(1) Receive buffer get point
The address of the next data to be gotten from the receive buffer
(3) Receive buffer put point
The receive buffer address into which the next data is to be put
(4) Receive buffer address
The receive buffer starting address
(5) Receive buffer size
The receive buffer size

Notes:
XON/XOFF control specification is invalid when the receive buffer size is smaller than 16 bytes.

SI/SO control must be specified when sending 8-bit data in the 7-bit code mode. Codes ØEH and ØFH cannot be transferred when SI/SO control is specified. This means that binary data cannot be sent nor received. See Section 5.2, "Serial Interfaces" for details.

XON/XOFF control must be specified when the transmission speed of the sender is greater than the processing speed of the receiver. This synchronizes the operations of the sender and receiver. Codes 11H and 13H cannot be transferred when XON/XOFF control is specified. Therefore, no binary data can be sent or received. See Section 5.2, "Serial Interfaces" for details.

(23-2) RSIOX CLOSE

Function: Closes the currently open interface.

Entry parameter: B = 2ØH

Return parameter: None.

Explanation:
RSIOX CLOSE disables serial interface receive interrupts and performs the close processing.

Note:
When a WBOOT is executed, the system automatically closes the serial interface.

(23-3) RSIOX INSTS

Function: Indicates whether received data is present in the receive buffer.

Entry parameter: B = 3ØH
HL = Starting address of the field for storing 9-byte return information

Return parameter: Z flag = 1: Normal termination
A = ØFFH: Received data present
= ØØH: No received data present
BC = Number of received data bytes in the buffer
HL = The address specified on entry The nine bytes starting at this address contain the return information described earlier (see RSIOX OPEN).

Z flag = Ø: Abnormal termination
A = Ø3H: The interface is not open.

Explanation:
RSIOX INSTS checks whether or not received data is present in the receive buffer and places the result in the return information block.

XON or XOFF codes in the XON/XOFF mode and SI and SO codes in the SI/SO mode are excluded from the received byte count.

(23-4) RSIOX OUTST

Function: Checks whether the interface is enabled for transmission.

Entry parameter: B = 4ØH
HL = Starting address of the field for storing 9-byte return information

Return parameter: Z flag = 1: Normal termination
A = ØØH: Transmission disabled
= ØFFH: Transmission enabled
HL = The address specified on entry The nine bytes starting at this address contain the return information described earlier (see RSIOX OPEN).

Z flag = Ø: Abnormal termination
A = Ø3H: The interface is not open.

Explanation:
RSIOX OUTST determines whether the interface is enabled or disabled for transmission by checking TxRDY. The interface is disabled for transmission if an XON is received when XON/XOFF control is specified.

RSIOX OUTST places the current transmission status to the return information block.

(23-5) RSIOX GET

Function: Gets one data byte from the receive buffer.

Entry parameter: B = 5ØH
HL = Starting address of the field for storing 9-byte return information

Return parameter: Z flag = 1: Normal termination
A = Received data
HL = The address specified on entry The nine bytes starting at this address contain the return information described earlier (see RSIOX OPEN).

Z flag = Ø: Abnormal termination
A = Ø3H: The interface is not open.
= Ø4H: CTRL/STOP is pressed.
= Ø5H: A receive buffer overflow occurred.

Explanation:

RSIOX GET gets one byte of received data from the receive buffer and loads it into the A register. If no data is present, RSIOS GET waits until a byte is received.

Any power-off or alarm/wake interrupts occurring during execution of RSIOX GET are processed. After processing an interrupt, RSIOX GET resumes processing at the point where the interrupt occurred.

(23-6) RSIOX PUT

Function: Transfers one data byte to the interface.

Entry parameter: B = 60H
C = Send data
HL = Starting address of the field for storing 9-byte return information

Return parameter: Z flag = 1: Normal termination
HL = The address specified on entry The nine bytes starting at this address contain the return information described earlier (see RSIOX OPEN).

Z flag = 0: Abnormal termination
A = 03H: The interface is not open.
= 04H: CTRL/STOP is pressed.

Explanation:

RSIOX PUT checks whether the interface is ready for transmission and, if it is, sends the given data to the interface.

RSIOX PUT determines whether the interface is ready in the same way as RSIOX OUTST. If the interface is not ready, RSIOX PUT waits until it is ready.

If the CTRL/STOP key is pressed while RSIOX PUT is waiting for the interface to get ready, RSIOX PUT returns control to the calling program with 04H in the A register.

Any power-off or alarm/wake interrupts occurring during execution of the RSIOX PUT routine are serviced. After the completion of the interrupt processing, the RSIOX PUT processing resumes at the point where the interrupt occurred.

(23-7) RSIOX CTLIN

Function: Reads the state of a given control line.

Entry parameter: B = 70H

Return parameter: Z flag = 1: Normal termination
A = Control line state

Z flag = 0: Abnormal termination
A = 03H: The interface is not open.

Explanation:
The format of the control line status parameter is shown below:

| Bit | Description |
|---|---|
| 7 | DSR (Data Set Ready) state<br>= Ø: Active<br>= 1: Not active |
| 6 | Not used |
| 5 | CTS (Clear To Send) state<br>= Ø: Not active<br>= 1: Active |
| 4 | Not used. |
| 3 | CD (Carrier Detect) state<br>= Ø: Not active<br>= 1: Active |
| 2 - Ø | Not used |

Both CTS and CD are valid only in the RS-232C mode. DSR is valid in RS-232C or SIO mode. In SIO mode, however, the SIN line is used instead of the DSR line.

(23-8) RSIOX SETCTL

Function: Sets the given control line to the specified state.

Entry parameter: B = 8ØH
A = Control line state (described below)

Return parameter: Z flag = 1: Normal termination
Z flag = Ø: Abnormal termination
A = Ø3H: The interface is not open.

Explanation:
The format of the control line status parameter is shown below:

| Bit | Description |
|---|---|
| 7 - 2 | Not used (must be Ø). |
| 1 | RTS (Request To Send) state<br>= Ø: Not active<br>= 1: Active |
| Ø | DTR (Data Transmit Ready) state<br>= Ø: Not active<br>= 1: Active |

Both RTS are valid only in the RS-232C mode. DTR is valid in RS-232C or SIO mode. In SIO mode, however, the SIN line is used instead of the DTR line.

RSIOX SETCTL is used to reset the control line state that was specified when the interface was opened.

(23-9) RSIOX ERSTS

Function: Reads the error status of the interface and clears the error flags.

Entry parameter: B = 9ØH

Return parameter: Z flag = 1: Normal termination
A = Error status (described below)
Z flag = Ø: Abnormal termination
A = Ø3H: The interface is not open.

Explanation:

The format of the control line status parameter is shown below:

| Bit | Description |
|---|---|
| 7 | DSR (Data Set Ready) state<br>= Ø: Active<br>= 1: Not active |
| 6 | Framing error state<br>= Ø: No framing error occurred.<br>= 1: Framing error occurred. |
| 5 | Receive overrun error state<br>= Ø: No receive overrun error occurred.<br>= 1: Receive overrun error occurred. |
| 4 | Parity error state<br>= Ø: No parity error occurred.<br>= 1: Parity error occurred. |
| 3 | CD (Carrier Detect ) state<br>= Ø: Not active<br>= 1: Active |
| 2 | Receive buffer overflow error state<br>= Ø: No receive buffer overflow error occurred.<br>= 1: Receive buffer overflow error occurred. |
| 1, Ø | Not used |

RSIOX ERSTS returns the same status information for the CD and DSR lines regardless of the type of the device.

RSIOX ERSTS clears all error states after reading the current error states.

Note:

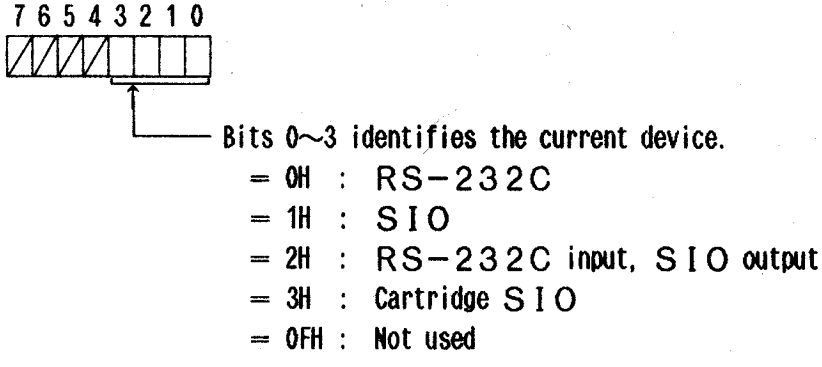
Once a receive buffer overflow error occurs, any subsequent bytes received until a byte is read from the receive buffer by the RSIOX GET routine are discarded.

(23-1Ø) RSIOX SENS

Function: Returns the serial interface in use.

Entry parameter: B = ØFØH

Return parameter: A = Current serial interface



Explanation:

