```
; Disassembly of the Smart Basic loader

OPENFILE      .equ         0FCC0h
READFILE      .equ         0FCD2h
CLOSEFILE     .equ         0FCC3h
SMARTWRITER   .equ         0FCE7h

DESTADDR      .equ         0d000h
BLOCKSLEFT    .equ         0d002h

ONEKB         .equ         0400h

FILENUMBER    .equ         0d004h

        .org    0c800h

        di                                  ; Disable interrupts
        jr      Continue                    ; Jump over program name
ProgName:
        .db     42h     ;'B'
        .db     41h     ;'A'
        .db     53h     ;'S'
        .db     49h     ;'I'
        .db     43h     ;'C'
        .db     50h     ;'P'
        .db     47h     ;'G'
        .db     4dh     ;'M'
        .db     02h
        .db     03h
ProgStart:
        .db     00h
        .db     01h
ProgSize:
        .db     1ch
        .db     00h
Continue:
        ld      sp,0fe58h                   ; Set the stack pointer
                                            ; Setup to load BasicPgm (Smart
Basic)
        ld      a,b                         ; B = current device number
        ld      b,01h                       ; File mode 1 = read
        ld      hl,ProgName                 ; Name of file to open
        call    OPENFILE                    ; EOS Open FIle
        jr      z,NoOpenError               ; Z flag set = no error
        jp      SMARTWRITER                 ; Exit to Smart Writer
NoOpenError:
        ld      (FILENUMBER),a              ; Save the File Number returned by
OPENFILE
        call    LoadBasic                   ; Load Smart Basic
        cp      0h                          ; Load return 0?
        jr      z,NoLoadError               ; Yes so no error
        jp      SMARTWRITER                 ; Exit to Smart Writer
NoLoadError:
        ld      a,(FILENUMBER)              ; Get the File Number
        call    CLOSEFILE                   ; Close the file
        jr      z,NoCloseError              ; No error closing
```

```
        jp      SMARTWRITER                     ; Exit to Smart Writer
NoCloseError:
        ld      hl,(ProgStart)                  ; Address we loaded Basic at
(0100h)
        jp      (hl)                            ; Jump to it
LoadBasic:
        ld      hl,ProgStart                    ; Address to load Basic to (0100h)
        ld      de,DESTADDR                     ; Copy it and the size to 0d000h
        ld      bc,04h                          ; (4 bytes)
        ldir                                    ;
LoadLoop:
        ld      hl,BLOCKSLEFT                   ; Decrement Number of blocks left
        dec     (hl)
        ld      a,(FILENUMBER)                  ; Load the FILENUMBER
        ld      hl,(DESTADDR)                   ; Where to store it
        ld      bc,ONEKB                        ; Load 1k of data
        call    READFILE
        jr      nz,LoadBasicError               ; Error loading - return
        ld      bc,ONEKB
        add     hl,bc                           ; Increment DESTADDR by 1kb
        ld      ix,0d000h                       ; Save HL to DESTADDR using IX
        ld      (ix+0),l
        ld      (ix+1),h
        ld      a,(0d002h)                      ; BLOCKSLEFT == 0?
        cp      0h                              ; Compare to A
        jr      nz,LoadLoop                     ; A <> 0 jump to LoadLoop
        ret                                     ; Return no error
LoadBasicError:
        ld      a,1h                            ; Return with an error
        ret

        .end
```