block 1
0000:6563 ?? .block 1
0000:6564 ?? .block 1
0000:6565 ?? .block 1
0000:6566 ?? .block 1
0000:6567 ?? .block 1
0000:6568 ?? .block 1
0000:6569 ?? .block 1
0000:656A ?? .block 1
0000:656B ?? .block 1
0000:656C ?? .block 1
0000:656D ?? .block 1
0000:656E ?? .block 1
0000:656F ?? .block 1
0000:6570 ?? .block 1
0000:6571 ?? .block 1
0000:6572 ?? .block 1
0000:6573 ?? .block 1
0000:6574 ?? .block 1
0000:6575 ?? .block 1
0000:6576 ?? .block 1
0000:6577 ?? .block 1
0000:6578 ?? .block 1
0000:6579 ?? .block 1
0000:657A ?? .block 1
0000:657B ?? .block 1
0000:657C ?? .block 1
0000:657D ?? .block 1
0000:657E ?? .block 1
0000:657F ?? .block 1
0000:6580 ?? .block 1
0000:6581 ?? .block 1
0000:6582 ?? .block 1
0000:6583 ?? .block 1
0000:6584 ?? .block 1
0000:6585 ?? .block 1
0000:6586 ?? .block 1
0000:6587 ?? .block 1
0000:6588 ?? .block 1
0000:6589 ?? .block 1
0000:658A ?? .block 1
0000:658B ?? .block 1
0000:658C ?? .block 1
0000:658D ?? .block 1
0000:658E ?? .block 1
0000:658F ?? .block 1
0000:6590 ?? .block 1
0000:6591 ?? .block 1
0000:6592 ?? .block 1
0000:6593 ?? .block 1
0000:6594 ?? .block 1
0000:6595 ?? .block 1
0000:6596 ?? .block 1
0000:6597 ?? .block 1
0000:6598 ?? .block 1
0000:6599 ?? .block 1
0000:659A ?? .block 1
0000:659B ?? .block 1
0000:659C ?? .block 1
0000:659D ?? .block 1
0000:659E ?? .block 1
0000:659F ?? .block 1
0000:65A0 ?? unk_0_65A0: .block 1 ; DATA XREF: 0000:103A|o
0000:65A0 ; sub_0_24EA+9|o ...
0000:65A1 ?? .block 1
0000:65A2 ?? .block 1
0000:65A3 ?? .block 1
0000:65A4 ?? .block 1
0000:65A5 ?? .block 1
0000:65A6 ?? .block 1
0000:65A7 ?? unk_0_65A7: .block 1 ; DATA XREF: 0000:1031|o
0000:65A8 ?? .block 1
```

```
0000:65A9 ?? .block 1
0000:65AA ?? .block 1
0000:65AB ?? .block 1
0000:65AC ?? .block 1
0000:65AD ?? .block 1
0000:65AE ?? .block 1
0000:65AF ?? .block 1
0000:65B0 ?? .block 1
0000:65B1 ?? .block 1
0000:65B2 ?? .block 1
0000:65B3 ?? .block 1
0000:65B4 ?? .block 1
0000:65B5 ?? .block 1
0000:65B6 ?? .block 1
0000:65B7 ?? .block 1
0000:65B8 ?? .block 1
0000:65B9 ?? .block 1
0000:65BA ?? .block 1
0000:65BB ?? .block 1
0000:65BC ?? .block 1
0000:65BD ?? .block 1
0000:65BE ?? .block 1
0000:65BF ?? .block 1
0000:65C0 ?? .block 1
0000:65C1 ?? .block 1
0000:65C2 ?? .block 1
0000:65C3 ?? .block 1
0000:65C4 ?? .block 1
0000:65C5 ?? .block 1
0000:65C6 ?? .block 1
0000:65C7 ?? .block 1
0000:65C8 ?? .block 1
0000:65C9 ?? .block 1
0000:65CA ?? .block 1
0000:65CB ?? .block 1
0000:65CC ?? .block 1
0000:65CD ?? .block 1
0000:65CE ?? .block 1
0000:65CF ?? .block 1
0000:65D0 ?? .block 1
0000:65D1 ?? .block 1
0000:65D2 ?? .block 1
0000:65D3 ?? .block 1
0000:65D4 ?? .block 1
0000:65D5 ?? .block 1
0000:65D6 ?? .block 1
0000:65D7 ?? .block 1
0000:65D8 ?? .block 1
0000:65D9 ?? .block 1
0000:65DA ?? .block 1
0000:65DB ?? .block 1
0000:65DC ?? .block 1
0000:65DD ?? .block 1
0000:65DE ?? .block 1
0000:65DF ?? .block 1
0000:65E0 ?? .block 1
0000:65E1 ?? .block 1
0000:65E2 ?? .block 1
0000:65E3 ?? .block 1
0000:65E4 ?? .block 1
0000:65E5 ?? .block 1
0000:65E6 ?? .block 1
0000:65E7 ?? .block 1
0000:65E8 ?? .block 1
0000:65E9 ?? .block 1
0000:65EA ?? .block 1
0000:65EB ?? .block 1
0000:65EC ?? .block 1
0000:65ED ?? .block 1
0000:65EE ?? .block 1
0000:65EF ?? .block 1
0000:65F0 ?? .block 1
0000:65F1 ?? .block 1
0000:65F2 ?? .block 1
0000:65F3 ?? .block 1
0000:65F4 ?? .block 1
0000:65F5 ?? .block 1
0000:65F6 ?? .block 1
0000:65F7 ?? .block 1
0000:65F8 ?? .block 1
0000:65F9 ?? .block 1
0000:65FA ?? .block 1
0000:65FB ?? .block 1
0000:65FC ?? .block 1
0000:65FD ?? .block 1
0000:65FE ?? .block 1
0000:65FF ?? .block 1
0000:6600 ?? unk_0_6600: .block 1 ; DATA XREF: 0000:1096!o
0000:6600 ; 0000:10CF!o ...