RSIOX SENS checks and indicate in the A register the serial interface that is currently being used.

```
                        ;***********************************************
                        ;       BIOS CALL SAMPLE PROGRAM
                        ;***********************************************
                        ;
                        ;       NOTE :
                        ;
                        ;       <> assemble condition <>
                        ;
                        ;       .Z80
                        ;
                        ;       <> loading address <>
                        ;
                        ;       .PHASE  100H
                        ;
                        ;       <> constant values <>
                        ;
EB03                    WBOOT           EQU     0EB03H          ; WBOOT BIOS entry address
EB06                    CONST           EQU     WBOOT   +03H    ; CONST BIOS entry address
EB09                    CONIN           EQU     WBOOT   +06H    ; CONIN BIOS entry address
EB0C                    CONOUT          EQU     WBOOT   +09H    ; CONOUT BIOS entry address
EB54                    RSIOX           EQU     WBOOT   +51H    ; RSIOX BIOS entry address
EB69                    CALLX           EQU     WBOOT   +66H    ; CALLX BIOS entry address
                        ;
0010                    RSOPN           EQU     10H             ; RS232C OPEN function
0020                    RSCLS           EQU     20H             ; CLOSE function
0030                    RSIST           EQU     30H             ; INPUT STATUS function
0040                    RSOST           EQU     40H             ; OUTPUT STATUS function
0050                    RSGET           EQU     50H             ; GET function
0060                    RSPUT           EQU     60H             ; PUT function
0090                    RSERR           EQU     90H             ; ERROR STATUS function
                        ;
EF31                    SRSADR          EQU     0EF31H          ; System serial parameter.
F52E                    DISBNK          EQU     0F52EH          ; Destination bank area
                        ;
0000                    HELP            EQU     00H             ; HELP code
000D                    CR              EQU     0DH             ; Carriage return code
000A                    LF              EQU     0AH             ; Line feed code
001B                    ESC             EQU     1BH             ; Escape code
0009                    TAB             EQU     09H             ; Tab code
                        ;
003C                    XUSRSCRN        EQU     003CH           ; Change to system screen.
003F                    XSYSSCRN        EQU     003FH           ; Change to user screen.
                        ;
                        ;***********************************************
                        ;       MAIN PROGRAM
                        ;***********************************************
                        ;
0100                    START:
0100    31 1000                 LD      SP,1000H        ; Set stack pointer.
                        ;
0103    21 EF31                 LD      HL,SRSADR       ; Copy open parameter from system area.
0106    11 02DB                 LD      DE,OPNPRM       ; Application parameter area.
0109    01 0009                 LD      BC,9            ; Parameter number.
010C    ED B0                   LDIR                    ; Copy.
                        ;
010E    21 02DB                 LD      HL,OPNPRM       ; Open parameter.
0111    06 10                   LD      B,RSOPN         ; RS232C open function.
0113    CD EB54                 CALL    RSIOX           ; OPEN.
0116    B7                      OR      A               ; Error return?
0117    C2 EB03                 JP      NZ,WBOOT        ; Yes, then WBOOT.
                        ;
011A                    KEYCHK:
011A    CD EB06                 CALL    CONST           ; Get key inputed status.
011D    3C                      INC     A               ; Input any key?
011E    CC 0137                 CALL    Z,PUT           ; Yes, then put the data.
                        ;
0121    21 02DB                 LD      HL,OPNPRM       ; Get input status.
0124    06 30                   LD      B,RSIST         ;  Input status function.
0126    CD EB54                 CALL    RSIOX           ;  Get input status.
0129    3C                      INC     A               ; If there is receiving data,
012A    CC 0189                 CALL    Z,GET           ;  then get the data
012D    18 EB                   JR      KEYCHK          ; Loop.
                        ;
012F                    PEND:
012F    06 20                   LD      B,RSCLS         ; Close RSIOX
0131    CD EB54                 CALL    RSIOX           ;
0134    C3 EB03                 JP      WBOOT           ; Program end.
                        ;
                        ;***********************************************
                        ;       PUT INPUTED DATA TO RS232C
                        ;***********************************************
                        ;
                        ;       NOTE :
                        ;
                        ;       <> entry parameter <>
                        ;               NON
                        ;       <> return parameter <>
                        ;               NON
                        ;       <> preserved registers <>
                        ;               NON
                        ;
                        ;       CAUTION :
                        ;               If inputed data is BREAK key, this program
                        ;                ends.
                        ;               If inputed data is HELP key, put from '0'
                        ;                to '9' to RS232C.
                        ;
0137                    PUT:
0137    CD EB09                 CALL    CONIN           ; Get inputed data.
013A    4F                      LD      C,A             ;
013B    FE 03                   CP      03H             ; If inputed key is BREAK,
013D    CA EB03                 JP      Z,WBOOT         ;  then end of program.
                        ;
0140    FE 00                   CP      HELP            ; If inputed key is HELP,
0142    CA 0169                 JP      Z,SEND          ;  then send '0' to '9'.
0145    C5                      PUSH    BC              ; Save input key code.
                        ;
```

```
0146    C5                      PUSH    BC              ; Save input key code.
0147    FE 0D                   CP      CR              ; If inputed key is RETURN,
0149    0E 0A                   LD      C,LF            ;  then LF console out.
014B    CC EB0C                 CALL    Z,CONOUT        ;
014E    C1                      POP     BC              ; Restore input key code.
014F    CD EB0C                 CALL    CONOUT          ; Console out inputing data.
                                ;
0152    06 90                   LD      B,RSERR         ; Get error status.
0154    CD EB54                 CALL    RSIOX           ;
0157    E6 74                   AND     01110100B       ; If error is happened.
0159    C4 01CF                 CALL    NZ,RGSDSP       ;  then display the error status.
                                ;
015C    C1                      POP     BC              ; Restore the input key code.
015D    21 02DB                 LD      HL,OPNPRM       ; Put inputing data to RS232C.
0160    06 60                   LD      B,RSPUT         ;  Put function code.
0162    CD EB54                 CALL    RSIOX           ;  Put data.
0165    C4 01CF                 CALL    NZ,RGSDSP       ; If error return, then display error.
0168    C9                      RET                     ;

                        ;       ************************************************
                        ;               SEND '0' TO '9'
                        ;       ************************************************
                        ;
                        ;       NOTE :
                        ;
                        ;       <> entry parameter <>
                        ;               NON
                        ;       <> return parameter <>
                        ;               NON
                        ;       <> preserved registers <>
                        ;               NON
                        ;
                        ;       CAUTION :
                        ;               If HELP key is pressed, then return.
0169                    ;
0169    0E 30           SEND:   LD      C,'0'           ; Start character code.
016B                    SEND10:
016B    C5                      PUSH    BC              ; Save send data.
016C    21 02DB                 LD      HL,OPNPRM       ; Put data to RS232C.
016F    06 60                   LD      B,RSPUT         ;  Put function.
0171    CD EB54                 CALL    RSIOX           ;  Put.
0174    C1                      POP     BC              ; Restore send data.
0175    0C                      INC     C               ; Send data update.
0176    3E 3A                   LD      A,'9'+1         ; Send '0' to '9'?
0178    B9                      CP      C               ;
0179    20 F0                   JR      NZ,SEND10       ; No.
                                ;
017B    CD EB06                 CALL    CONST           ; Check input status.
017E    B7                      OR      A               ; No key is pressed?
017F    28 E6                   JR      Z,SEND          ; Yes.
0181    CD EB09                 CALL    CONIN           ; Get pressed key code.
0184    FE 00                   CP      HELP            ; HELP key is pressed?
0186    20 E1                   JR      NZ,SEND         ; No.
0188    C9                      RET                     ;

                        ;       ************************************************
                        ;               GET RECIVED DATA
                        ;       ************************************************
                        ;
                        ;       NOTE :
                        ;
                        ;       <> entry parameter <>
                        ;               NON
                        ;       <> return parameter <>
                        ;               NON
                        ;       <> preserved registers <>
                        ;               NON
                        ;
                        ;       CAUTION :
0189                    ;
0189    06 90           GET:    LD      B,RSERR         ; Check error status.
018B    CD EB54                 CALL    RSIOX           ; Get error status.
018E    E6 74                   AND     01110100B       ; Error is happened?
0190    C4 01CF                 CALL    NZ,RGSDSP       ; Yes, then display error.
                                ;
0193    21 02DB                 LD      HL,OPNPRM       ; Get received data.
0196    06 50                   LD      B,RSGET         ;  Get function code.
0198    CD EB54                 CALL    RSIOX           ;  Get.
019B    C4 01CF                 CALL    NZ,RGSDSP       ; If error, then display error.
                                ;
019E    4F                      LD      C,A             ; Console out received data.
019F    CD 01A9                 CALL    RVSON           ; Reverse on.
01A2    CD EB0C                 CALL    CONOUT          ; Display received data.
01A5    CD 01B6                 CALL    RVSOFF          ; Reverse off.
01A8    C9                      RET                     ;

                        ;       ************************************************
                        ;               REVERSE MODE ON
                        ;       ************************************************
                        ;
                        ;          E :
                        ;
                        ;       <> entry parameter <>
                        ;               NON
                        ;       <> return parameter <>
                        ;               NON
                        ;       <> preserved registers <>
                        ;               BC
                        ;
                        ;       CAUTION :
01A9                    ;
01A9    C5              RVSON:  PUSH    BC              ; Save BC register.
01AA    0E 1B                   LD      C,ESC           ; Reverse on command.
01AC    CD EB0C                 CALL    CONOUT          ;  ESC + '0'
```

```
01AF    0E 30                   LD      C,'0'           ;
01B1    CD EB0C                 CALL    CONOUT          ;
01B4    C1                      POP     BC              ; Restore BC register.
01B5    C9                      RET                     ;

                        ;***********************************************
                        ;       REVERSE MODE OFF
                        ;***********************************************
                        ;
                        ;       NOTE :
                        ;
                        ;       <> entry parameter <>
                        ;               NON
                        ;       <> return parameter <>
                        ;               NON
                        ;       <> preserved registers <>
                        ;               BC
                        ;
                        ;       CAUTION :
                        ;
01B6                    RVSOFF:
01B6    C5                      PUSH    BC              ; Save BC register
01B7    0E 1B                   LD      C,ESC           ; Reverse off command
01B9    CD EB0C                 CALL    CONOUT          ;  ESC + '1'
01BC    0E 31                   LD      C,'1'           ;
01BE    CD EB0C                 CALL    CONOUT          ;
01C1    C1                      POP     BC              ; Restore BC register
01C2    C9                      RET                     ;

                        ;***********************************************
                        ;       DISPLAY MESSAGE UNTIL FIND 0
                        ;***********************************************
                        ;
                        ;       NOTE :
                        ;
                        ;       <> entry parameter <>
                        ;               HL : Message data top address.
                        ;       <> return parameter <>
                        ;               NON
                        ;       <> preserved registers <>
                        ;               NON
                        ;
                        ;       CAUTION :
                        ;
01C3                    MSGDSP:
01C3    7E                      LD      A,(HL)          ; Get display data.
01C4    B7                      OR      A               ; Data is end code?
01C5    C8                      RET     Z               ; Yes.
                        ;
01C6    4F                      LD      C,A             ;
01C7    E5                      PUSH    HL              ; Save data address.
01C8    CD EB0C                 CALL    CONOUT          ; Console out the data.
01CB    E1                      POP     HL              ; Restore data address.
01CC    23                      INC     HL              ; Data adddress update.
01CD    18 F4                   JR      MSGDSP          ; Loop.

                        ;***********************************************
                        ;       DISPLAY REGISTERS
                        ;***********************************************
                        ;
                        ;       NOTE :
                        ;
                        ;       <> entry parameter <>
                        ;               NON
                        ;       <> return parameter <>
                        ;               NON
                        ;       <> preserved registers <>
                        ;               All registers
                        ;
                        ;       CAUTION :
                        ;
01CF                    RGSDSP:
01CF    F5                      PUSH    AF              ; Save all registers.
01D0    C5                      PUSH    BC              ;
01D1    D5                      PUSH    DE              ;
01D2    E5                      PUSH    HL              ;
                        ;
01D3    F5                      PUSH    AF              ; Save display resisters
01D4    C5                      PUSH    BC              ;
01D5    D5                      PUSH    DE              ;
01D6    E5                      PUSH    HL              ;
                        ;
01D7    DD 21 003F              LD      IX,XSYSSCRN     ; Change to system screen.
01DB    3E FF                   LD      A,0FFH          ;  Set system bank.
01DD    32 F52E                 LD      (DISBNK),A      ;
01E0    CD EB69                 CALL    CALLX           ;  Call OS jump table.
                        ;
01E3    0E 0C                   LD      C,0CH           ; Clear screen & home.
01E5    CD EB0C                 CALL    CONOUT          ;
                        ;
01E8    21 0286                 LD      HL,HLDSP        ; HL register display.
01EB    C1                      POP     BC              ;  HL register data.
01EC    CD 024C                 CALL    BINASC          ;  Convert binary to ASCII.
01EF    21 0291                 LD      HL,DEDSP        ; DE register display.
01F2    C1                      POP     BC              ;  DE register data.
01F3    CD 024C                 CALL    BINASC          ;  Convert binary to ASCII.
01F6    21 029D                 LD      HL,BCDSP        ; BC register display.
01F9    C1                      POP     BC              ;  BC register data.
01FA    CD 024C                 CALL    BINASC          ;  Convert binary to ASCII.
01FD    21 02A8                 LD      HL,AFDSP        ; AF register display.
0200    C1                      POP     BC              ;  AF register data.
0201    CD 024C                 CALL    BINASC          ;  Convert binary to ASCII.
```

```
0204  21 02AE                   LD      HL,DTDSP        ; Return information area display.
0207  ED 5B 02DB                LD      DE,(OPNPRM)     ;
020B  CD 024A                   CALL    BINASC0         ;
020E  ED 5B 02DD                LD      DE,(OPNPRM+2)   ;
0212  CD 024A                   CALL    BINASC0         ;
0215  ED 5B 02DF                LD      DE,(OPNPRM+4)   ;
0219  CD 024A                   CALL    BINASC0         ;
021C  ED 5B 02E1                LD      DE,(OPNPRM+6)   ;
0220  CD 024A                   CALL    BINASC0         ;
0223  ED 5B 02E3                LD      DE,(OPNPRM+8)   ;
0227  16 00                     LD      D,0             ;
0229  CD 024A                   CALL    BINASC0         ;
                        ;
022C  21 026B                   LD      HL,RGSMSG       ; Message display.
022F  CD 01C3                   CALL    MSGDSP          ;
0232                    RETRY:
0232  CD EB09                   CALL    CONIN           ; Key input.
0235  FE 00                     CP      HELP            ; HELP key?
0237  20 F9                     JR      NZ,RETRY        ; No.
                        ;
0239                    DSPEND:
0239  DD 21 003C                LD      IX,XUSRSCRN     ; Change to user screen.
023D  3E FF                     LD      A,0FFH          ;  OS bank
023F  32 F52E                   LD      (DISBNK),A      ;
0242  CD EB69                   CALL    CALLX           ;
                        ;
0245  E1                        POP     HL              ; Restore registers.
0246  D1                        POP     DE              ;
0247  C1                        POP     BC              ;
0248  F1                        POP     AF              ;
0249  C9                        RET                     ;
                        ;
                        ;       ************************************************
                        ;               CHANGE BINARY TO ASCII
                        ;       ************************************************
                        ;
                        ;       NOTE :
                        ;
                        ;       <> entry parameter <>
                        ;               BC : Binary data.
                        ;               HL : ASCII data setting address.
                        ;       <> return parameter <>
                        ;               HL : HL + 2
                        ;          preserved registers <>
                        ;               NON
                        ;
                        ;       CAUTION :
                        ;
024A                    BINASC0:
024A  43                        LD      B,E             ; E --> B
024B  4A                        LD      C,D             ; D --> C
                        ;
024C                    BINASC:
024C  78                        LD      A,B             ; Change B register.
024D  CD 0251                   CALL    BIN10           ; Convert.
0250  79                        LD      A,C             ; Change C register.
                        ;
0251                    BIN10:
0251  F5                        PUSH    AF              ; Save binary data.
0252  1F                        RRA                     ; Shift 4 bit.
0253  1F                        RRA                     ;
0254  1F                        RRA                     ;
0255  1F                        RRA                     ;
0256  CD 025A                   CALL    CONV00          ; Binary --> ASCII
0259  F1                        POP     AF              ; Restore binary data.
                        ;
025A                    CONV00:
025A  E6 0F                     AND     0FH             ; LSB 4 bit.
025C  FE 0A                     CP      10              ; 0 -- 9 ?
025E  38 06                     JR      C,CONV20        ; Yes.
0260  D6 09                     SUB     9               ; Change 'A' to 'F'.
0262  F6 40                     OR      01000000B       ;
0264  18 02                     JR      CONV25          ;
0266                    CONV20:
0266  F6 30                     OR      00110000B       ; Change '0' to '9'.
0268                    CONV25:
0268  77                        LD      (HL),A          ; Set converted data to (hl)
0269  23                        INC     HL              ; Pointer update.
026A  C9                        RET                     ;
                        ;
                        ;       ************************************************
                        ;               MESSAGE DATA
                        ;       ************************************************
                        ;
026B                    RGSMSG:
026B  0D 0A                     DB      CR,LF
026D  50 61 72 61               DB      'Parameter display',CR,LF
0271  6D 65 74 65
0275  72 20 64 69
0279  73 70 6C 61
027D  79 0D 0A
0280  48 4C 20 2D               DB      'HL -- '
0284  2D 20
0286                    HLDSP:
0286                            DS      4
028A  09                        DB      TAB
028B  44 45 20 2D               DB      'DE -- '
028F  2D 20
0291                    DEDSP:
0291                            DS      4
0295  0D 0A                     DB      CR,LF
0297  42 43 20 2D               DB      'BC -- '
029B  2D 20
029D                    BCDSP:
029D                            DS      4
02A1  09                        DB      TAB
02A2  41 46 20 2D               DB      'AF -- '
```

```
02A6    2D 20
02A8                            AFDSP:
02A8                                    DS      4
02AC    0D 0A                           DB      CR,LF
02AE                            DTDSP:
02AE                                    DS      20
02C2    0D 0A                           DB      CR,LF
02C4    50 72 65 73                     DB      'Press HELP to continue'
02C8    73 20 48 45
02CC    4C 50 20 74
02D0    6F 20 63 6F
02D4    6E 74 69 6E
02D8    75 65
02DA    00                              DB      0
                                ;
                                ;       ************************************************
                                ;               WORK AREA
                                ;       ************************************************

02DB                            OPNPRM:
02DB                                    DS      9               ; RSIOX open parameter area
                                ;
                                        END
```

