

INPUT FILENAME : BUFFER.ASM  
 OUTPUT FILENAME : BUFFER.DBJ

\*\*\*\*\*  
 .PL 60  
 .PM 79  
 .TITLE BUFFERED I/O SOFTWARE VERSION 2.01 12/2/84  
 .VER 2.01 FIX-RESET BUFFER FLAG ON SPACE COMPRESSION  
 .COPYRIGHT 1984 SIMPLIMAY PRODUCTS CO.  
 \*\*\*\*\*

\*\*\*\*\*  
 BUFFER PARAMETER TABLE  
 \*\*\*\*\*

```

; BYTE DESCRIPTION
; 0,1 Data in Buffer address pointer
; 2,3 Data out Buffer address pointer
; 4 Reserve Cell number
; 5 ETX,ETB/ACK data counter
; 6 Reserve Cell used flag
; 7 Buffer Full flag
; 8 Protocol flags
; 9 Despool function flags
; 10 Data in Stack pointer
; 11 Data out Stack pointer
; 12,13 Data counter
; 14,15 Hold buffer data counter
; 16,17 Hold buffer address pointer
; 18 Hold buffer out Stack pointer
; 19 Present Compressed Space address
; 21-75 Buffer Cell Stack
;
; PROTOCOL FLAGS
;
; BIT DESCRIPTION
; 0 Protocol Active
; 1 Wait flag
; 2 Xon/Xoff Active
; 3 Etx/Ack Active
; 4 Etd/Ack Active
; 5-7 Not Used
;
; Despool Flags
;
; BIT DESCRIPTION
; 0 Space compression active
; 1 Hold buffer active
; 2 Page pause active
; 3 Next page flag
; 4 Space present flag
; 5 No buffering active
    
```

\*\*\*\*\*  
 COMMAND SUMMARY  
 \*\*\*\*\*

BAUD RATE	SERIAL A	SERIAL B
110	00H	10H
300	01H	11H
600	02H	12H
1,200	03H	13H
2,400	04H	14H
4,800	05H	15H
9,600	06H	16H
19,200	07H	17H

\*\*\*\*\*  
 COUNTER FUNCTIONS  
 \*\*\*\*\*

COUNTER PERIOD	SER A	SER B	PAR B
LSB COUNT	40H	50H	60H
MSB COUNT	41H	51H	61H
CLEAR BUFFER	42H	52H	62H
CLEAR BUFFER	43H	53H	63H
PAGE PAUSE ON	44H	54H	64H
PAGE PAUSE OFF	45H	55H	65H
IMMEDIATE PAUSE	46H	56H	66H
NEXT PAGE	47H	57H	67H
SPACE COMPRESSION ON	48H	58H	68H
SPACE COMPRESSION OFF	49H	59H	69H
BUFFERING OFF	4AH	5AH	6AH
BUFFERING ON	4BH	5BH	6BH

\*\*\*\*\*  
 DESPOOL FUNCTIONS  
 \*\*\*\*\*

DESPOOL FUNCTIONS	SER A	SER B	PAR B
BUFFER HOLD ON	40H	50H	60H
BUFFER HOLD OFF	41H	51H	61H
COPY BUFFER	42H	52H	62H
CLEAR BUFFER	43H	53H	63H
PAGE PAUSE ON	44H	54H	64H
PAGE PAUSE OFF	45H	55H	65H
IMMEDIATE PAUSE	46H	56H	66H
NEXT PAGE	47H	57H	67H
SPACE COMPRESSION ON	48H	58H	68H
SPACE COMPRESSION OFF	49H	59H	69H
BUFFERING OFF	4AH	5AH	6AH
BUFFERING ON	4BH	5BH	6BH

\*\*\*\*\*  
 PROTOCOL FUNCTIONS  
 \*\*\*\*\*

PROTOCOL FUNCTIONS	SER A	SER B
BREAK ON	80H	90H
BREAK OFF	81H	91H

```

: XON/XOFF          EQU 92H      92H
: ETX/ACK          EQU 83H      93H
: ETB/ACK          EQU 84H      94H
: ALL PROTOCOLS OFF EQU 85H      95H
-----
: SOFTWARE RESET    FFH
: *****
: START:
:
:   DEFINED PARAMETERS
:
:   S100 I/O PORTS
:
BUSST EQU 40H    ;S100 I/O PORT STATUS
BSAU EQU 41H    ;SERIAL A OUTPUT PORT
BSBU EQU 42H    ;SERIAL B OUTPUT PORT
BPAD EQU 43H    ;PARALLEL A OUTPUT PORT
BRUFST EQU 44H  ;BUFFER STATUS PORT
BINDTA EQU 45H  ;INPUT DATA PORT
BINADD EQU 46H  ;INPUT DATA ADDRESS PORT
:
:   BAUD GENERATOR / COUNTER
:
BAUD1 EQU 00H   ;BAUD RATE A COUNT REGISTER
BAUD2 EQU 01H   ;BAUD RATE B COUNT REGISTER
CTR EQU 02H    ;INTERRUPT COUNT REGISTER
CTMODE EQU 03H ;BAUD / COUNTER MODE REGISTER
:
:   PARALLEL I/O PORT 1 & 2
:
PADATA EQU 10H  ;PARALLEL A DATA REGISTER
PBDATA EQU 11H  ;PARALLEL B DATA REGISTER
PSTAT EQU 12H  ;PARALLEL STATUS REGISTER
PCMD EQU 13H   ;COMMAND MODE REGISTER
:
:   SERIAL I/O PORT A
:
SADATA EQU 20H  ;SERIAL DATA REGISTER
SACMD EQU 21H  ;SERIAL COMMAND REGISTER
:
:   SERIAL I/O PORT B
:
SBDATA EQU 30H  ;SERIAL DATA REGISTER
SBCMD EQU 31H  ;SERIAL COMMAND REGISTER
:
:   ABSOLUTE ADDRESSES
:
STACK EQU START+0F00H ;STACK POINTER
:CELL PARAMETERS
:
0000
: BEGINNING OF PROGRAM