0000:6601 ?? .block 1
0000:6602 ?? .block 1
0000:6603 ?? unk_0_6603: .block 1 ; DATA XREF: 0000:10BA!o
0000:6604 ?? .block 1
0000:6605 ?? .block 1
0000:6606 ?? .block 1
0000:6607 ?? unk_0_6607: .block 1 ; DATA XREF: 0000:10C6!o
0000:6608 ?? .block 1
0000:6609 ?? .block 1
0000:660A ?? .block 1
0000:660B ?? .block 1
0000:660C ?? .block 1
0000:660D ?? unk_0_660D: .block 1 ; DATA XREF: 0000:10AA!o
0000:660E ?? .block 1
0000:660F ?? .block 1
0000:6610 ?? .block 1
0000:6611 ?? .block 1
0000:6612 ?? .block 1
0000:6613 ?? .block 1
0000:6614 ?? .block 1
0000:6615 ?? .block 1
0000:6616 ?? .block 1
0000:6617 ?? .block 1
0000:6618 ?? .block 1
0000:6619 ?? .block 1
0000:661A ?? .block 1
0000:661B ?? .block 1
```

```
0000:661C ?? .block 1
0000:661D ?? .block 1
0000:661E ?? .block 1
0000:661F ?? .block 1
0000:6620 ?? .block 1
0000:6621 ?? .block 1
0000:6622 ?? .block 1
0000:6623 ?? .block 1
0000:6624 ?? .block 1
0000:6625 ?? .block 1
0000:6626 ?? .block 1
0000:6627 ?? .block 1
0000:6628 ?? .block 1
0000:6629 ?? .block 1
0000:662A ?? .block 1
0000:662B ?? .block 1
0000:662C ?? .block 1
0000:662D ?? .block 1
0000:662E ?? .block 1
0000:662F ?? .block 1
0000:6630 ?? .block 1
0000:6631 ?? .block 1
0000:6632 ?? .block 1
0000:6633 ?? .block 1
0000:6634 ?? .block 1
0000:6635 ?? .block 1
0000:6636 ?? .block 1
0000:6637 ?? .block 1
0000:6638 ?? .block 1
0000:6639 ?? .block 1
0000:663A ?? .block 1
0000:663B ?? .block 1
0000:663C ?? .block 1
0000:663D ?? .block 1
0000:663E ?? .block 1
0000:663F ?? .block 1
0000:6640 ?? .block 1
0000:6641 ?? .block 1
0000:6642 ?? .block 1
0000:6643 ?? .block 1
0000:6644 ?? .block 1
0000:6645 ?? .block 1
0000:6646 ?? .block 1
0000:6647 ?? .block 1
0000:6648 ?? .block 1
0000:6649 ?? .block 1
0000:664A ?? .block 1
0000:664B ?? .block 1
0000:664C ?? .block 1
0000:664D ?? .block 1
0000:664E ?? .block 1
0000:664F ?? .block 1
0000:6650 ?? .block 1
0000:6651 ?? .block 1
0000:6652 ?? .block 1
0000:6653 ?? .block 1
0000:6654 ?? .block 1
0000:6655 ?? .block 1
0000:6656 ?? .block 1
0000:6657 ?? .block 1
0000:6658 ?? .block 1
0000:6659 ?? .block 1
0000:665A ?? .block 1
0000:665B ?? .block 1
0000:665C ?? .block 1
0000:665D ?? .block 1
0000:665E ?? .block 1
0000:665F ?? .block 1
0000:6660 ?? .block 1
0000:6661 ?? .block 1
0000:6662 ?? .block 1
0000:6663 ?? .block 1
0000:6664 ?? .block 1
0000:6665 ?? .block 1
0000:6666 ?? .block 1
0000:6667 ?? .block 1
0000:6668 ?? .block 1
0000:6669 ?? .block 1
0000:666A ?? .block 1
0000:666B ?? .block 1
0000:666C ?? .block 1
0000:666D ?? .block 1
0000:666E ?? .block 1
0000:666F ?? .block 1
0000:6670 ?? .block 1
0000:6671 ?? .block 1
0000:6672 ?? .block 1
0000:6673 ?? .block 1
0000:6674 ?? .block 1
0000:6675 ?? .block 1
0000:6676 ?? .block 1
0000:6677 ?? .block 1
0000:6678 ?? .block 1
0000:6679 ?? .block 1
0000:667A ?? .block 1
0000:667B ?? .block 1
0000:667C ?? .block 1
0000:667D ?? .block 1
0000:667E ?? .block 1
0000:667F ?? .block 1
0000:6680 ?? unk_0_6680: .block 1 ; DATA XREF: sub_0_11A6+15!o
; sub_0_281D+5!o ...
0000:6681 ?? .block 1
0000:6682 ?? .block 1
0000:6683 ?? unk_0_6683: .block 1 ; DATA XREF: sub_0_11A6!o
0000:6684 ?? .block 1
0000:6685 ?? .block 1
0000:6686 ?? .block 1
0000:6687 ?? unk_0_6687: .block 1 ; DATA XREF: sub_0_11A6+C!o
0000:6688 ?? .block 1
0000:6689 ?? .block 1
0000:668A ?? .block 1
0000:668B ?? .block 1
0000:668C ?? .block 1
0000:668D ?? .block 1
0000:668E ?? .block 1
```

```

0000:668F ??                .block 1
0000:6690 ??                .block 1
unk_0_6690:                ; DATA XREF: sub_0_2ED4+15↑o
0000:6691 ??                .block 1
0000:6692 ??                .block 1
0000:6693 ??                .block 1
0000:6694 ??                .block 1
0000:6695 ??                .block 1
0000:6696 ??                .block 1
0000:6697 ??                .block 1
0000:6698 ??                .block 1
0000:6699 ??                .block 1
0000:669A ??                .block 1
0000:669B ??                .block 1
0000:669C ??                .block 1
0000:669D ??                .block 1
0000:669E ??                .block 1
0000:669F ??                .block 1
0000:66A0 ??                .block 1
unk_0_66A0:                ; DATA XREF: sub_0_3A2+1A↑o
                                ; sub_0_11FA↑o ...