(24) MASKI

Function: Sets an interrupt mask or checks the current mask status.

Entry address: WBOOT + 57H or ØEB5AH

Entry parameter: B = Interrupt mask data
C = 75Ø8 interrupt mask data (described later)

Return parameter: B = Old interrupt mask status
C = Old 75Ø8 interrupt mask status

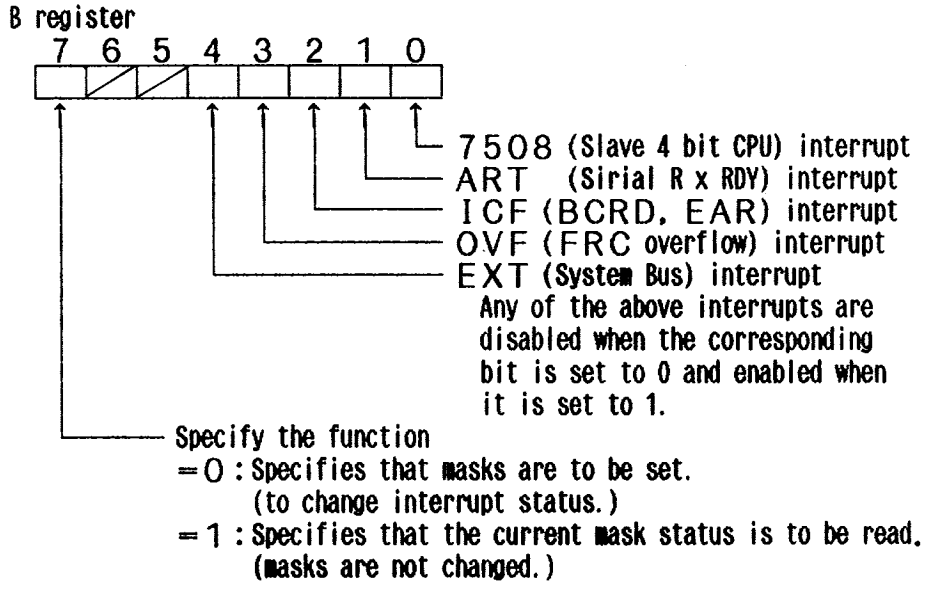
Explanation:
MASKI enables or disables the five types of interrupts supported by the PINE and three types of interrupts that are associated with the 75Ø8. See the next page for entry and return parameters.

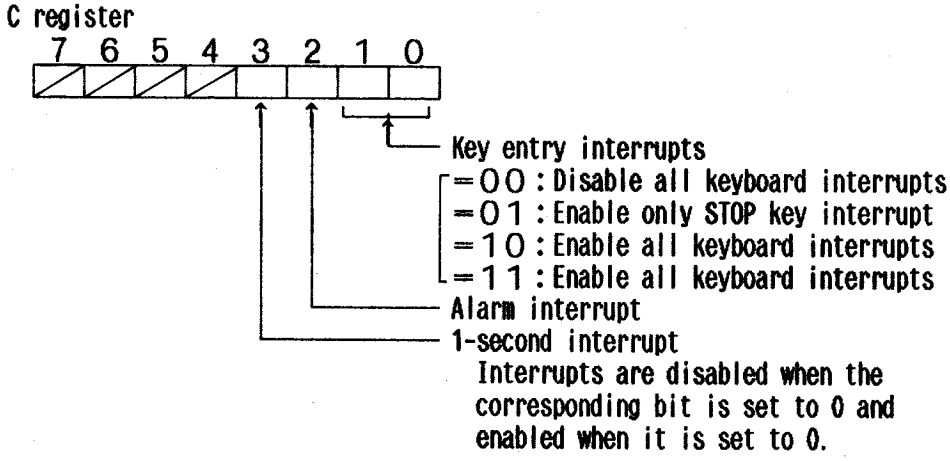
Notes:
When changing the interrupt status, read the current interrupt status with MASKI and set or reset the necessary bit.

It is desirable to restore the original interrupt status after user processing is completed.

See also: Section 4.7, "Interrupts"

Entry parameters






Return parameters

MASKI returns the interrupt mask status at the time when the routine is called. The correspondence between the bits of the BC register and the interrupt types or status is the same as that shown above, except B register bit 7.

Reference:

The current interrupt status is loaded in the following system areas:

ISTS7508 (ØEF93H) 1 byte
- Current 7508 interrupt mask status.
Bits 7 - 4: Don't care.
Bit 3: 1-second interrupt
= Ø: Disabled
= 1: Enabled
Bit 2: Alarm interrupt
= Ø: Disabled
= 1: Enabled
Bits 1, Ø: Keyboard interrupt
= ØØ: All keyboard interrupts disabled.
= Ø1: Only STOP key interrupt enabled.
= 1Ø: All keyboard interrupts enabled.
= 11: All keyboard interrupts enabled.
The default setting is ØBH.

RZIER (ØF53EH) 1 byte
- Current interrupt mask status.
Bits 7 - 5: Don't care.
Bit 4: EXT interrupt
= Ø: Disabled
= 1: Enabled
Bit 3: OVF interrupt
= Ø: Disabled
= 1: Enabled
Bit 2: ICF interrupt
= Ø: Disabled
= 1: Enabled
Bit 1: ART interrupt
= Ø: Disabled
= 1: Enabled
Bit Ø: 7508 interrupt
= Ø: Disabled
= 1: Enabled
The default setting is Ø9H.

Enabling only STOP key interrupts has no effect when the item keyboard is installed. Attempting to do so will disable all keyboard interrupt.

```
                        ;***************************************************
                        ;       BIOS MASKI SAMPLE PROGRAM
                        ;***************************************************
                        ;
                        ;NOTE :
                        ;       This sample program is that all interrupt
                        ;        makes disable except for STOP key inputing.
                        ;
                        ;<> assemble condition <>
                        ;
                        .Z80
                        ;
                        ;<> loading address <>
                        ;
                        .PHASE  100H
                        ;
                        ;<> constant values <>
                        ;
EB5A                    MASKI           EQU     0EB5AH  ; MASKI entry address.
EB03                    WBOOT           EQU     0EB03H  ; WBOOT entry address.
EB09                    CONIN           EQU     0EB09H  ; CONIN entry address.
                        ;
1000                    MAINSP          EQU     1000H   ; Stack pointer.
                        ;
                        ;***************************************************
                        ;       MAIN PROGRAM
                        ;***************************************************
                        ;
                        ;NOTE :
                        ;
0100                    START:
0100    31 1000                 LD      SP,MAINSP       ; Set stack pointer.
                        ;
0103    06 80                   LD      B,80H           ; Get current interrupt status.
0105    CD EB5A                 CALL    MASKI           ;
0108    C5                      PUSH    BC              ; Save current interrupt status.
                        ;
0109    79                      LD      A,C             ; Disable all key interrupt except
010A    E6 FC                   AND     11111100B       ;  STOP key.
010C    F6 01                   OR      00000001B       ;
010E    4F                      LD      C,A             ;
010F    CD EB5A                 CALL    MASKI           ; Set new interrupt status.
                        ;
                        ;Application inserts the process in this part
                        ; which needs to disable interrupt.
                        ;In case of this sample program, STOP key onle can input.
                        ;
0112    CD EB09                 CALL    CONIN           ; Key in. (Only STOP key)
                        ;
0115    C1                      POP     BC              ; Restore interrupt status.
0116    CD EB5A                 CALL    MASKI           ; Restore old interrupt.
                        ;
0119    C3 EB03                 JP      WBOOT           ; Jump WBOOT.
                        ;
                                END
```

(25) LOADX

Function: Reads one byte of data from the specified address on the specified bank.

Entry address: WBOOT + 5AH or ØEB5DH

Entry parameter: C = Bank from which data is to be read
= ØFFH: System bank
= ØØH: Bank Ø
= Ø1H: Bank 1
= Ø2H: Bank 2
HL = Address of the data to be read

Return parameter: A = Contents of the read data

Explanation:
All registers other register than A hold the previous values.

Since LOADX makes no parameter check, normal operation is not guaranteed if a value other than -1 to 2 is specified in C.