```

```

PTRHSA EQU START+800H  ; BUFFER TO SERIAL A
PTRSBA EQU PTRHSA+80  ; BUFFER A TO HOST
PTRSBH EQU PTRSBA+80  ; BUFFER TO SERIAL B
PTRSBH+80 EQU PTRSBH+80 ; BUFFER B TO HOST
PTRPAH EQU PTRSBH+80  ; BUFFER PAR A TO HOST
PTRPHB EQU PTRPAH+80  ; BUFFER TO PARALLEL B
:
MEMMAP EQU PTRPHB+80  ; MEMORY USE BIT MAP
CMDPTR EQU CMDPTR+80  ; COMMAND VECTOR
INTMP EQU INTMP+80    ; TEMPORARY STORAGE
CTR01 EQU CTR01+80    ; COUNTER DATA
BUFST EQU BUFST+80    ; BUFFER STATUS BYTE
:
BSAD0K EQU 01H        ; HOST STATUS MASKS
BSRD0K EQU 02H
BPAD0K EQU 04H
:
PTRSAA+12 EQU PTRSAA+12 ; CHARACTER COUNT LOCATIONS
PTRSBA+12 EQU PTRSBA+12
PTRSBH+12 EQU PTRSBH+12
PTRPAH+12 EQU PTRPAH+12
PTRPHB+12 EQU PTRPHB+12
:
CONSTANTS
:
NULL EQU 0           ; NULL
ZERO EQU 0           ; ZERO
ETX EQU 3            ; ETX
ETB EQU 5            ; ETB
ACK EQU 6            ; ACK
XON EQU 11H         ; XON
XOFF EQU 13H        ; XOFF
MAXCNT EQU 0CH      ; MAXCNT
OFFSET EQU 21       ; OFFSET
MAPCNT EQU 24       ; MAPCNT
ZCNT EQU 237H       ; ZCNT
SEREN EQU 37H       ; SEREN
SERM16 EQU 4EH      ; SERM16
B110 EQU 1047       ; B110
B300 EQU 384        ; B300
B600 EQU 192        ; B600
B1200 EQU 96        ; B1200
B2400 EQU 48        ; B2400
B4800 EQU 24        ; B4800
B9600 EQU 12        ; B9600
B19200 EQU 6        ; B19200
:
DRG EQU 00H         ; BEGINNING OF PROGRAM