0000:66A1 ??                .block 1
0000:66A2 ??                .block 1
0000:66A3 ??                .block 1
0000:66A4 ??                .block 1
0000:66A5 ??                .block 1
0000:66A6 ??                .block 1
0000:66A7 ??                .block 1
0000:66A8 ??                .block 1
0000:66A9 ??                .block 1
0000:66AA ??                .block 1
0000:66AB ??                .block 1
0000:66AC ??                .block 1
0000:66AD ??                .block 1
0000:66AE ??                .block 1
0000:66AF ??                .block 1
0000:66B0 ??                .block 1
0000:66B1 ??                .block 1
0000:66B2 ??                .block 1
0000:66B3 ??                .block 1
0000:66B4 ??                .block 1
0000:66B5 ??                .block 1
0000:66B6 ??                .block 1
0000:66B7 ??                .block 1
0000:66B8 ??                .block 1
0000:66B9 ??                .block 1
0000:66BA ??                .block 1
0000:66BB ??                .block 1
0000:66BC ??                .block 1
0000:66BD ??                .block 1
0000:66BE ??                .block 1
0000:66BF ??                .block 1
0000:66C0 ??                .block 1
0000:66C1 ??                .block 1
0000:66C2 ??                .block 1
0000:66C3 ??                .block 1
0000:66C4 ??                .block 1
0000:66C5 ??                .block 1
0000:66C6 ??                .block 1
0000:66C7 ??                .block 1
0000:66C8 ??                .block 1
0000:66C9 ??                .block 1
0000:66CA ??                .block 1
0000:66CB ??                .block 1
0000:66CC ??                .block 1
0000:66CD ??                .block 1
0000:66CE ??                .block 1
0000:66CF ??                .block 1
0000:66D0 ??                .block 1
0000:66D1 ??                .block 1
0000:66D2 ??                .block 1
0000:66D3 ??                .block 1
0000:66D4 ??                .block 1
0000:66D5 ??                .block 1
0000:66D6 ??                .block 1
0000:66D7 ??                .block 1
0000:66D8 ??                .block 1
0000:66D9 ??                .block 1
0000:66DA ??                .block 1
0000:66DB ??                .block 1
0000:66DC ??                .block 1
0000:66DD ??                .block 1
0000:66DE ??                .block 1
0000:66DF ??                .block 1
0000:66E0 ??                .block 1
0000:66E1 ??                .block 1
0000:66E2 ??                .block 1
0000:66E3 ??                .block 1
0000:66E4 ??                .block 1
0000:66E5 ??                .block 1
0000:66E6 ??                .block 1
0000:66E7 ??                .block 1
0000:66E8 ??                .block 1
0000:66E9 ??                .block 1
0000:66EA ??                .block 1
0000:66EB ??                .block 1
0000:66EC ??                .block 1
0000:66ED ??                .block 1
0000:66EE ??                .block 1
0000:66EF ??                .block 1
0000:66F0 ??                .block 1
0000:66F1 ??                .block 1
0000:66F2 ??                .block 1
0000:66F3 ??                .block 1
0000:66F4 ??                .block 1
0000:66F5 ??                .block 1
0000:66F6 ??                .block 1
0000:66F7 ??                .block 1
0000:66F8 ??                .block 1
0000:66F9 ??                .block 1
0000:66FA ??                .block 1
0000:66FB ??                .block 1
0000:66FC ??                .block 1
0000:66FD ??                .block 1
0000:66FE ??                .block 1
0000:66FF ??                .block 1
unk_0_6700:                ; DATA XREF: sub_0_1F72+5↑o
                                ; sub_0_286F+1B↑o ...
0000:6700 ??                .block 1

```

0000:6701	??	.block	1
0000:6702	??	.block	1
0000:6703	??	.block	1
0000:6704	??	.block	1
0000:6705	??	.block	1
0000:6706	??	.block	1
0000:6707	unk_0_6707:	.block	1
0000:6708		.block	1
0000:6709		.block	1
0000:670A		.block	1
0000:670B		.block	1
0000:670C		.block	1
0000:670D		.block	1
0000:670E		.block	1
0000:670F		.block	1
0000:6710		.block	1
0000:6711		.block	1
0000:6712		.block	1
0000:6713		.block	1
0000:6714		.block	1
0000:6715		.block	1
0000:6716		.block	1
0000:6717		.block	1
0000:6718		.block	1
0000:6719		.block	1
0000:671A		.block	1
0000:671B		.block	1
0000:671C		.block	1
0000:671D		.block	1
0000:671E		.block	1
0000:671F		.block	1
0000:6720		.block	1
0000:6721		.block	1
0000:6722		.block	1
0000:6723		.block	1
0000:6724		.block	1
0000:6725		.block	1
0000:6726		.block	1
0000:6727		.block	1
0000:6728		.block	1
0000:6729		.block	1
0000:672A		.block	1
0000:672B		.block	1
0000:672C		.block	1
0000:672D		.block	1
0000:672E		.block	1
0000:672F		.block	1
0000:6730		.block	1
0000:6731		.block	1
0000:6732		.block	1
0000:6733		.block	1
0000:6734		.block	1
0000:6735		.block	1
0000:6736		.block	1
0000:6737		.block	1
0000:6738		.block	1
0000:6739		.block	1
0000:673A		.block	1
0000:673B		.block	1
0000:673C		.block	1
0000:673D		.block	1
0000:673E		.block	1
0000:673F		.block	1
0000:6740		.block	1