See also: Section 4.4, "Bank Switching"

```
                        ;       ************************************************
                        ;               LOADX SAMPLE PROGRAM
                        ;       ************************************************
                        ;
                        ;       NOTE :
                        ;               This sample program is reading from
                        ;               data in target bank.
                        ;
                        ;       <> assemble condition <>
                        ;
                        ;       .Z80
                        ;
                        ;       <> loading address <>
                        ;
                        ;       .PHASE  100H
                        ;
                        ;       <> constant values <>
                        ;
                        ;       BIOS entry
EB03                    WBOOT   EQU     0EB03H          ; Warm Boot entry
EB06                    CONST   EQU     WBOOT   +03H
EB09                    CONIN   EQU     WBOOT   +06H
EB0C                    CONOUT  EQU     WBOOT   +09H    ; Console out entry
EB5D                    LOADX   EQU     WBOOT   +5AH
                        ;
                        ;       Bank value
                        ;
00FF                    SYSBANK EQU     0FFH            ; System bank
0000                    BANK0   EQU     000H            ; Bank 0 (RAM)
0001                    BANK1   EQU     001H            ; Bank 1 (ROM capsel 1)
0002                    BANK2   EQU     002H            ; Bank 2 (ROM capsel 2)
                        ;
                        ;
001B                    ESC     EQU     1BH
0005                    EOL     EQU     05H
0008                    BS      EQU     08H
000D                    CR      EQU     0DH
000A                    LF      EQU     0AH
0003                    STOP    EQU     03H
                        ;       ************************************************
                        ;               MAIN PROGRAM
                        ;       ************************************************
                        ;
                        ;       NOTE :
                        ;               This program is dumping memory.
                        ;
0100                    MAIN:
0100    31 1000                 LD      SP,1000H        ; Set stack pointer.
                        ;
0103    21 02B9                 LD      HL,MSG01        ; Display opening message.
0106    CD 028E                 CALL    DSPMSG          ;
                        ;
0109                    MAIN10:
0109    CD 01CD                 CALL    GETADDR         ; Get address.
010C    DA EB03                 JP      C,WBOOT         ; End if STOP pressed.
010F    CD 0232                 CALL    GETBNK          ; Select bank.
0112    DA EB03                 JP      C,WBOOT         ; End if STOP is pressed.
                        ;
0115    CD 011A                 CALL    DUMPF           ; Dump memory.
0118    18 EF                   JR      MAIN10          ; Loop.
                        ;       ************************************************
                        ;               DUMP FUNCTION
                        ;       ************************************************
                        ;
                        ;       NOTE :
                        ;               Dump memory function
                        ;
                        ;       <> entry parameter <>
                        ;               NON
                        ;       <> return parameter <>
                        ;               NON
                        ;       <> preserved registers <>
                        ;               NON
                        ;
                        ;       CAUTION :
                        ;               If Space bar is pressed, then stop display.
                        ;               If ESC key is pressed, then exit this routine.
                        ;
011A                    DUMPF:
011A    21 02CE                 LD      HL,MSG02        ; Display dump guide line.
011D    CD 028E                 CALL    DSPMSG          ;
                        ;
0120                    DUMP01:
0120    11 03B6                 LD      DE,ASCDATA      ; Getting memory save area.
0123    2A 03BE                 LD      HL,(ADDR)       ; Dump start address.
0126    CD 018D                 CALL    ASCDSP4         ; Display by ASCII.
0129    06 08                   LD      B,08H           ; Loop counter.
012B    3A 03C0                 LD      A,(BANK)        ; Dump target bank.
012E    4F                      LD      C,A             ;
                        ;
012F                    DUMP10:
012F    CD 02A7                 CALL    SPACE           ; Display space.
0132    CD EB5D                 CALL    LOADX           ; Get memory.
0135    12                      LD      (DE),A          ; Set getting data.
0136    CD 0183                 CALL    ASCDSP2         ; Display by ASCII.
0139    23                      INC     HL              ; Pointer update.
013A    13                      INC     DE              ;
013B    10 F2                   DJNZ    DUMP10          ; Loop 8 times.
                        ;
013D    22 03BE                 LD      (ADDR),HL       ; New address.
0140    CD 02A7                 CALL    SPACE           ; Display 2 spaces.
0143    CD 02A7                 CALL    SPACE           ;
0146    CD 015E                 CALL    ASCDSPX         ; Display memory by ASCII.
                        ;
```

```
0149  CD EB06          CALL    CONST           ; Input any key?
014C  3C               INC     A               ;
014D  20 D1            JR      NZ,DUMP01       ; No.
                ;
014F  CD EB09          CALL    CONIN           ; Get inputed key.
0152  FE 1B            CP      ESC             ; ESC?
0154  C8               RET     Z               ; Yes.
0155  FE 20            CP      20H             ; Space?
0157  20 C7            JR      NZ,DUMP01       ; No.
                ;
0159  CD EB09          CALL    CONIN           ; Stop dump until any key inputed.
015C  18 C2            JR      DUMP01          ;
                ;
                ;
                ;
015E                ASCDSPX:
015E  F5               PUSH    AF              ; Save registers.
015F  C5               PUSH    BC              ;
0160  D5               PUSH    DE              ;
0161  E5               PUSH    HL              ;
                ;
0162  21 03B6          LD      HL,ASCDATA      ; Save data address.
0165  06 08            LD      B,08H           ; Loop counter.
0167                ASCDX1:
0167  7E               LD      A,(HL)          ; Get data.
0168  FE 20            CP      20H             ; Control code?
016A  30 02            JR      NC,ASCDX2       ; No.
                ;
016C  3E 2E            LD      A,'.'           ; Change data to '.'.
016E                ASCDX2:
016E  CD 029A          CALL    CONOUTS         ; Display data.
0171  23               INC     HL              ;
0172  10 F3            DJNZ    ASCDX1          ; Loop 8 times.
                ;
0174  0E 0D            LD      C,CR            ; Carriage return.
0176  CD EB0C          CALL    CONOUT          ;
0179  0E 0A            LD      C,LF            ; Line feed.
017B  CD EB0C          CALL    CONOUT          ;
                ;
017E  E1               POP     HL              ; Restore registers.
017F  D1               POP     DE              ;
0180  C1               POP     BC              ;
0181  F1               POP     AF              ;
0182  C9               RET                     ;
                ;
                ;
                ;
0183                ASCDSP2:
0183  F5               PUSH    AF              ; Save registers.
0184  C5               PUSH    BC              ;
0185  D5               PUSH    DE              ;
0186  E5               PUSH    HL              ;
                ;
0187  21 0000          LD      HL,0000H        ; Set binary data to HL
018A  6F               LD      L,A             ;  A --> HL
018B  18 16            JR      ASCD42          ;
                ;
                ;
                ;
018D                ASCDSP4:
018D  F5               PUSH    AF              ; Save registers.
018E  C5               PUSH    BC              ;
018F  D5               PUSH    DE              ;
0190  E5               PUSH    HL              ;
                ;
0191  11 1000          LD      DE,4096         ; 1000H
0194  CD 01BB          CALL    ASCD45          ; Get 16**3
0197  CD 029A          CALL    CONOUTS         ; Display data.
                ;
019A  11 0100          LD      DE,256          ; 100H
019D  CD 01BB          CALL    ASCD45          ; Get 16**2
01A0  CD 029A          CALL    CONOUTS         ; Display data.
                ;
01A3                ASCD42:
01A3  11 0010          LD      DE,16           ; 10H
01A6  CD 01BB          CALL    ASCD45          ; Get 16**1
01A9  CD 029A          CALL    CONOUTS         ; Display data.
                ;
01AC  7D               LD      A,L             ; Get 16**0
01AD  C6 30            ADD     A,30H           ;  Change to ASCII.
01AF  4F               LD      C,A             ;
01B0  CD 01C6          CALL    ASCD48          ;
01B3  CD 029A          CALL    CONOUTS         ; Display data.
                ;
01B6  E1               POP     HL              ; Save registers.
01B7  D1               POP     DE              ;
01B8  C1               POP     BC              ;
01B9  F1               POP     AF              ;
01BA  C9               RET                     ;
                ;
                ;
                ;
01BB                ASCD45:
01BB  0E 30            LD      C,'0'           ;
                ;
01BD                ASCD46:
01BD  B7               OR      A               ; Reset carry bit.
01BE  ED 52            SBC     HL,DE           ;
01C0  38 03            JR      C,ASCD47        ;
                ;
01C2  0C               INC     C               ; Counter increase.
01C3  18 F8            JR      ASCD46          ; Loop
                ;
01C5                ASCD47:
01C5  19               ADD     HL,DE           ; Restore data
                ;
```

```
01C6                    ASCD48:
01C6    79                      LD      A,C             ;
01C7    FE 3A                   CP      ':'             ; If larger than '9',
01C9    D8                      RET     C               ;  then convert to 'A' -- 'F'.
01CA    C6 07                   ADD     A,'A'-':'-1     ;
                        ;
01CC    C9                      RET                     ;
                        ;
                        ;
                        ;       ************************************************
                        ;               INPUT ADDRESS DATA
                        ;       ************************************************
                        ;
                        ;       NOTE :
                        ;               Get address data routine
                        ;
                        ;       <> entry parameter <>
                        ;               NON
                        ;       <> return parameter <>
                        ;               CY   : Return information
                        ;                  =0 -- Normal end
                        ;                  =1 -- ESC key inputed
                        ;       <> preserved registers <>
                        ;               NON
                        ;
                        ;       CAUTION :
                        ;
01CD                    GETADDR:
01CD    21 02F8                 LD      HL,MSG03        ; Display inputing address message.
01D0    CD 028E                 CALL    DSPMSG          ;
                        ;
01D3    21 03B3                 LD      HL,INCNT        ; Inputed data counter reset.
01D6    36 00                   LD      (HL),00H        ;
01D8    11 03B4                 LD      DE,INDATA       ; Inputed data store area.
                        ;
01DB                    GETA10:
01DB    CD 02AF                 CALL    CONINS          ; Get inputed key.
01DE    FE 03                   CP      STOP            ; STOP?
01E0    37                      SCF                     ;  Set carry flag.
01E1    C8                      RET     Z               ; Yes.
01E2    FE 08                   CP      BS              ; Back space?
01E4    28 18                   JR      Z,GETA20        ; Yes.
01E6    FE 0D                   CP      CR              ; Carriage return?
01E8    28 26                   JR      Z,GETA30        ; Yes.
                        ;
01EA    CD 021D                 CALL    CHKHEX          ; Check HEX data.
01ED    38 EC                   JR      C,GETA10        ; Not hexa data.
                        ;
01EF    4F                      LD      C,A             ;
01F0    7E                      LD      A,(HL)          ; Inputed counter check.
01F1    FE 04                   CP      04H             ; Max 4 character.
01F3    30 E6                   JR      NC,GETA10       ; Character over.
                        ;
01F5    79                      LD      A,C             ;
01F6    CD 029A                 CALL    CONOUTS         ; Display inputed char.
01F9    12                      LD      (DE),A          ; Store data.
01FA    34                      INC     (HL)            ; Counter update.
01FB    13                      INC     DE              ; Pointer update.
01FC    18 DD                   JR      GETA10          ; Loop.
                        ;
01FE                    GETA20:
01FE    AF                      XOR     A               ; Back space process.
01FF    BE                      CP      (HL)            ; No inputed character?
0200    28 D9                   JR      Z,GETA10        ; Yes.
0202    35                      DEC     (HL)            ; Counter decrement.
0203    1B                      DEC     DE              ; Pointer decrement.
0204    3E 08                   LD      A,BS            ; Cursor left.
0206    CD 029A                 CALL    CONOUTS         ;
0209    3E 05                   LD      A,EOL           ; Erase end of line.
020B    CD 029A                 CALL    CONOUTS         ;
020E    18 CB                   JP      GETA10          ;
                        ;
0210                    GETA30:
0210    46                      LD      B,(HL)          ; Carriage return process.
0211    21 03B4                 LD      HL,INDATA       ; Change ASCII to binary.
0214    CD 0262                 CALL    ASCBIN          ;
0217    ED 53 03BE              LD      (ADDR),DE       ; Store converted data.
021B    B7                      OR      A               ; Carry off.
021C    C9                      RET                     ;
                        ;
                        ;
021D                    CHKHEX:
021D    FE 30                   CP      '0'             ; 00H -- 2FH?
021F    D8                      RET     C               ; Yes.
0220    FE 3A                   CP      ':'             ; 30H -- 39H?
0222    3F                      CCF                     ;
0223    D0                      RET     NC              ; Yes.
0224    FE 41                   CP      'A'             ; 3AH -- 40H?
0226    D8                      RET     C               ; Yes.
0227    FE 47                   CP      'G'             ; 'A' -- 'F'?
0229    3F                      CCF                     ;
022A    D0                      RET     NC              ; Yes.
022B    FE 61                   CP      'a'             ; 47H -- 60H?
022D    D8                      RET     C               ; Yes.
022E    FE 67                   CP      'g'             ; 'a' -- 'f'?
0230    3F                      CCF                     ;
0231    C9                      RET                     ;
                        ;
                        ;
```

```
                        ;***********************************************
                        ;       SELECT BANK
                        ;***********************************************
                        ;
                        ;       NOTE :
                        ;               Select bank routine
                        ;
                        ;       <> entry parameter <>
                        ;               NON
                        ;          return parameter <>
                        ;               CY  : Return information
                        ;                  =0 -- Normal end
                        ;                  =1 -- ESC key inputed
                        ;       <> preserved registers <>
                        ;               NON
                        ;
                        ;       CAUTION :
                        ;
0232                    GETBNK:
0232    21 033C                 LD      HL,MSG04        ; Display selecting bank message.
0235    CD 028E                 CALL    DSPMSG          ;
                        ;
0238                    GETB05:
0238    CD EB09                 CALL    CONIN           ; Key in.
023B    FE 03                   CP      STOP            ; STOP?
023D    37                      SCF                     ;
023E    C8                      RET     Z               ; Yes.
                        ;
023F    FE 31                   CP      '1'             ; '1' is system bank.
0241    0E FF                   LD      C,SYSBANK       ;
0243    28 14                   JR      Z,GETB10        ;
0245    FE 32                   CP      '2'             ; '2' is bank 0.
0247    0E 00                   LD      C,BANK0         ;
0249    28 0E                   JR      Z,GETB10        ;
024B    FE 33                   CP      '3'             ; '3' is bank 1.
024D    0E 01                   LD      C,BANK1         ;
024F    28 08                   JR      Z,GETB10        ;
0251    FE 34                   CP      '4'             ; '4' is bank 2.
0253    0E 02                   LD      C,BANK2         ;
0255    28 02                   JR      Z,GETB10        ;
0257    18 DF                   JR      GETB05          ; Other inputed character.
                        ;
0259                    GETB10:
0259    CD 029A                 CALL    CONOUTS         ; display inputed code.
                        ;
025C    79                      LD      A,C             ; Set data.
025D    32 03C0                 LD      (BANK),A        ;
0260    B7                      OR      A               ; Carry off.
0261    C9                      RET                     ;
                        ;
                        ;
                        ;***********************************************
                        ;       CHANGE ASCII TO BINARY
                        ;***********************************************
                        ;
                        ;       NOTE :
                        ;               Change ASCII HEX data to binary data.
                        ;
                        ;       <> entry parameter <>
                        ;               HL : ASCII data top address.
                        ;               B  : Data count
                        ;       <> return parameter <>
                        ;               DE : Binary data
                        ;       <> preserved registers <>
                        ;               NON
                        ;
                        ;       CAUTION :
                        ;
0262                    ASCBIN:
0262    11 0000                 LD      DE,0000H
0265    78                      LD      A,B
0266    B7                      OR      A
0267    C8                      RET     Z
                        ;
0268                    ASC10:
0268    C5                      PUSH    BC
0269    7E                      LD      A,(HL)
026A    CD 0284                 CALL    ASC20
026D    4F                      LD      C,A
026E    06 00                   LD      B,00H
0270    EB                      EX      DE,HL
0271    09                      ADD     HL,BC
0272    EB                      EX      DE,HL
0273    C1                      POP     BC
0274    05                      DEC     B
0275    C8                      RET     Z
                        ;
0276    C5                      PUSH    BC
0277    06 04                   LD      B,04H
0279                    ASC15:
0279    B7                      OR      A
027A    CB 13                   RL      E
027C    CB 12                   RL      D
027E    10 F9                   DJNZ    ASC15
                        ;
0280    23                      INC     HL
0281    C1                      POP     BC
0282    18 E4                   JR      ASC10
                        ;
                        ;
0284                    ASC20:
0284    D6 30                   SUB     '0'
0286    FE 0A                   CP      0AH
0288    D8                      RET     C
0289    E6 DF                   AND     11011111B
028B    D6 1B                   SUB     'A'-'0'+10
028D    C9                      RET
                        ;
```

```
                        ;***********************************************
                        ;       MESSAGE DISPLAY
                        ;***********************************************
                        ;
                        ;       NOTE :
                        ;               Display message until found 00H.
                        ;
                        ;       <> entry parameter <>
                        ;               HL : Message data top address.
                        ;       <> return parameter <>
                        ;               NON
                        ;       <> preserved registers <>
                        ;               NON
                        ;
                        ;       CAUTION :
                        ;
028E                    DSPMSG:
028E    7E              LD      A,(HL)
028F    B7              OR      A
0290    C8              RET     Z
                        ;
0291    4F              LD      C,A
0292    E5              PUSH    HL
0293    CD EB0C         CALL    CONOUT
0296    E1              POP     HL
0297    23              INC     HL
0298    18 F4           JR      DSPMSG
                        ;
                        ;***********************************************
                        ;       CONSOLE OUT
                        ;***********************************************
                        ;
                        ;       NOTE :
                        ;
                        ;       * entry parameter <>
                        ;               A  : Console out data
                        ;       <> return parameter <>
                        ;               NON
                        ;       <> preserved registers <>
                        ;               All registers
                        ;
                        ;       CAUTION :
                        ;
029A                    CONOUTS:
029A    F5              PUSH    AF
029B    C5              PUSH    BC
029C    D5              PUSH    DE
029D    E5              PUSH    HL
029E    4F              LD      C,A
029F                    CONO10:
029F    CD EB0C         CALL    CONOUT
02A2    E1              POP     HL
02A3    D1              POP     DE
02A4    C1              POP     BC
02A5    F1              POP     AF
02A6    C9              RET
                        ;
02A7                    SPACE:
02A7    F5              PUSH    AF
02A8    C5              PUSH    BC
02A9    D5              PUSH    DE
02AA    E5              PUSH    HL
02AB    0E 20           LD      C,20H
02AD    18 F0           JR      CONO10
                        ;
                        ;***********************************************
                        ;       CONSOLE IN
                        ;***********************************************
                        ;
                        ;       NOTE :
                        ;
                        ;       <> entry parameter <>
                        ;               NON
                        ;       * return parameter <>
                        ;               A  : Console in data
                        ;       <> preserved registers <>
                        ;               All registers without AF
                        ;
                        ;       CAUTION :
                        ;
02AF                    CONINS:
02AF    C5              PUSH    BC
02B0    D5              PUSH    DE
02B1    E5              PUSH    HL
02B2    CD EB09         CALL    CONIN
02B5    E1              POP     HL
02B6    D1              POP     DE
02B7    C1              POP     BC
02B8    C9              RET
                        ;
02B9                    MSG01:
02B9    53 74 61 72     DB      'Start dump program',CR,LF
02BD    74 20 64 75
02C1    6D 70 20 70
02C5    72 6F 67 72
02C9    61 6D 0D 0A
02CD    00              DB      00H
02CE                    MSG02:
02CE    0D 0A           DB      CR,LF
02D0    0D 0A           DB      CR,LF
02D2    41 64 64 72     DB      'Addr 00 01 02 03 04 05 06 07  ASCII',CR,LF
02D6    20 30 30 20
02DA    30 31 20 30
02DE    32 20 30 33
02E2    20 30 34 20
```

```
02E6    30 35 20 30
02EA    36 20 30 37
02EE    20 20 41 53
02F2    43 49 49 0D
02F6    0A
02F7    00                          DB      00H
02F8                        MSG03:
02F8    0D 0A                       DB      CR,LF
02FA    49 6E 70 75                 DB      'Input dump start address (hexa data)',CR,LF
02FE    74 20 64 75
0302    6D 70 20 73
0306    74 61 72 74
030A    20 61 64 64
030E    72 65 73 73
0312    20 28 68 65
0316    78 61 20 64
031A    61 74 61 29
031E    0D 0A
0320    20 20 28 45                 DB      '  (Exit by pressing STOP)',CR,LF
0324    78 69 74 20
0328    62 79 20 70
032C    72 65 73 73
0330    69 6E 67 20
0334    53 54 4F 50
0338    29 0D 0A
033B    00                          DB      00H
033C                        MSG04:
033C    0D 0A                       DB      CR,LF
033E    53 65 6C 65                 DB      'Select dump bank',CR,LF
0342    63 74 20 64
0346    75 6D 70 20
034A    62 61 6E 6B
034E    0D 0A
0350    20 20 31 20                 DB      '  1 -- System bank',CR,LF
0354    2D 2D 20 53
0358    79 73 74 65
035C    6D 20 62 61
0360    6E 6B 0D 0A
0364    20 20 32 20                 DB      '  2 -- Bank 0 (RAM)',CR,LF
0368    2D 2D 20 42
036C    61 6E 6B 20
0370    30 20 28 52
0374    41 4D 29 0D
0378    0A
0379    20 20 33 20                 DB      '  3 -- Bank 1',CR,LF
037D    2D 2D 20 42
0381    61 6E 6B 20
0385    31 0D 0A
0388    20 20 34 20                 DB      '  4 -- Bank 2',CR,LF
038C    2D 2D 20 42
0390    61 6E 6B 20
0394    32 0D 0A
0397    20 20 28 45                 DB      '  (Exit by pressing STOP)',CR,LF
039B    78 69 74 20
039F    62 79 20 70
03A3    72 65 73 73
03A7    69 6E 67 20
03AB    53 54 4F 50
03AF    29 0D 0A
03B2    00                          DB      00H
03B3                        ;
03B3                        INCNT:  DS      1
03B4                        INDATA: DS      2
03B6                        ASCDATA:DS      8
03BE                        ADDR:   DS      2
03C0                        BANK:   DS      1
                            ;
                                    END
```

