```
         LD SP,$fffe             ; $0000  Setup Stack

         XOR A                   ; $0003  Zero the memory from $8000-$9FFF (VRAM)
         LD HL,$9fff             ; $0004
 Addr_0007:
         LD (HL-),A              ; $0007
         BIT 7,H         ; $0008
         JR NZ, Addr_0007        ; $000a

         LD HL,$ff26             ; $000c  Setup Audio
         LD C,$11                ; $000f
         LD A,$80                ; $0011
         LD (HL-),A              ; $0013
         LD ($FF00+C),A  ; $0014
         INC C                   ; $0015
         LD A,$f3                ; $0016
         LD ($FF00+C),A  ; $0018
         LD (HL-),A              ; $0019
         LD A,$77                ; $001a
         LD (HL),A               ; $001c

         LD A,$fc                ; $001d  Setup BG palette
         LD ($FF00+$47),A        ; $001f

         LD DE,$0104             ; $0021  Convert and load logo data from cart into Video RAM
         LD HL,$8010             ; $0024
 Addr_0027:
         LD A,(DE)               ; $0027
         CALL $0095              ; $0028
         CALL $0096              ; $002b
         INC DE          ; $002e
         LD A,E          ; $002f
         CP $34          ; $0030
         JR NZ, Addr_0027        ; $0032

         LD DE,$00d8             ; $0034  Load 8 additional bytes into Video RAM
         LD B,$08                ; $0037
 Addr_0039:
         LD A,(DE)               ; $0039
         INC DE          ; $003a
         LD (HL+),A              ; $003b
         INC HL          ; $003c
         DEC B                   ; $003d
         JR NZ, Addr_0039        ; $003e

         LD A,$19                ; $0040  Setup background tilemap
         LD ($9910),A    ; $0042
         LD HL,$992f             ; $0045
```

```
Addr_0048:
        LD C,$0c                ; $0048
Addr_004A:
        DEC A                   ; $004a
        JR Z, Addr_0055 ; $004b
        LD (HL-),A              ; $004d
        DEC C                   ; $004e
        JR NZ, Addr_004A        ; $004f
        LD L,$0f                ; $0051
        JR Addr_0048    ; $0053

        ; === Scroll logo on screen, and play logo sound===

Addr_0055:
        LD H,A          ; $0055  Initialize scroll count, H=0
        LD A,$64                ; $0056
        LD D,A          ; $0058  set loop count, D=$64
        LD ($FF00+$42),A        ; $0059  Set vertical scroll register
        LD A,$91                ; $005b
        LD ($FF00+$40),A        ; $005d  Turn on LCD, showing Background
        INC B                   ; $005f  Set B=1
Addr_0060:
        LD E,$02                ; $0060
Addr_0062:
        LD C,$0c                ; $0062
Addr_0064:
        LD A,($FF00+$44)        ; $0064  wait for screen frame
        CP $90          ; $0066
        JR NZ, Addr_0064        ; $0068
        DEC C                   ; $006a
        JR NZ, Addr_0064        ; $006b
        DEC E                   ; $006d
        JR NZ, Addr_0062        ; $006e

        LD C,$13                ; $0070
        INC H                   ; $0072  increment scroll count
        LD A,H          ; $0073
        LD E,$83                ; $0074
        CP $62          ; $0076  $62 counts in, play sound #1
        JR Z, Addr_0080 ; $0078
        LD E,$c1                ; $007a
        CP $64          ; $007c
        JR NZ, Addr_0086        ; $007e  $64 counts in, play sound #2
Addr_0080:
        LD A,E          ; $0080  play sound
        LD ($FF00+C),A  ; $0081
        INC C                   ; $0082
        LD A,$87                ; $0083
        LD ($FF00+C),A  ; $0085
```

```
 Addr_0086:
         LD A,($FF00+$42)        ; $0086
         SUB B                   ; $0088
         LD ($FF00+$42),A        ; $0089  scroll logo up if B=1
         DEC D                   ; $008b
         JR NZ, Addr_0060        ; $008c

         DEC B                   ; $008e  set B=0 first time
         JR NZ, Addr_00E0        ; $008f    ... next time, cause jump to "Nintendo Logo check"

         LD D,$20                ; $0091  use scrolling loop to pause
         JR Addr_0060    ; $0093

         ; ==== Graphic routine ====

         LD C,A          ; $0095  "Double up" all the bits of the graphics data
         LD B,$04                ; $0096     and store in Video RAM
 Addr_0098:
         PUSH BC         ; $0098
         RL C                    ; $0099
         RLA                     ; $009b
         POP BC          ; $009c
         RL C                    ; $009d
         RLA                     ; $009f
         DEC B                   ; $00a0
         JR NZ, Addr_0098        ; $00a1
         LD (HL+),A              ; $00a3
         INC HL          ; $00a4
         LD (HL+),A              ; $00a5
         INC HL          ; $00a6
         RET                     ; $00a7

 Addr_00A8:
         ;Nintendo Logo
         .DB $CE,$ED,$66,$66,$CC,$0D,$00,$0B,$03,$73,$00,$83,$00,$0C,$00,$0D
         .DB $00,$08,$11,$1F,$88,$89,$00,$0E,$DC,$CC,$6E,$E6,$DD,$DD,$D9,$99
         .DB $BB,$BB,$67,$63,$6E,$0E,$EC,$CC,$DD,$DC,$99,$9F,$BB,$B9,$33,$3E

 Addr_00D8:
         ;More video data
         .DB $3C,$42,$B9,$A5,$B9,$A5,$42,$3C

         ; ===== Nintendo logo comparison routine =====

 Addr_00E0:
         LD HL,$0104             ; $00e0 ; point HL to Nintendo logo in cart
         LD DE,$00a8             ; $00e3 ; point DE to Nintendo logo in DMG rom

 Addr_00E6:
         LD A,(DE)               ; $00e6
         INC DE          ; $00e7
         CP (HL)         ; $00e8 ;compare logo data in cart to DMG rom
         JR NZ,$fe               ; $00e9 ;if not a match, lock up here
         INC HL          ; $00eb
         LD A,L          ; $00ec
         CP $34          ; $00ed ;do this for $30 bytes
         JR NZ, Addr_00E6        ; $00ef

         LD B,$19                ; $00f1
         LD A,B          ; $00f3
 Addr_00F4:
         ADD (HL)                ; $00f4
         INC HL          ; $00f5
         DEC B                   ; $00f6
         JR NZ, Addr_00F4        ; $00f7
         ADD (HL)                ; $00f9
         JR NZ,$fe               ; $00fa ; if $19 + bytes from $0134-$014D  don't add to $00
                                                 ;  ... lock up

         LD A,$01                ; $00fc
         LD ($FF00+$50),A        ; $00fe ;turn off DMG rom
```