0000:6741		.block	1
0000:6742		.block	1
0000:6743		.block	1
0000:6744		.block	1
0000:6745		.block	1
0000:6746		.block	1
0000:6747		.block	1
0000:6748		.block	1
0000:6749		.block	1
0000:674A		.block	1
0000:674B		.block	1
0000:674C		.block	1
0000:674D		.block	1
0000:674E		.block	1
0000:674F		.block	1
0000:6750		.block	1
0000:6751		.block	1
0000:6752		.block	1
0000:6753		.block	1
0000:6754		.block	1
0000:6755		.block	1
0000:6756		.block	1
0000:6757		.block	1
0000:6758		.block	1
0000:6759		.block	1
0000:675A		.block	1
0000:675B		.block	1
0000:675C		.block	1
0000:675D		.block	1
0000:675E		.block	1
0000:675F		.block	1
0000:6760		.block	1
0000:6761		.block	1
0000:6762		.block	1
0000:6763		.block	1
0000:6764		.block	1
0000:6765		.block	1
0000:6766		.block	1
0000:6767		.block	1
0000:6768		.block	1
0000:6769		.block	1
0000:676A		.block	1
0000:676B		.block	1
0000:676C		.block	1
0000:676D		.block	1
0000:676E		.block	1
0000:676F		.block	1
0000:6770		.block	1
0000:6771		.block	1
0000:6772		.block	1
0000:6773		.block	1
0000:6774		.block	1

; DATA XREF: 0000:1009|o

0000:6775	??	.block	1
0000:6776	??	.block	1
0000:6777	??	.block	1
0000:6778	??	.block	1
0000:6779	??	.block	1
0000:677A	??	.block	1
0000:677B	??	.block	1
0000:677C	??	.block	1
0000:677D	??	.block	1
0000:677E	??	.block	1
0000:677F	??	.block	1
0000:6780	??	.block	1
0000:6781	??	.block	1
0000:6782	??	.block	1
0000:6783	??	.block	1
0000:6784	??	.block	1
0000:6785	??	.block	1
0000:6786	??	.block	1
0000:6787	??	.block	1
0000:6788	??	.block	1
0000:6789	??	.block	1
0000:678A	??	.block	1
0000:678B	??	.block	1
0000:678C	??	.block	1
0000:678D	??	.block	1
0000:678E	??	.block	1
0000:678F	??	.block	1
0000:6790	??	.block	1
0000:6791	??	.block	1
0000:6792	??	.block	1
0000:6793	??	.block	1
0000:6794	??	.block	1
0000:6795	??	.block	1
0000:6796	??	.block	1
0000:6797	??	.block	1
0000:6798	??	.block	1
0000:6799	??	.block	1
0000:679A	??	.block	1
0000:679B	??	.block	1
0000:679C	??	.block	1
0000:679D	??	.block	1
0000:679E	??	.block	1
0000:679F	??	.block	1
0000:67A0	??	.block	1
0000:67A1	??	.block	1
0000:67A2	??	.block	1
0000:67A3	??	.block	1
0000:67A4	??	.block	1
0000:67A5	??	.block	1
0000:67A6	??	.block	1
0000:67A7	??	.block	1
0000:67A8	??	.block	1
0000:67A9	??	.block	1
0000:67AA	??	.block	1
0000:67AB	??	.block	1
0000:67AC	??	.block	1
0000:67AD	??	.block	1
0000:67AE	??	.block	1
0000:67AF	??	.block	1
0000:67B0	??	.block	1
0000:67B1	??	.block	1
0000:67B2	??	.block	1
0000:67B3	??	.block	1
0000:67B4	??	.block	1
0000:67B5	??	.block	1
0000:67B6	??	.block	1
0000:67B7	??	.block	1
0000:67B8	??	.block	1
0000:67B9	??	.block	1
0000:67BA	??	.block	1
0000:67BB	??	.block	1
0000:67BC	??	.block	1
0000:67BD	??	.block	1
0000:67BE	??	.block	1
0000:67BF	??	.block	1
0000:67C0	??	.block	1
0000:67C1	??	.block	1
0000:67C2	??	.block	1
0000:67C3	??	.block	1
0000:67C4	??	.block	1
0000:67C5	??	.block	1
0000:67C6	??	.block	1
0000:67C7	??	.block	1
0000:67C8	??	.block	1
0000:67C9	??	.block	1
0000:67CA	??	.block	1
0000:67CB	??	.block	1
0000:67CC	??	.block	1
0000:67CD	??	.block	1
0000:67CE	??	.block	1
0000:67CF	??	.block	1
0000:67D0	??	.block	1
0000:67D1	??	.block	1
0000:67D2	??	.block	1
0000:67D3	??	.block	1
0000:67D4	??	.block	1
0000:67D5	??	.block	1
0000:67D6	??	.block	1
0000:67D7	??	.block	1
0000:67D8	??	.block	1
0000:67D9	??	.block	1
0000:67DA	??	.block	1
0000:67DB	??	.block	1
0000:67DC	??	.block	1
0000:67DD	??	.block	1
0000:67DE	??	.block	1
0000:67DF	??	.block	1
0000:67E0	??	.block	1
0000:67E1	??	.block	1
0000:67E2	??	.block	1
0000:67E3	??	.block	1
0000:67E4	??	.block	1
0000:67E5	??	.block	1
0000:67E6	??	.block	1
0000:67E7	??	.block	1
0000:67E8	??	.block	1

0000:67E9	??	.block	1
0000:67EA	??	.block	1
0000:67EB	??	.block	1
0000:67EC	??	.block	1
0000:67ED	??	.block	1
0000:67EE	??	.block	1
0000:67EF	??	.block	1
0000:67F0	??	.block	1
0000:67F1	??	.block	1
0000:67F2	??	.block	1
0000:67F3	??	.block	1
0000:67F4	??	.block	1
0000:67F5	??	.block	1
0000:67F6	??	.block	1
0000:67F7	??	.block	1
0000:67F8	??	.block	1
0000:67F9	??	.block	1
0000:67FA	??	.block	1
0000:67FB	??	.block	1
0000:67FC	??	.block	1
0000:67FD	??	.block	1
0000:67FE	??	.block	1
0000:67FF	??	.block	1
0000:6800	??	.block	1
0000:6801	??	.block	1
0000:6802	??	.block	1
0000:6803	??	.block	1
0000:6804	??	.block	1
0000:6805	??	.block	1
0000:6806	??	.block	1
0000:6807	unk_0_6807:	.block	1
0000:6808	??	.block	1
0000:6809	??	.block	1
0000:680A	??	.block	1
0000:680B	??	.block	1
0000:680C	??	.block	1
0000:680D	??	.block	1
0000:680E	??	.block	1
0000:680F	??	.block	1
0000:6810	??	.block	1
0000:6811	??	.block	1
0000:6812	??	.block	1
0000:6813	??	.block	1
0000:6814	??	.block	1
0000:6815	??	.block	1
0000:6816	??	.block	1
0000:6817	??	.block	1
0000:6818	??	.block	1
0000:6819	??	.block	1
0000:681A	??	.block	1
0000:681B	??	.block	1
0000:681C	??	.block	1
0000:681D	??	.block	1
0000:681E	??	.block	1
0000:681F	??	.block	1
0000:6820	??	.block	1
0000:6821	??	.block	1
0000:6822	??	.block	1
0000:6823	??	.block	1
0000:6824	??	.block	1
0000:6825	??	.block	1
0000:6826	??	.block	1