(26) STORX

Function: Writes one byte of data to the specified address on the specified bank.

Entry address: WBOOT + 5DH or ØEB60H

Entry parameter: A = Data to be written
C = Bank to which data is to be written
= ØFFH: System bank
= ØØH: Bank Ø
= Ø1H: Bank 1
= Ø2H: Bank 2
HL = Address of the data to be written

Return parameter: None.

Explanation:
All registers retain the previous values.

Since STORX makes no parameter check, normal operation is not guaranteed if a value other than -1 to 2 is specified in C.

Note:
The C register should be set to ØØH because data can be written only in RAM.

See also: Section 4.4, "Bank Switching"

(27) LDIRX

Function: Transfers the specified number of data bytes on the specified bank onto bank Ø.

Entry address: WBOOT + 6ØH or ØEB63H

Entry parameter: A = Transferred bank
= ØFFH: System bank
= ØØH: Bank Ø
= Ø1H: Bank 1
= Ø2H: Bank 2
HL = Starting address of the data to be transferred
DE = Starting address of the destination to which data is to be transferred
BC = Number of bytes to be transferred

Return parameter: BC = ØØØØH
DE = (DE on entry) + (BC on entry)
HL = (HL on entry) + (BC on entry)

Explanation:
Since LDIRX makes no error check, normal operation is not guaranteed if a value other than -1 to 2 is specified in C. This routine is equivalent to the LDIR instruction with a bank switching capability.

See also: Section 4.4, "Bank Switching"

(28) JUMPX

Function: Causes the CPU to jump to the specified bank address.

Entry address: WBOOT + 63H or ØEB66H

Entry parameter: IX = Jump address
DISBNK (ØF52EH) = Destination bank number
= ØFFH: System bank
= ØØH: Bank Ø
= Ø1H: Bank 1
= Ø2H: Bank 2

Return parameter: None.

Explanation:
All registers retain the previous values when control is transferred to the destination of jump.

Since JUMPX makes no parameter check, normal operation is not guaranteed if a value other than -1 to 2 is specified in DISBNK.

Notes:
The system-supplied stack in the BIOS is used when this routine is called. Consequently, the called routine must reserve a stack for itself. Otherwise, the CPU will hang up when the routine calls BIOS or BDOS.

Since the system is in a state in which it is still executing BIOS, the user program should forces the system to exit that state by calling RSPSTBIOS (ØFF96H).

See also: Section 4.4, "Bank Switching"

(29) CALLX

Function: Calls the specified bank address.

Entry address: WBOOT + 66H or ØEB69H

Entry parameter: IX = Called routine address
DISBNK (ØF52EH) = Called bank number
= ØFFH: System bank
= ØØH: Bank Ø
= Ø1H: Bank 1
= Ø2H: Bank 2

Return parameter:
All registers except IX and IY retain the previous values.

Explanation:
When control is transferred to the called routine, all registers retains the values set up when CALLX is called.

Since CALLX makes no parameter check, normal operation is not guaranteed if a value other than -1 to 2 is specified in DISBNK.

Notes:

When this routine is called, the system-supplied stack in the BIOS is used and the system is in a state in which it is still executing BIOS. Accordingly, the user must take the notes given in the JUMPX description into consideration.

If the called program reserves its own stack, it must restore the original stack when returning control.

This routine should be used to call utility routines in the system ROM.

See also: Section 4.4, "Bank Switching"

(30) GETPFK

Function: Reads the character string from a specified PF key.

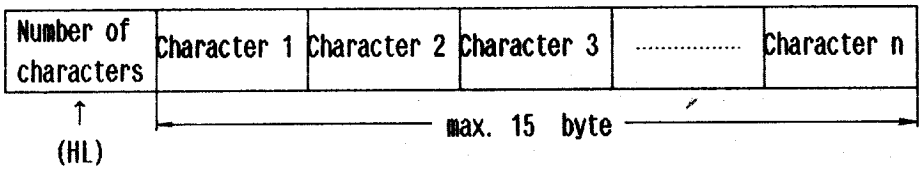
Entry address: WBOOT + 69H or 0EB6CH

Entry parameter: C = PF key number
= 00H - 09H: Specifies a PF key on the standard keyboard.
= 40H - 7EH: Specifies an item function key.
HL = Starting address of the 16-byte buffer into which the character string is to be read.

Return parameter: HL = Retains the previous value.

Explanation:

GETPFK gets the character string defined for a given PF key. The format of the character string is shown below.




"Number of characters" indicates the number of characters in the string defined for the PF key. It must be in the range from 00H to 0FH. 00H indicates that no string is defined for the PF key.