0000:6827	??	.block	1
0000:6828	??	.block	1
0000:6829	??	.block	1
0000:682A	??	.block	1
0000:682B	??	.block	1
0000:682C	??	.block	1
0000:682D	??	.block	1
0000:682E	??	.block	1
0000:682F	??	.block	1
0000:6830	??	.block	1
0000:6831	??	.block	1
0000:6832	??	.block	1
0000:6833	??	.block	1
0000:6834	??	.block	1
0000:6835	??	.block	1
0000:6836	??	.block	1
0000:6837	??	.block	1
0000:6838	??	.block	1
0000:6839	??	.block	1
0000:683A	??	.block	1
0000:683B	??	.block	1
0000:683C	??	.block	1
0000:683D	??	.block	1
0000:683E	??	.block	1
0000:683F	??	.block	1
0000:6840	??	.block	1
0000:6841	??	.block	1
0000:6842	??	.block	1
0000:6843	??	.block	1
0000:6844	??	.block	1
0000:6845	??	.block	1
0000:6846	??	.block	1
0000:6847	??	.block	1
0000:6848	??	.block	1
0000:6849	??	.block	1
0000:684A	??	.block	1
0000:684B	??	.block	1
0000:684C	??	.block	1
0000:684D	??	.block	1
0000:684E	??	.block	1
0000:684F	??	.block	1
0000:6850	??	.block	1
0000:6851	??	.block	1
0000:6852	??	.block	1
0000:6853	??	.block	1
0000:6854	??	.block	1
0000:6855	??	.block	1
0000:6856	??	.block	1
0000:6857	??	.block	1
0000:6858	??	.block	1
0000:6859	??	.block	1
0000:685A	??	.block	1
0000:685B	??	.block	1
0000:685C	??	.block	1

; DATA XREF: 0000:1012|o

```
0000:685D ?? .block 1
0000:685E ?? .block 1
0000:685F ?? .block 1
0000:6860 ?? .block 1
0000:6861 ?? .block 1
0000:6862 ?? .block 1
0000:6863 ?? .block 1
0000:6864 ?? .block 1
0000:6865 ?? .block 1
0000:6866 ?? .block 1
0000:6867 ?? .block 1
0000:6868 ?? .block 1
0000:6869 ?? .block 1
0000:686A ?? .block 1
0000:686B ?? .block 1
0000:686C ?? .block 1
0000:686D ?? .block 1
0000:686E ?? .block 1
0000:686F ?? .block 1
0000:6870 ?? .block 1
0000:6871 ?? .block 1
0000:6872 ?? .block 1
0000:6873 ?? .block 1
0000:6874 ?? .block 1
0000:6875 ?? .block 1
0000:6876 ?? .block 1
0000:6877 ?? .block 1
0000:6878 ?? .block 1
0000:6879 ?? .block 1
0000:687A ?? .block 1
0000:687B ?? .block 1
0000:687C ?? .block 1
0000:687D ?? .block 1
0000:687E ?? .block 1
0000:687F ?? .block 1
0000:6880 ?? .block 1
0000:6881 ?? .block 1
0000:6882 ?? .block 1
0000:6883 ?? .block 1
0000:6884 ?? .block 1
0000:6885 ?? .block 1
0000:6886 ?? .block 1
0000:6887 ?? .block 1
0000:6888 ?? .block 1
0000:6889 ?? .block 1
0000:688A ?? .block 1
0000:688B ?? .block 1
0000:688C ?? .block 1
0000:688D ?? .block 1
0000:688E ?? .block 1
0000:688F ?? .block 1
0000:6890 ?? .block 1
0000:6891 ?? .block 1
0000:6892 ?? .block 1
0000:6893 ?? .block 1
0000:6894 ?? .block 1
0000:6895 ?? .block 1
0000:6896 ?? .block 1
0000:6897 ?? .block 1
0000:6898 ?? .block 1
0000:6899 ?? .block 1
0000:689A ?? .block 1
0000:689B ?? .block 1
0000:689C ?? .block 1
0000:689D ?? .block 1
0000:689E ?? .block 1
0000:689F ?? .block 1
0000:68A0 ?? .block 1
0000:68A1 ?? .block 1
0000:68A2 ?? .block 1
0000:68A3 ?? .block 1
0000:68A4 ?? .block 1
0000:68A5 ?? .block 1
0000:68A6 ?? .block 1
0000:68A7 ?? .block 1
0000:68A8 ?? .block 1
0000:68A9 ?? .block 1
0000:68AA ?? .block 1
0000:68AB ?? .block 1
0000:68AC ?? .block 1
0000:68AD ?? .block 1
0000:68AE ?? .block 1
0000:68AF ?? .block 1
0000:68B0 ?? .block 1
0000:68B1 ?? .block 1
0000:68B2 ?? .block 1
0000:68B3 ?? .block 1
0000:68B4 ?? .block 1
0000:68B5 ?? .block 1
0000:68B6 ?? .block 1
0000:68B7 ?? .block 1
0000:68B8 ?? .block 1
0000:68B9 ?? .block 1
0000:68BA ?? .block 1
0000:68BB ?? .block 1
0000:68BC ?? .block 1
0000:68BD ?? .block 1
0000:68BE ?? .block 1
0000:68BF ?? .block 1
0000:68C0 ?? .block 1
0000:68C1 ?? .block 1
0000:68C2 ?? .block 1
0000:68C3 ?? .block 1
0000:68C4 ?? .block 1
0000:68C5 ?? .block 1
0000:68C6 ?? .block 1
0000:68C7 ?? .block 1
0000:68C8 ?? .block 1
0000:68C9 ?? .block 1
0000:68CA ?? .block 1
0000:68CB ?? .block 1
0000:68CC ?? .block 1
0000:68CD ?? .block 1
0000:68CE ?? .block 1
0000:68CF ?? .block 1
0000:68D0 ?? .block 1
```



```

0000:68D1 ?? .block 1
0000:68D2 ?? .block 1
0000:68D3 ?? .block 1
0000:68D4 ?? .block 1
0000:68D5 ?? .block 1
0000:68D6 ?? .block 1
0000:68D7 ?? .block 1
0000:68D8 ?? .block 1
0000:68D9 ?? .block 1
0000:68DA ?? .block 1
0000:68DB ?? .block 1
0000:68DC ?? .block 1
0000:68DD ?? .block 1
0000:68DE ?? .block 1
0000:68DF ?? .block 1
0000:68E0 ?? .block 1
0000:68E1 ?? .block 1
0000:68E2 ?? .block 1
0000:68E3 ?? .block 1
0000:68E4 ?? .block 1
0000:68E5 ?? .block 1
0000:68E6 ?? .block 1
0000:68E7 ?? .block 1
0000:68E8 ?? .block 1
0000:68E9 ?? .block 1
0000:68EA ?? .block 1
0000:68EB ?? .block 1
0000:68EC ?? .block 1
0000:68ED ?? .block 1
0000:68EE ?? .block 1
0000:68EF ?? .block 1
0000:68F0 ?? .block 1
0000:68F1 ?? .block 1
0000:68F2 ?? .block 1
0000:68F3 ?? .block 1
0000:68F4 ?? .block 1
0000:68F5 ?? .block 1
0000:68F6 ?? .block 1
0000:68F7 ?? .block 1
0000:68F8 ?? .block 1
0000:68F9 ?? .block 1
0000:68FA ?? .block 1
0000:68FB ?? .block 1
0000:68FC ?? .block 1
0000:68FD ?? .block 1
0000:68FE ?? .block 1
0000:68FF ?? .block 1
0000:6900 ?? ?? ?? ??+soft_sprite_ram:.block 180h
0000:6900 ?? ?? ?? ??+ ; DATA XREF: 0000:0139|o
0000:6900 ?? ?? ?? ??+ ; clear_tiles_and_sprites+11|o ...