Related function: PUTPFK

See also: Section 3.5, "Keyboard"

```
                        ;*************************************************
                        ;       BIOS GETPFK SAMPLE PROGRAM
                        ;*************************************************
                        ;
                        ;       NOTE :
                        ;               This sample program is displaying present
                        ;               defined function key list.
                        ;
                        ;       <> assemble condition <>
                        ;
                        ;       .Z80
                        ;
                        ;       <> loading address <>
                        ;
                        ;       .PHASE  100H
                        ;
                        ;       <> constant values <>
                        ;
EB03                    WBOOT           EQU     0EB03H  ; WBOOT entry address.
EB09                    CONIN           EQU     0EB09H  ; CONIN entry address.
EB0C                    CONOUT          EQU     0EB0CH  ; CONOUT entry address.
EB6C                    GETPFK          EQU     0EB6CH  ; GETPFK entry address.
                        ;
1000                    MAINSP          EQU     1000H   ; Stack pointer.
                        ;
000D                    CR              EQU     0DH     ; Carriage return code.
000A                    LF              EQU     0AH     ; Line feed code.
0003                    BREAK           EQU     03H     ; STOP code.
                        ;
                        ;*************************************************
                        ;       MAIN PROGRAM
                        ;*************************************************
                        ;
                        ;       NOTE :
                        ;
0100                    START:
0100    31 1000                 LD      SP,MAINSP       ; Set stack pointer.
                        ;
0103    06 0A                   LD      B,09H-00H+1     ; Loop counter.
0105    0E 00                   LD      C,00H           ; PF1 to PF10
                        ;
0107                    LOOP1:
0107    C5                      PUSH    BC              ; Save counter & PFK code.
0108    21 014E                 LD      HL,PFKBUF       ; PFK data reading area.
010B    CD EB6C                 CALL    GETPFK          ; Get PFK data.
010E    CD 012A                 CALL    PFKDSP          ; Display PFK data.
0111    C1                      POP     BC              ; Restore counter & PFK code.
0112    0C                      INC     C               ; Increase PFK code.
0113    10 F2                   DJNZ    LOOP1           ; Loop 10 times.
                        ;
0115    06 3F                   LD      B,7EH-40H+1     ; Loop counter.
0117    0E 40                   LD      C,40H           ; ITEM FK code.
                        ;
0119                    LOOP2:
0119    C5                      PUSH    BC              ; Save counter & ITEM FK code.
011A    21 014E                 LD      HL,PFKBUF       ; PFK data reading area.
011D    CD EB6C                 CALL    GETPFK          ; Get PFK data.
0120    CD 012A                 CALL    PFKDSP          ; Display PFK data.
0123    C1                      POP     BC              ; Restore counter & ITEM FK code.
0124    0C                      INC     C               ; Increase ITEMFK code.
0125    10 F2                   DJNZ    LOOP2           ; Loop 63 times.
                        ;
0127    C3 EB03                 JP      WBOOT           ; End of main program.
                        ;
                        ;*************************************************
                        ;       DISPLAY FUNCTION KEY DATA
                        ;*************************************************
                        ;
                        ;       NOTE :
                        ;               Display the function key data and
                        ;               when input any key except STOP, return.
                        ;               If STOP key is pressed, then WBOOT.
                        ;
                        ;       <> entry parameter <>
                        ;               HL : PFK string top address.
                        ;       <> return parameter <>
                        ;               NON
                        ;       <> preserved registers <>
                        ;               NON
                        ;
                        ;       CAUTION :
                        ;
012A                    PFKDSP:
012A    7E                      LD      A,(HL)          ; Get string length.
012B    B7                      OR      A               ; Length is 0?
012C    28 0D                   JR      Z,PFKEND        ; Yes.
                        ;
012E    47                      LD      B,A             ; String length --> counter
012F    23                      INC     HL              ; String pointer 1 increase.
                        ;
0130                    PFKLOOP:
0130    E5                      PUSH    HL              ; Save pointer.
0131    C5                      PUSH    BC              ; Save counter.
0132    4E                      LD      C,(HL)          ; Get display data.
0133    CD EB0C                 CALL    CONOUT          ; Display the data.
0136    C1                      POP     BC              ; Restore counter.
0137    E1                      POP     HL              ; Restore pointer.
0138    23                      INC     HL              ; Pointer update.
0139    10 F5                   DJNZ    PFKLOOP         ; Loop (by string length)
                        ;
013B                    PFKEND:
013B    0E 0D                   LD      C,CR            ; Display CR & LF.
013D    CD EB0C                 CALL    CONOUT          ;
0140    0E 0A                   LD      C,LF            ;
0142    CD EB0C                 CALL    CONOUT          ;
                        ;
```

```
0145    CD EB09                 CALL    CONIN           ; Get any inputed key.
0148    FE 03                   CP      BREAK           ; STOP code?
014A    CA EB03                 JP      Z,WBOOT         ; Yes. then WBOOT.
014D    C9                      RET                     ; else return.
                        ;
                        ;       ************************************************
                        ;       WORK AREA
                        ;       ************************************************
                        ;
014E                    PFKBUF:
014E                            DS      16              ; PFK data reading area.
                        ;
                                END
```

(31) PUTPFK

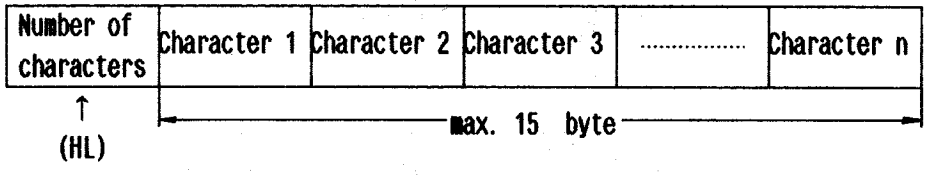
Function: Defines a character string for a PF key.

Entry address: WBOOT + 6CH or ØEB6FH

Entry parameter: C = PF key number or control information
= ØØH - Ø9H: Specifies a PF key on the standard keyboard.
= 4ØH - 7EH: Specifies an item function key.
= ØFFH: Cancels all item functions.
= ØFEH: Resets the item flag.
= ØFDH: Sets the item flag.
HL = Starting address of the character string to be assigned

Return parameter: HL = Retains the previous value.

Explanation:
PUTPFK assigns a character string to a PF key in the 16-byte format shown below. The maximum string length is 15 characters.




"Number of characters" specifies the number of characters to be assigned to the specified PF key. It must be in the range from ØØH to ØFH ØØH indicates that no string is defined for the PF key.

Item functions are enabled if the item flag is set. This ensures that the standard keyboard keys whose number are in the range 4ØH to 7EH can be used as function keys by assigning item functions with the item flag set.

See also: Section 3.5, "Keyboard"

Reference:

The system uses the following areas for setting function keys:

YPFRSTR (0F012H) 2 bytes
- PF key table starting address

The default value is WPFKTBL (0F545H). The YPFRSTR must be located at address 8000H or higher.

ITEMFLG (0F01DH) 1 byte
- Item mode flag
  = 00H: Normal mode
  = 80H: Item mode

WPFKTBL (0F545H) 160 bytes
- PF key table

| | 1 byte | 15 byte |
|---|---|---|
| PF 1 | Number of Characters | Character string |
| PF 2 | Number of Characters | Character string |
| ⋮ | ⋮ | ⋮ |
| PF10 | Number of Characters | Character string |

RWITEMTOP (0CC00H) 1024 bytes
- Item function key table

| | 1 byte | 15 byte |
|---|---|---|
| 40H | Number of Characters | Character string |
| 41H | Number of Characters | Character string |
| ⋮ | ⋮ | ⋮ |
| 7EH | Number of Characters | Character string |
| 7FH | (Not used) | (Not used) |

PFKTAB (0F02DH) 160 bytes
- PF key initial data

The table format is the same as that of WPFKTBL. The contents of this area are copied into WPFKTBL by BOOT or WBOOT that is invoked during execution of a nonresident program.

The PF keys are initialized as follows:

```
PF1: dir ␣      PF5: basic ␣        PF9: filink (CR)↵
  2: type ␣       6: config (CR)↵    10:
  3: stat ␣       7: submit ␣
  4: pip          8: term (CR)↵
```

The initial values of the item function keys (loaded in RWITEMTOP) are listed below:

The numbers are item function key numbers, and the values are the Character codes of the characters for which the keys are defined.

| Number | Value | Number | Value | Number | Value | Number | Value | Number | Value | Number | Value | Number | Value | Number | Value |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| 40H | 40H | 48H | 48H | 50H | 50H | 58H | 58H | 60H | 1BH | 68H | 21H | 70H | 29H | 78H | 3AH |
| 41H | 41H | 49H | 49H | 51H | 51H | 59H | 59H | 61H | 13H | 69H | 22H | 71H | 5EH | 79H | 3BH |
| 42H | 42H | 4AH | 4AH | 52H | 52H | 5AH | 5AH | 62H | 00H | 6AH | 23H | 72H | 5FH | 7AH | 3CH |
| 43H | 43H | 4BH | 4BH | 53H | 53H | 5BH | 5BH | 63H | 12H | 6BH | 24H | 73H | 60H | 7BH | 3DH |
| 44H | 44H | 4CH | 4CH | 54H | 54H | 5CH | 5CH | 64H | 0BH | 6CH | 25H | 74H | 7BH | 7CH | 3FH |
| 45H | 45H | 4DH | 4DH | 55H | 55H | 5DH | 5DH | 65H | 0CH | 6DH | 26H | 75H | 7CH | 7DH | 3FH |
| 46H | 46H | 4EH | 4EH | 56H | 56H | 5EH | 20H | 66H | 7FH | 6EH | 27H | 76H | 7DH | 7EH | 20H |
| 47H | 47H | 4FH | 4FH | 57H | 57H | 5FH | | 67H | 08H | 6FH | 28H | 77H | 7EH | 7FH | |

Key numbers 40H through 7FH for the standard keyboard are all cleared (undefined).

(32) READSW

Function: Reads switch status.

Entry address: WBOOT + 6FH or 0EB72H

Entry parameter: C = Selection number
= 02H: Read DIP switch settings.
= 04H: Read power switch status.

Return parameter: A = DIP switch settings or power switch status.

Explanation:

When C = 02H
The DIP switch settings are read into the A register in the following format:

| DIP sw | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 |
|---|---|---|---|---|---|---|---|---|
| A reg. | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |

A 0 in an A register bit corresponds to the OFF state and a 1 to the ON state. It is set to 1 if the corresponding switch is ON.

When C = 04H
A = 00H: Power switch is off.
= 01H: Power switch is on.

When C = other than 02H and 04H
READSW does nothing.