0000:6900 ?? ?? ?? ??+ ; 0- 1 = pauline
0000:6900 ?? ?? ?? ??+ ; 2-11 = kong
0000:6900 ?? ?? ?? ??+ ; 12-
0000:6900 ?? ?? ?? ??+ ; 19 = mario
0000:6A80 ?? .block 1
0000:6A81 ?? .block 1
0000:6A82 ?? .block 1
0000:6A83 ?? .block 1
0000:6A84 ?? .block 1
0000:6A85 ?? .block 1
0000:6A86 ?? .block 1
0000:6A87 ?? .block 1
0000:6A88 ?? .block 1
0000:6A89 ?? .block 1
0000:6A8A ?? .block 1
0000:6A8B ?? .block 1
0000:6A8C ?? .block 1
0000:6A8D ?? .block 1
0000:6A8E ?? .block 1
0000:6A8F ?? .block 1
0000:6A90 ?? .block 1
0000:6A91 ?? .block 1
0000:6A92 ?? .block 1
0000:6A93 ?? .block 1
0000:6A94 ?? .block 1
0000:6A95 ?? .block 1
0000:6A96 ?? .block 1
0000:6A97 ?? .block 1
0000:6A98 ?? .block 1
0000:6A99 ?? .block 1
0000:6A9A ?? .block 1
0000:6A9B ?? .block 1
0000:6A9C ?? .block 1
0000:6A9D ?? .block 1
0000:6A9E ?? .block 1
0000:6A9F ?? .block 1
0000:6AA0 ?? .block 1
0000:6AA1 ?? .block 1
0000:6AA2 ?? .block 1
0000:6AA3 ?? .block 1
0000:6AA4 ?? .block 1
0000:6AA5 ?? .block 1
0000:6AA6 ?? .block 1
0000:6AA7 ?? .block 1
0000:6AA8 ?? .block 1
0000:6AA9 ?? .block 1
0000:6AAA ?? .block 1
0000:6AAB ?? .block 1
0000:6AAC ?? .block 1
0000:6AAD ?? .block 1
0000:6AAE ?? .block 1
0000:6AAF ?? .block 1
0000:6AB0 ?? .block 1
0000:6AB1 ?? .block 1
0000:6AB2 ?? .block 1
0000:6AB3 ?? .block 1
0000:6AB4 ?? .block 1
0000:6AB5 ?? .block 1
0000:6AB6 ?? .block 1
0000:6AB7 ?? .block 1
0000:6AB8 ?? .block 1
0000:6AB9 ?? .block 1
0000:6ABA ?? .block 1
0000:6ABB ?? .block 1
0000:6ABC ?? .block 1
0000:6ABD ?? .block 1
0000:6ABE ?? .block 1

```

0000:6ABF	??	.block	1
0000:6AC0	??	.block	1
0000:6AC1	??	.block	1
0000:6AC2	??	.block	1
0000:6AC3	??	.block	1
0000:6AC4	??	.block	1
0000:6AC5	??	.block	1
0000:6AC6	??	.block	1
0000:6AC7	??	.block	1
0000:6AC8	??	.block	1
0000:6AC9	??	.block	1
0000:6ACA	??	.block	1
0000:6ACB	??	.block	1
0000:6ACC	??	.block	1
0000:6ACD	??	.block	1
0000:6ACE	??	.block	1
0000:6ACF	??	.block	1
0000:6AD0	??	.block	1
0000:6AD1	??	.block	1
0000:6AD2	??	.block	1
0000:6AD3	??	.block	1
0000:6AD4	??	.block	1
0000:6AD5	??	.block	1
0000:6AD6	??	.block	1
0000:6AD7	??	.block	1
0000:6AD8	??	.block	1
0000:6AD9	??	.block	1
0000:6ADA	??	.block	1
0000:6ADB	??	.block	1
0000:6ADC	??	.block	1
0000:6ADD	??	.block	1
0000:6ADE	??	.block	1
0000:6ADF	??	.block	1
0000:6AE0	??	.block	1
0000:6AE1	??	.block	1
0000:6AE2	??	.block	1
0000:6AE3	??	.block	1
0000:6AE4	??	.block	1
0000:6AE5	??	.block	1
0000:6AE6	??	.block	1
0000:6AE7	??	.block	1
0000:6AE8	??	.block	1
0000:6AE9	??	.block	1
0000:6AEA	??	.block	1
0000:6AEB	??	.block	1
0000:6AEC	??	.block	1
0000:6AED	??	.block	1
0000:6AEE	??	.block	1
0000:6AEF	??	.block	1
0000:6AF0	??	.block	1
0000:6AF1	??	.block	1
0000:6AF2	??	.block	1
0000:6AF3	??	.block	1
0000:6AF4	??	.block	1
0000:6AF5	??	.block	1
0000:6AF6	??	.block	1
0000:6AF7	??	.block	1
0000:6AF8	??	.block	1
0000:6AF9	??	.block	1
0000:6AFA	??	.block	1
0000:6AFB	??	.block	1
0000:6AFC	??	.block	1
0000:6AFD	??	.block	1
0000:6AFE	??	.block	1
0000:6AFF	??	.block	1
0000:6B00	??	.block	1
0000:6B01	??	.block	1
0000:6B02	??	.block	1
0000:6B03	??	.block	1
0000:6B04	??	.block	1
0000:6B05	??	.block	1
0000:6B06	??	.block	1
0000:6B07	??	.block	1
0000:6B08	??	.block	1
0000:6B09	??	.block	1
0000:6B0A	??	.block	1
0000:6B0B	??	.block	1
0000:6B0C	??	.block	1
0000:6B0D	??	.block	1
0000:6B0E	??	.block	1
0000:6B0F	??	.block	1
0000:6B10	??	.block	1
0000:6B11	??	.block	1
0000:6B12	??	.block	1
0000:6B13	??	.block	1
0000:6B14	??	.block	1
0000:6B15	??	.block	1
0000:6B16	??	.block	1
0000:6B17	??	.block	1
0000:6B18	??	.block	1
0000:6B19	??	.block	1
0000:6B1A	??	.block	1
0000:6B1B	??	.block	1
0000:6B1C	??	.block	1
0000:6B1D	??	.block	1
0000:6B1E	??	.block	1
0000:6B1F	??	.block	1
0000:6B20	??	.block	1
0000:6B21	??	.block	1
0000:6B22	??	.block	1
0000:6B23	??	.block	1
0000:6B24	??	.block	1
0000:6B25	??	.block	1
0000:6B26	??	.block	1
0000:6B27	??	.block	1
0000:6B28	??	.block	1
0000:6B29	??	.block	1
0000:6B2A	??	.block	1
0000:6B2B	??	.block	1
0000:6B2C	??	.block	1
0000:6B2D	??	.block	1
0000:6B2E	??	.block	1
0000:6B2F	??	.block	1
0000:6B30	??	.block	1
0000:6B31	??	.block	1
0000:6B32	??	.block	1

0000:6B33	??	.block	1
0000:6B34	??	.block	1
0000:6B35	??	.block	1
0000:6B36	??	.block	1
0000:6B37	??	.block	1
0000:6B38	??	.block	1
0000:6B39	??	.block	1
0000:6B3A	??	.block	1
0000:6B3B	??	.block	1
0000:6B3C	??	.block	1
0000:6B3D	??	.block	1
0000:6B3E	??	.block	1
0000:6B3F	??	.block	1
0000:6B40	??	.block	1
0000:6B41	??	.block	1
0000:6B42	??	.block	1
0000:6B43	??	.block	1
0000:6B44	??	.block	1
0000:6B45	??	.block	1
0000:6B46	??	.block	1
0000:6B47	??	.block	1
0000:6B48	??	.block	1
0000:6B49	??	.block	1
0000:6B4A	??	.block	1
0000:6B4B	??	.block	1
0000:6B4C	??	.block	1
0000:6B4D	??	.block	1
0000:6B4E	??	.block	1
0000:6B4F	??	.block	1
0000:6B50	??	.block	1
0000:6B51	??	.block	1
0000:6B52	??	.block	1
0000:6B53	??	.block	1
0000:6B54	??	.block	1
0000:6B55	??	.block	1
0000:6B56	??	.block	1
0000:6B57	??	.block	1
0000:6B58	??	.block	1
0000:6B59	??	.block	1
0000:6B5A	??	.block	1
0000:6B5B	??	.block	1
0000:6B5C	??	.block	1
0000:6B5D	??	.block	1
0000:6B5E	??	.block	1
0000:6B5F	??	.block	1
0000:6B60	??	.block	1
0000:6B61	??	.block	1
0000:6B62	??	.block	1
0000:6B63	??	.block	1
0000:6B64	??	.block	1
0000:6B65	??	.block	1
0000:6B66	??	.block	1
0000:6B67	??	.block	1
0000:6B68	??	.block	1
0000:6B69	??	.block	1
0000:6B6A	??	.block	1
0000:6B6B	??	.block	1
0000:6B6C	??	.block	1
0000:6B6D	??	.block	1
0000:6B6E	??	.block	1
0000:6B6F	??	.block	1
0000:6B70	??	.block	1
0000:6B71	??	.block	1
0000:6B72	??	.block	1
0000:6B73	??	.block	1
0000:6B74	??	.block	1
0000:6B75	??	.block	1
0000:6B76	??	.block	1
0000:6B77	??	.block	1
0000:6B78	??	.block	1
0000:6B79	??	.block	1
0000:6B7A	??	.block	1
0000:6B7B	??	.block	1
0000:6B7C	??	.block	1
0000:6B7D	??	.block	1
0000:6B7E	??	.block	1
0000:6B7F	??	.block	1
0000:6B80	??	.block	1
0000:6B81	??	.block	1
0000:6B82	??	.block	1
0000:6B83	??	.block	1
0000:6B84	??	.block	1
0000:6B85	??	.block	1
0000:6B86	??	.block	1
0000:6B87	??	.block	1
0000:6B88	??	.block	1
0000:6B89	??	.block	1
0000:6B8A	??	.block	1
0000:6B8B	??	.block	1
0000:6B8C	??	.block	1
0000:6B8D	??	.block	1
0000:6B8E	??	.block	1
0000:6B8F	??	.block	1
0000:6B90	??	.block	1
0000:6B91	??	.block	1
0000:6B92	??	.block	1
0000:6B93	??	.block	1
0000:6B94	??	.block	1
0000:6B95	??	.block	1
0000:6B96	??	.block	1
0000:6B97	??	.block	1
0000:6B98	??	.block	1
0000:6B99	??	.block	1
0000:6B9A	??	.block	1
0000:6B9B	??	.block	1
0000:6B9C	??	.block	1
0000:6B9D	??	.block	1
0000:6B9E	??	.block	1
0000:6B9F	??	.block	1
0000:6BA0	??	.block	1
0000:6BA1	??	.block	1
0000:6BA2	??	.block	1
0000:6BA3	??	.block	1
0000:6BA4	??	.block	1
0000:6BA5	??	.block	1
0000:6BA6	??	.block	1

```

0000:6BA7 ?? .block 1
0000:6BA8 ?? .block 1
0000:6BA9 ?? .block 1
0000:6BAA ?? .block 1
0000:6BAB ?? .block 1
0000:6BAC ?? .block 1
0000:6BAD ?? .block 1
0000:6BAE ?? .block 1
0000:6BAF ?? .block 1
0000:6BB0 ?? .block 1
0000:6BB1 ?? .block 1
0000:6BB2 ?? .block 1
0000:6BB3 ?? .block 1
0000:6BB4 ?? .block 1
0000:6BB5 ?? .block 1
0000:6BB6 ?? .block 1
0000:6BB7 ?? .block 1
0000:6BB8 ?? .block 1
0000:6BB9 ?? .block 1
0000:6BBA ?? .block 1
0000:6BBB ?? .block 1
0000:6BBC ?? .block 1
0000:6BBD ?? .block 1
0000:6BBE ?? .block 1
0000:6BBF ?? .block 1
0000:6BC0 ?? .block 1
0000:6BC1 ?? .block 1
0000:6BC2 ?? .block 1
0000:6BC3 ?? .block 1
0000:6BC4 ?? .block 1
0000:6BC5 ?? .block 1
0000:6BC6 ?? .block 1
0000:6BC7 ?? .block 1
0000:6BC8 ?? .block 1
0000:6BC9 ?? .block 1