See also: Section 7.2, "DIP Switches"

```
                        ;       ************************************************
                        ;               READ DIP SWICH & POWER SWITCH
                        ;       ************************************************
                        ;
                        ;       NOTE :
                        ;               This sample program is reading switch
                        ;               status and displaying it.
                        ;       <> assemble condition <>
                        ;
                                .Z80
                        ;
                        ;       <> loading address <>
                        ;
                                .PHASE  100H
                        ;
                        ;       <> constant values <>
                        ;
                        ;       BIOS entry
                        ;
EB03                    WBOOT           EQU     0EB03H          ; Warm Boot entry
EB0C                    CONOUT          EQU     WBOOT   +09H    ; Console out entry
EB72                    READSW          EQU     WBOOT   +6FH    ; Read switch entry
                        ;
0009                    TAB             EQU     009H    ; Tab code
000A                    LF              EQU     00AH    ; Line feed
000C                    CLS             EQU     00CH    ; Clear screen
000D                    CR              EQU     00DH    ; Carriage return
001B                    ESC             EQU     01BH    ; Escape
00E2                    ON              EQU     0E2H    ; ON code
00E3                    OFF             EQU     0E3H    ; OFF code
                        ;
                        ;       ************************************************
                        ;               MAIN PROGRAM
                        ;       ************************************************
                        ;
                        ;       NOTE :
                        ;
0100                    MAIN:
0100    31 1000                 LD      SP,1000H        ; Set stack pointer.
                        ;
0103    CD 011C                 CALL    SETCHAR         ; Set user-defined char.
                        ;
0106    0E 02                   LD      C,02H           ; Read dip switch.
0108    CD EB72                 CALL    READSW          ;
010B    CD 0144                 CALL    DSET            ; Set dip switch data.
                        ;
010E    0E 04                   LD      C,04H           ; Read power switch.
0110    CD EB72                 CALL    READSW          ;
0113    CD 0151                 CALL    PSET            ; Set power switch ata.
                        ;
0116    CD 0163                 CALL    DSPMSG          ; Display switch status.
                        ;
0119    C3 EB03                 JP      WBOOT           ; End
                        ;
                        ;
                        ;       ************************************************
                        ;               SET USER DEFINE CHARACTER
                        ;       ************************************************
                        ;
                        ;       NOTE :
                        ;               Set user defined character.
                        ;               E0H & E1H is used.
                        ;
                        ;       <> entry parameter <>
                        ;               NON
                        ;          return parameter <>
                        ;               NON
                        ;          preserved registers <>
                        ;               NON
                        ;
                        ;       CAUTION :
                        ;
011C                    SETCHAR:
011C    21 012D                 LD      HL,CHARDATA     ; Data top address.
011F    46                      LD      B,(HL)          ; Get data counter.
0120    23                      INC     HL              ;
                        ;
0121                    SET10:
0121    4E                      LD      C,(HL)          ; Get conout data.
0122    E5                      PUSH    HL              ; Save registers.
0123    C5                      PUSH    BC              ;
0124    CD EB0C                 CALL    CONOUT          ; Console out.
0127    C1                      POP     BC              ; Restore registers.
0128    E1                      POP     HL              ;
0129    23                      INC     HL              ; Pointer update
012A    10 F5                   DJNZ    SET10           ; Loop
012C    C9                      RET                     ;
                        ;
012D                    CHARDATA:
012D    16                      DB      22              ; Data number.
012E    1B E0 E2                DB      ESC,0E0H,0E2H   ; 0E2H char data.
0131    3F 3F 3F 3F             DB      3FH,3FH,3FH,3FH
0135    21 21 21 3F             DB      21H,21H,21H,3FH
                        ;
0139    1B E0 E3                DB      ESC,0E0H,0E3H   ; 0E3H char data.
013C    3F 21 21 21             DB      3FH,21H,21H,21H
0140    3F 3F 3F 3F             DB      3FH,3FH,3FH,3FH
                        ;
                        ;
```

```
                        ;       ************************************************
                        ;               SET DIP SWITCH DATA
                        ;       ************************************************
                        ;
                        ;       NOTE :
                        ;               Set dip switch data to message area.
                        ;
                        ;       <> entry parameter <>
                        ;               A  : Dip switch data.
                        ;       <> return parameter <>
                        ;               NON
                        ;       <> preserved registers <>
                        ;               NON
                        ;
                        ;       CAUTION :
                        ;
0144                    DSET:
0144    21 0190                 LD      HL,DIPSW        ; Data setting addr.
0147    06 08                   LD      B,8             ; Loop counter.
                        ;
0149                    DSET10:
0149    07                      RLCA                    ; Set switch status to CY.
014A    CD 0159                 CALL    SETONOFF        ; Set data.
014D    23                      INC     HL              ; Next setting address.
014E    10 F9                   DJNZ    DSET10          ; Loop.
                        ;
0150    C9                      RET                     ;
                        ;
                        ;       ************************************************
                        ;               SET POWER SWITCH DATA
                        ;       ************************************************
                        ;
                        ;       NOTE :
                        ;               Set power switch data to message area.
                        ;
                        ;       <> entry parameter <>
                        ;               A  : Power switch data.
                        ;       <> return parameter <>
                        ;               NON
                        ;       <> preserved registers <>
                        ;               NON
                        ;
                        ;       CAUTION ;
                        ;
0151                    PSET:
0151    21 01B4                 LD      HL,POWSW        ; Data setting address.
0154    0F                      RRCA                    ; Switch data --> CY
0155    CD 0159                 CALL    SETONOFF        ; Set data.
0158    C9                      RET                     ;
                        ;
                        ;       ************************************************
                        ;               SELECT BDOS ERROR RECOVERY
                        ;       ************************************************
                        ;
                        ;       NOTE :
                        ;               Select BDOS error recovery type.
                        ;                1. Using SETERR and RSTERR
                        ;                2. Replacing BDOS error vector
                        ;
                        ;       <> entry parameter <>
                        ;               CY : ON/OFF information.
                        ;                  =1 -- ON
                        ;                  =0 -- OFF
                        ;       <> return parameter <>
                        ;               NON
                        ;       <> preserved registers <>
                        ;               BC,DE,HL
                        ;
                        ;       CAUTION :
                        ;
0159                    SETONOFF:
0159    C5                      PUSH    BC              ; Save register.
015A    06 E2                   LD      B,ON            ; ON code --> B
015C    38 02                   JR      C,MSET10        ; ON.
015E    06 E3                   LD      B,OFF           ; Set OFF code.
0160                    MSET10:
0160    70                      LD      (HL),B          ; Set data.
0161    C1                      POP     BC              ; Restore register.
0162    C9                      RET                     ;
                        ;
                        ;       ************************************************
                        ;               DISPLAY SWITCH MESSAGE
                        ;       ************************************************
                        ;
                        ;       NOTE :
                        ;
                        ;       <> entry parameter <>
                        ;               NON
                        ;       <> return parameter <>
                        ;               NON
                        ;       <> preserved registers <>
                        ;               NON
                        ;
                        ;       CAUTION :
                        ;
0163                    DSPMSG:
0163    21 0172                 LD      HL,MSG          ; Message data top addr.
0166                    DSP10:
0166    7E                      LD      A,(HL)          ; Get data.
0167    B7                      OR      A               ; End of data?
0168    C8                      RET     Z               ; Yes.
                        ;
0169    4F                      LD      C,A             ; Set parameter.
016A    E5                      PUSH    HL              ; Save register.
016B    CD EB0C                 CALL    CONOUT          ; Console out.
```

```
016E   E1                      POP     HL              ; Restore register.
016F   23                      INC     HL              ; Pointer update.
0170   18 F4                   JR      DSP10           ; Loop.
                       ;
                       ;       Message and work area
                       ;
0172                   MSG:
0172   0C                      DB      CLS
0173   09 09 4F 4E             DB      TAB,TAB,'ON 87654321',CR,LF
0177   20 38 37 36
017B   35 34 33 32
017F   31 0D 0A
0182   44 49 50 20             DB      'DIP SWITCH',TAB,'
0186   53 57 49 54
018A   43 48 09 20
018E   20 20
0190                   DIPSW:  DS      8
0198   0D 0A                   DB      CR,LF
019A   09 09 4F 46             DB      TAB,TAB,'OFF',CR,LF
019E   46 0D 0A
                       ;
01A1   09 09 4F 4E             DB      TAB,TAB,'ON',CR,LF
01A5   0D 0A
01A7   50 4F 57 45             DB      'POWER SWITCH',TAB
01AB   52 20 53 57
01AF   49 54 43 48
01B3   09
01B4                   POWSW:  DS      1
01B5   0D 0A                   DB      CR,LF
01B7   09 09 4F 46             DB      TAB,TAB,'OFF',CR,LF
01BB   46 0D 0A
01BE   00                      DB      00H
                       ;
                               END
```

(33) RDVRAM

Function: Reads data from the virtual screen.

Entry address: WBOOT + 75H or ØEB78H

Entry parameter: B = Starting column number in which the read is to begin
C = Starting row number in which the read is to begin
DE = Number of characters to be read
HL = Address of the area for storing the read data (the area must not be smaller than the byte count specified in DE)

Return parameter: A = Return information
= ØØH: Normal termination
= Ø1H: Read error
= ØFFH: Parameter error

Explanation:

The range of values allowed for the B and C registers varies with the size of the virtual screen.

1 <= B <= Number of columns in virtual screen (4Ø or 8Ø)
1 <= C <= Number of lines in virtual screen (8 - 25)

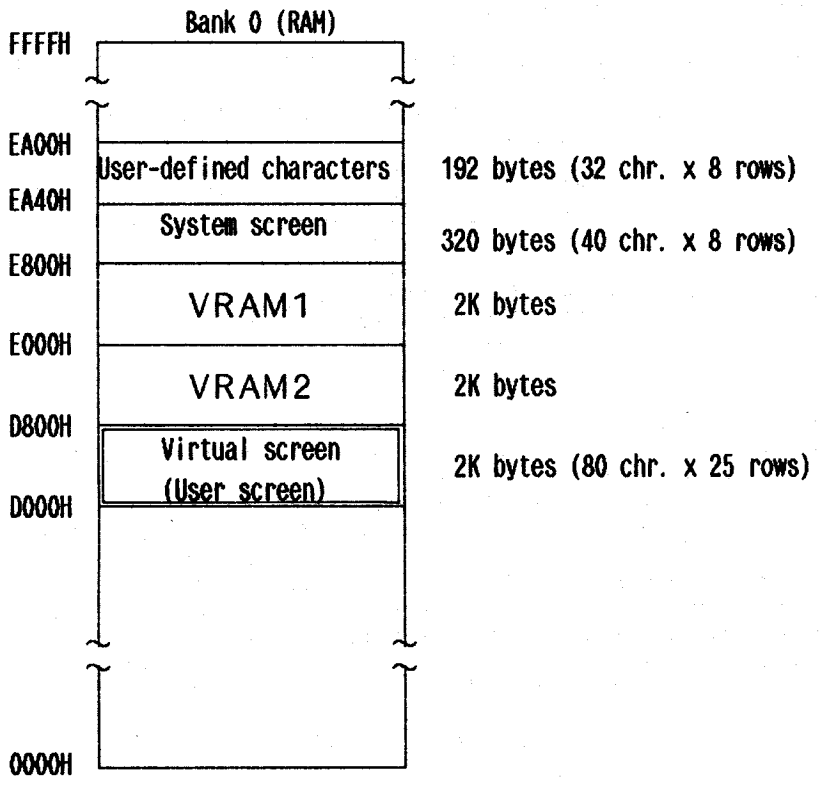
When the number specified in DE is too large and display extends beyond the screen, 2ØH (space) codes are returned as extra characters until the number of the returned characters matches the value specified in DE. If this condition occurs, the A register is loaded with a return code of Ø1H.

RDVRAM returns the read data in ASCII codes.

See also: Section 3.6, "LCD Display"

Reference:

The virtual screen (user screen) is located in memory as shown below:




Note:

In the system screen mode, RDVRAM reads data from the system screen.

(34) MCMTX

Function: Performs a microcassette function.

Entry address: WBOOT + 78H or ØEB7BH

Entry parameter:
MCMTX executes one of the 23 microcassette functions that is specified by the B register. The microcassette functions are listed below.

| B reg. | Function | Description |
|---|---|---|
| 00H | MIRDST | Read tape status |
| 01H | MIRDCT | Read tape counter |
| 02H | MISTCT | Set tape counter |
| 03H | MISTOP | Stop motor |
| 04H | MIPLAY | Start motor in PLAY mode |
| 05H | MIREC | Start motor in RECORD mode |
| 06H | MIFF | Start motor in FAST FEED mode |
| 07H | MISREW | Start motor in SLOW REWIND mode |
| 08H | MIREW | Start motor in REWIND mode |
| 09H | MIFFTE | Fast feed to tape end |
| 0AH | MIRWTT | Rewind to tape top |
| 0BH | MIHDON | Turn head on |
| 0CH | MIHDOF | Turn head off |
| 0DH | MISKTP | Seek tape |
| 0EH | MISTMP | Set tape move protect counter |
| 0FH | MIRSMP | Reset tape move protect counter |
| 10H | MIRDBL | Read one block |
| 11H | MIWTBL | Write one block |
| 12H | | Undefined |
| 13H | | Undefined |
| 14H | MIGTWD | Get write protection pin status |
| 15H | MILEDON | Turn LED on |
| 16H | MILEDOF | Turn LED off |
| 17H | MISDAT | Save data area |
| 18H | MILDAT | Load data area |

See Section 3.7, "MTOS/MIOS Operations" for detailed information about the individual functions.

(35) POWEROFF

Function: Turns off PINE main power.

Entry address: WBOOT + 7BH or ØEB7EH

Entry parameter: C = Power off mode
= ØØH: Turn off power in continue mode.
= Ø1H: Turn off power in restart mode.

Return parameter: None.

Explanation:
POWEROFF saves the current system status and turns off PINE main power. Once this routine is executed, only the 75Ø8 slave CPU and RAM are supplied with power. The 75Ø8 only updates the clock every 1Ø seconds and is mostly in the sleep state.

Notes:
Reset the continue mode with BIOS CONTINUE when turning off system power in the restart mode. Otherwise, the continue mode specification will be given priority and power may not be turned off in the restart mode.

See Section 2.5, "Power-off" for power-off processing.

See also: Section 2.5, "Power-off."

(36) USERBIOS

Function: Provides an entry point to USERBIOS.

Entry address: WBOOT + 7EH or ØEB81H

Entry parameter: User-supplied

Return parameter: User-supplied

Explanation:
USERBIOS provides an entry point through which the user can call his own BIOS routine. The user BIOS routine itself is located in the user BIOS area.

The starting address of the user-supplied BIOS routine must be set in this entry address.

The default of USERBIOS is the address of a subroutine which contains nothing but a RET instruction.

Note:
USERBIOS is set to the default value by a reset or system initialize.

See also: Section 4.1, "User BIOS"

(37) AUTOST

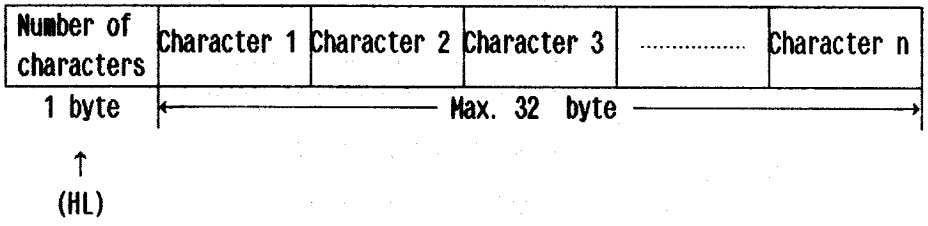
Function: Defines or cancels an auto start string.

Entry address: WBOOT + 81H or ØEB84H

Entry parameter: C = Function
= ØØH: Cancel an auto start string.
= Ø1H: Define an auto start string.
HL = Starting address of the string to be defined (Valid only when C = Ø1H.)

Return parameter: None.

Explanation:
The string format is shown below:




"Number of characters" indicates the length of the specified string and must be in the range from ØØH to 2ØH. ØØH specifies that the auto start string is to be cancelled.

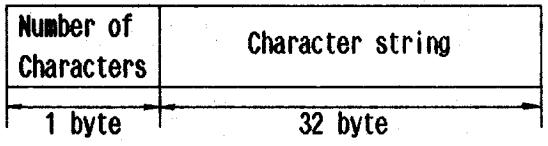
See Section 2.7 for the auto start function.

Note:
When the specified string is shorter than 32 bytes, 2ØH (space) codes are appended to the end of the string to fill the 32-byte field. When a length longer than 32 bytes is specified, the extra bytes are ignored.

See also: Section 2.7, "Auto Start Function"

Reference:
The system uses the following areas for defining an auto start string:

AUTOSTRT (ØF3BDH) 34 bytes
- Auto start string area




"Number of characters" indicates the length of the specified character string. A ØØH indicates that no auto start string is defined. The 34th byte is not used.

```
                        ;       ************************************************
                        ;               SET AUTO START STRING
                        ;       ************************************************
                        ;
                        ;       NOTE :
                        ;               This sample program sets auto start string.
                        ;               If '^' + character are inputed, they are
                        ;               translated into control character.
                        ;
                        ;       <> assemble condition <>
                        ;
                                .Z80
                        ;
                        ;       <> loading address <>
                        ;
                                .PHASE  100H
                        ;
                        ;       <> constant values <>
                        ;
                        ;       BDOS entry
                        ;
0005                    BDOS            EQU     00005H  ; BDOS entry address.
                        ;
0009                    STRING_OUT      EQU     09H     ; BDOS function
000A                    STRING_IN       EQU     0AH     ;
                        ;
                        ;       BIOS entry
                        ;
EB03                    WBOOT           EQU     0EB03H          ; Warm Boot entry
EB84                    AUTOST          EQU     WBOOT   +81H    ; Auto start entry
                        ;
                        ;       System area
                        ;
                        ;       ************************************************
                        ;               MAIN PROGRAM
                        ;       ************************************************
                        ;
                        ;       NOTE :
                        ;               This program sets auto start string,
                        ;               whitch is inputed from keyboard.
                        ;
0100                    MAIN:
0100    31 1000                 LD      SP,1000H        ; Set stack pointer.
                        ;
0103    0E 09                   LD      C,STRING_OUT    ; Console out string.
0105    11 0153                 LD      DE,MSG01        ;  Message address.
0108    CD 0005                 CALL    BDOS            ;
                        ;
010B    0E 0A                   LD      C,STRING_IN     ; Input into console buffer.
010D    11 017D                 LD      DE,IN_BUFF      ;  Buffer address.
0110    CD 0005                 CALL    BDOS            ;
                        ;
0113    CD 0121                 CALL    CHK_CTRL        ; Check control code.
                        ;
0116    21 017E                 LD      HL,IN_BUFF+1    ; Set auto sart string.
0119    0E 01                   LD      C,01H           ;  Set.
011B    CD EB84                 CALL    AUTOST          ;
                        ;
011E    C3 EB03                 JP      WBOOT           ; Program end.
                        ;
                        ;       ************************************************
                        ;               CHANGE ^+CHARACTER TO CONTROL CODE
                        ;       ************************************************
                        ;
                        ;       NOTE :
                        ;
                        ;       <> entry parameter <>
                        ;               NON
                        ;       <> return parameter <>
                        ;               NON
                        ;       <> preserved registers <>
                        ;               NON
                        ;
                        ;       CAUTION :
                        ;
0121                    CHK_CTRL:
0121    21 017E                 LD      HL,IN_BUFF+1    ; Inputed character no. check.
0124    7E                      LD      A,(HL)          ;  Get inputed character no.
0125    B7                      OR      A               ;  No character inputed?
0126    C8                      RET     Z               ;  Yes.
                        ;
0127    4F                      LD      C,A             ; Char. No. --> C
0128    23                      INC     HL              ; Char. data top addr. --> HL
0129    54                      LD      D,H             ; Char. data top addr. --> DE
012A    5D                      LD      E,L             ;
                        ;
012B                    CHK10:
012B    1A                      LD      A,(DE)          ; Get character.
012C    FE 5E                   CP      '^'             ;  '^' code?
012E    28 07                   JR      Z,CHK20         ; Yes.
                        ;
0130    77                      LD      (HL),A          ; Set data.
0131    13                      INC     DE              ; Get pointer update.
0132    23                      INC     HL              ; Put pointer update.
0133    0D                      DEC     C               ; Counter decrement.
0134    C8                      RET     Z               ; End of char.
0135    18 F4                   JR      CHK10           ; Loop.
                        ;
0137                    CHK20:
0137    13                      INC     DE              ; Get pointer increment.
0138    0D                      DEC     C               ; Counter check.
0139    C8                      RET     Z               ; No char. exists.
                        ;
013A    1A                      LD      A,(DE)          ; Get character.
013B    D6 40                   SUB     40H             ; 40H to 7FH?
013D    38 EC                   JR      C,CHK10         ; No.
013F    FE 40                   CP      40H             ;
```

```
0141    30 E8                       JR      NC,CHK10        ; No.
                                    ;
0143    E6 DF                       AND     11011111B       ; 00H to 1FH
0145    77                          LD      (HL),A          ; Set new data.
0146    13                          INC     DE              ; Get pointer update.
0147    23                          INC     HL              ; Put pointer update.
0148    3A 017E                     LD      A,(IN_BUFF+1)   ; Character No. decrement.
014B    3D                          DEC     A               ;
014C    32 017E                     LD      (IN_BUFF+1),A   ;
014F    0D                          DEC     C               ; Character remain?
0150    C8                          RET     Z               ; No.
0151    18 D8                       JR      CHK10           ; Loop until end of char.
                                    ;
                                    ;       Message and work area.
                                    ;
0153                                MSG01:
0153    49 4E 50 55                 DB      'INPUT AUTO START STRING (Max 32 char.)'
0157    54 20 41 55
015B    54 4F 20 53
015F    54 41 52 54
0163    20 53 54 52
0167    49 4E 47 20
016B    28 4D 61 78
016F    20 33 32 20
0173    63 68 61 72
0177    2E 29
0179    0D 0A 09 24                 DB      0DH,0AH,09H,'$'
                                    ;
017D                                IN_BUFF:
017D    20                          DB      32              ; Max input character No.
017E                                DS      1               ;  Inputed char. No. area.
017F                                DS      32              ;  Inputed data area.
                                    ;
                                    END
```

(38) RESIDENT

Function: Turns on or off the resident function.

Entry address: WBOOT + 84H or ØEB87H

Entry parameter: C = Controls the resident function.
= ØØH: Disable resident.
= Ø1H: Enable resident.

Return parameter: None.

Explanation:
RESIDENT enables or disables the resident function. For details of this function, see Section 4.5.

Reference:
The system uses the following area for resident control:

RESEXQ (ØEF28H) 1 byte
- Resident execution flag
= ØØH: Resident function not specified.
= Nonzero: Resident function specified.

See also: Section 4.5, "Resident Processing"

(39) CONTINUE

Function: Sets or resets the continue mode.

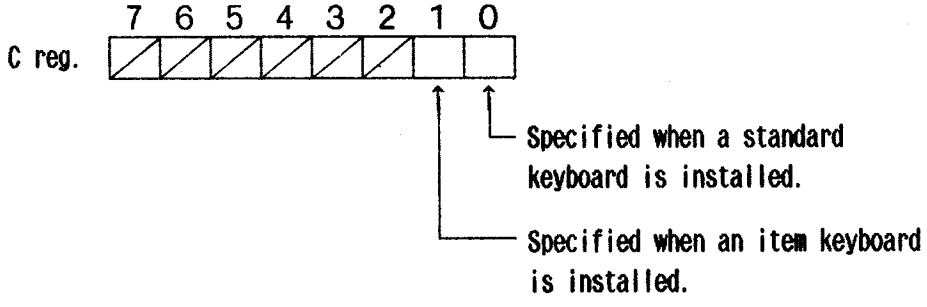
Entry address: WBOOT + 87H or ØEB8AH

Entry parameter: C = Controls the continue mode.

Return parameter: None.

Explanation:
CONTINUE sets or resets the continue flag.

The entry parameter has the following format:




A 1 in bit Ø or 1 sets the continue mode and a Ø resets the continue mode.

The continue mode may be set for the standard and item keyboards independently.

Reference:

The system uses the following areas to control the continue mode:

FRCECNTN (ØF311H) 1 byte
- Continue flag for the standard keyboard
  = ØØH: Restart mode (default)
  = Nonzero: Continue mode

IFRCECNT (ØF312H) 1 byte
- Continue flag for the item keyboard
  = ØØH: Restart mode
  = Nonzero: Continue mode (default)

## 3.5 Keyboard

### 3.5.1 General

The PINE supports the standard and the item keyboards. The PINE keyboards are controlled by the slave CPU (7508), which communicates with the main CPU via an serial interface.

The 7508 checks the state of all keys approximately every 30 ms. If a key entry has been made, the 7508 loads the key position code into its own key buffer and sends an interrupt signal to the main CPU.

The main CPU, on receipt of the interrupt, fetches the position code via the interrupt processing routine and loads it into its key buffer.

The data in the key buffer is taken and processed one byte at a time by the BIOS CONIN or CONST routine.

This section describes the difference between the standard and item keyboards and the method of controlling key entries.

### 3.5.2 Keyboard Functions

#### 3.5.2.1 Keys

(1) Number of keys
    Standard keyboard: 72
    Item keyboard: 58
(2) Number of switch keys
    Standard keyboard: 6 (CTRL, SHIFT x 2, CAPS, GRPH, KANA or NUM)
    Item keyboard: 4 (SHIFT, STOP, INIT, CTRL)

#### 3.5.2.2 Key scanning

The 7508 scans the keyboard every 30 ms. If the depression of a key other than switch keys is sensed, the 7508 generates the position of the key in the matrix as a make code. If a switch key is pressed, it outputs a make code when the key is pressed and outputs a break code when it is released. If two or more keys are pressed consecutively, the 7508 outputs the corresponding codes accordingly. If two or more keys are pressed simultaneously, the 7508 outputs the code of whichever key is scanned first. If the keys are switch keys, it outputs the codes of all switch keys that are pressed.

#### 3.5.2.3 Auto repeat function

All standard keyboard keys other than the switch keys, the STOP, ESC, PAUSE, and HELP keys, and function keys PF1 - PF5 are repeatable. The auto repeat function is disabled by default when an item keyboard is installed. The auto repeat on/off state, repeat start time, and repeat interval time can be changed using the BIOS CONOUT routine (see the paragraphs on the CONOUT BIOS function.

#### 3.5.2.4 Inhibiting keyboard interrupts

The PINE can be disabled or enabled for keyboard interrupts using commands. It may also be enabled only for STOP key interrupts. The steps below show how the 7508 processes keyboard interrupts when only STOP key interrupts are enabled.