0000:6BCA ?? .block 1
0000:6BCB ?? .block 1
0000:6BCC ?? .block 1
0000:6BCD ?? .block 1
0000:6BCE ?? .block 1
0000:6BCF ?? .block 1
0000:6BD0 ?? .block 1
0000:6BD1 ?? .block 1
0000:6BD2 ?? .block 1
0000:6BD3 ?? .block 1
0000:6BD4 ?? .block 1
0000:6BD5 ?? .block 1
0000:6BD6 ?? .block 1
0000:6BD7 ?? .block 1
0000:6BD8 ?? .block 1
0000:6BD9 ?? .block 1
0000:6BDA ?? .block 1
0000:6BDB ?? .block 1
0000:6BDC ?? .block 1
0000:6BDD ?? .block 1
0000:6BDE ?? .block 1
0000:6BDF ?? .block 1
0000:6BE0 ?? .block 1
0000:6BE1 ?? .block 1
0000:6BE2 ?? .block 1
0000:6BE3 ?? .block 1
0000:6BE4 ?? .block 1
0000:6BE5 ?? .block 1
0000:6BE6 ?? .block 1
0000:6BE7 ?? .block 1
0000:6BE8 ?? .block 1
0000:6BE9 ?? .block 1
0000:6BEA ?? .block 1
0000:6BEB ?? .block 1
0000:6BEC ?? .block 1
0000:6BED ?? .block 1
0000:6BEE ?? .block 1
0000:6BEF ?? .block 1
0000:6BF0 ?? .block 1
0000:6BF1 ?? .block 1
0000:6BF2 ?? .block 1
0000:6BF3 ?? .block 1
0000:6BF4 ?? .block 1
0000:6BF5 ?? .block 1
0000:6BF6 ?? .block 1
0000:6BF7 ?? .block 1
0000:6BF8 ?? .block 1
0000:6BF9 ?? .block 1
0000:6BFA ?? .block 1
0000:6BFB ?? .block 1
0000:6BFC ?? .block 1
0000:6BFD ?? .block 1
0000:6BFE ?? .block 1
0000:6BFF ?? .block 1
0000:6BFF ; end of 'RAM'
0000:6BFF ; =====
0000:7000 ; Segment type: Regular
0000:7000 ; segment 'SPRAM'
0000:7000 .org 7000h
0000:7000 ?? ?? ?? ??+SPRAM_start: .block 400h ; DATA XREF: 0000:013D!o
0000:7000 ?? ?? ?? ??+ ; 0000:0276!o
0000:7000 ?? ?? ?? ??+; end of 'SPRAM' ; 2 banks of 128 sprites
0000:7000 ?? ?? ?? ??+ ; - only 16 displayed per scanline
0000:7000 ?? ?? ?? ??+ ; @0 7:0=y
0000:7000 ?? ?? ?? ??+ ; @1 7=flipy,6:0=code
0000:7000 ?? ?? ?? ??+ ; @2 7=flipx,3:0=colour
0000:7000 ?? ?? ?? ??+ ; @3 7:0=x
0000:7400 ; =====
0000:7400 ; Segment type: Regular
0000:7400 ; segment 'VRAM'
0000:7400 .org 7400h
0000:7400 ?? ?? ?? ??+VRAM_start: .block 400h ; DATA XREF: 0000:0285!o
0000:7400 ?? ?? ?? ??+ ; clear_tiles_and_sprites!o ...
0000:7400 ?? ?? ?? ??+; end of 'VRAM'
0000:7400 ?? ?? ?? ??+
0000:7800 ; =====
0000:7800 ; Segment type: Regular

```

```

0000:7800          ; segment 'I8257'
0000:7800          .org 7800h
0000:7800 ?? ?? ?? ??+i8257_io: .block 10h
0000:7800 ?? ?? ?? ??+; end of 'I8257'
0000:7800 ?? ?? ?? ??+
0000:7C00          ; =====
0000:7C00
0000:7C00          ; Segment type: Regular
0000:7C00          ; segment 'IN0'
0000:7C00          .org 7C00h
0000:7C00 ??      in0: .block 1
0000:7C00          ; end of 'IN0'
0000:7C00
0000:7C80          ; =====
0000:7C80
0000:7C80          ; Segment type: Regular
0000:7C80          ; segment 'IN1'
0000:7C80          .org 7C80h
0000:7C80 ??      in1: .block 1
0000:7C80          ; end of 'IN1'
0000:7C80
0000:7D00          ; =====
0000:7D00
0000:7D00          ; Segment type: Regular
0000:7D00          ; segment 'IO'
0000:7D00          .org 7D00h
0000:7D00 ??      in2_snd_latch: .block 1
0000:7D00          ; DATA XREF: 0000:0072|r
0000:7D01 ??      .block 1
0000:7D02 ??      .block 1
0000:7D03 ??      .block 1
0000:7D04 ??      .block 1
0000:7D05 ??      .block 1
0000:7D06 ??      .block 1
0000:7D07 ??      .block 1
0000:7D08 ?? ?? ?? ??+ .block 78h
0000:7D08 ??      dsw_audio_irq: .block 1
0000:7D81 ??      .block 1
0000:7D82 ??      flipscreen: .block 1
0000:7D82          ; DATA XREF: 0000:01E4|w
0000:7D83 ??      .block 1
0000:7D84 ??      spritebank: .block 1
0000:7D85 ??      nmi_mask: .block 1
0000:7D85 ??      p8257_drq: .block 1
0000:7D86 ?? ??      palette_bank: .block 2
0000:7D86          ; DATA XREF: 0000:02A8|w
0000:7D86          ; 0000:0779|o ...
0000:7D88 ?? ?? ?? ??+ .block 78h
0000:7D88 ?? ?? ?? ??+; end of 'IO'
0000:7D88 ?? ?? ?? ??+
0000:7D88 ?? ?? ?? ??+
0000:7D88 ?? ?? ?? ??+
0000:7D88 ?? ?? ?? ??+ .end

```