```



```

00D1 3E 09 LD A,09H
00D3 D3 13 OUT (PCMD),A
00D5 FB ;
;
; STATUS OUTPUT ROUTINE
00D6 2A 5C 08 ; STOUT: LD
00D9 7C LD HL,(CTRSAB)
00DB 7C LD A,H
00DB C4 93 01 DR NZ,BUFSTA
00DB 7C 0C 08 LD HL,(CTRSAB)
00E1 7C LD A,H
00E2 B5 DR NZ,BUFSTA
00E3 C4 30 01 CALL HL,(CTRSBH)
00E6 2A FC 08 LD HL,(CTRSBH)
00E9 7C LD A,H
00EA B5 DR NZ,BUFSTA
00EB C4 A5 01 CALL HL,(CTRSAB)
00EE 2A AC 08 LD HL,(CTRSAB)
00F1 7C LD A,H
00F2 B5 DR NZ,BUFSTA
00F3 C4 AD 01 CALL NZ,BUFSTA
00F6 2A 4C 09 LD HL,(CTRSAB)
00F9 7C LD A,H
00FA B5 DR NZ,BUFSTA
00FB C4 D8 01 CALL NZ,BUFSTA
00FE 2A 9C 09 LD HL,(CTRSBH)
0101 7C LD A,H
0102 B5 DR NZ,BUFSTA
0103 C4 6A 01 CALL NZ,BUFSTA
;
; BREAK SERVICE ROUTINE
0106 DB 21 IN A,(SACMD)
0108 E6 40 AND 40H
010A F3 DI
010B 3A 36 0A LD A,(BUFSTA)
010E CB BF RES 7,A
0110 CB BF JR Z,NBRKA
0112 CB FF SET Z,NBRKA
0114 32 36 0A LD (BUFSTA),A
0118 FB EI (BBUFSTA),A
0119 D3 44 ;
;
011A DB 31 IN A,(SBCHD)
011C E6 40 AND 40H
011E F3 DI
011F 3A 36 0A LD A,(BUFSTA)
0122 CB B7 RES 6,A
0124 28 02 JR Z,NBRKB
0126 CB F7 SET 6,A
0128 32 36 0A LD (BUFSTA),A
012B FB EI (BBUFSTA),A
012C D3 44 ; TELL HOST
;
012E 18 A6 JR STSOUT
;
; SER A BUFFER OUT TO SERIAL A
0130 DD 21 00 08 BUFSTA: LD 1X,PTRSHA
0134 CE 21 LD C,SACMD
0136 C6 01 LD B,1
0138 C4 F0 01 CALL NZ,SERST
013D 20 42 JR NZ,BUFFRE
0140 D3 20 CALL GETDATA
0142 CD 7F 01 OUT (SADATA),A
0145 DD CB 08 46 CALL SET_BUFFER_STATUS
0149 C4 FC 01 BIT 0,(1X+8)
014C C9 CALL NZ,EXPROT
;
; SER B BUFFER TO SERIAL B
014D DD 21 60 08 BUFSTA: LD 1X,PTRSHB
0151 0E 31 LD C,SBCHD
0153 06 02 LD B,2
0155 CD F0 01 CALL NZ,SERST
0158 20 25 JR NZ,BUFFRE
015A CD 2A 02 CALL GETDATA
015D D3 30 OUT (SBDATA),A
015F CD 7F 01 CALL SET_BUFFER_STATUS
0162 DD CB 08 46 BIT 0,(1X+8)
0166 C4 FC 01 CALL NZ,EXPROT
0169 C9 RET
;
; PAR B BUFFER TO PARALLEL B
016A DD 21 90 09 BUFSTA: LD B,4
016C DD 21 90 09 LD 1X,PTRHPB
0170 D8 12 IN A,(PSTAT)
0172 E6 02 AND 02H
0174 28 09 JR Z,BUFFRE
;
0176 CD 2A 02 CALL GETDATA
0179 D3 11 OUT (PBDATA),A
017B CD 7F 01 CALL SET_BUFFER_STATUS
017E C9 RET
;
; UPDATE BUFFER STATUS TO HOST
017F DD CB 07 46 BUFSTA: BIT 0,(1X+7)
0183 C0 RET NZ
0184 F3 DI
0185 3A 36 0A LD A,(BUFSTA)
0188 A0 AND B
0189 28 06 JR Z,DIVFRE
018B A8 XOR B
018C D3 44 OUT (BBUFSTA),A
;
; CHECK FLAG
; ON DOWN'T CLEAR
; SET STATUS BYTE
; IS IT RESET?
; YES DO NOTHING
; RESET BIT
; TELL HOST

```

```

018E 32 36 0A LD (BUFST),A
0191 FB DVERFR: EI
0192 C9 RET
;
; SER A BUFFER TO HOST
;
0193 DD 21 50 08 BUFTHA: LD IX,PTPSAH
0197 0E 21 C,SACHD
0199 DB 40 A,(BUSST)
019B E6 01 AND BSAOMK
019D C0 RET NZ
;
019E CD 2A 02 CALL GETDTA
01A1 D3 41 OUT (BSAD),A
01A3 1B 10 JR BSTPCK
;
; SER B BUFFER TO HOST
;
01A5 DD 21 F0 08 BUFTHB: LD IX,PTPSBH
01A9 0E 31 C,SBCMD
01AB DB 40 A,(BUSST)
01AD E6 02 AND BSBOMK
01AF C0 RET NZ
;
01B0 CD 2A 02 CALL GETDTA
01B3 D3 42 OUT (BSBD),A
;
; BUFFER STATUS CHECK
;
01B5 DD CB 07 46 BSTPCK: BIT 0,(1X+7)
01B9 C0 RET NZ
01BA 3E 37 LD A,SEREN
01BC ED 79 OUT (C),A
01BE DD CB 08 4E BIT 1,(1X+8)
01C2 C8 RET Z
01C3 DD CB 08 BE RES 1,(1X+8)
01C7 DD CB 08 56 RET 2,(1X+8)
01C9 C8 RET Z
01CC ED 78 IN A,(C)
01CE FE 04 CP 4
01D0 28 FA JR Z,SST
01D2 3E 11 LD A,XON
01D4 0D DEC C
01D5 ED 79 OUT (C),A
01D7 C9 RET
;
; PAR A BUFFER TO HOST
;
01D9 DD 21 40 09 BUFHPA: LD IX,PTRPAH
;
; LOAD CELL PARAMATER POINTER
;
01DC DB 40 IN A,(BUSST)
01DE E6 04 AND BPAOMK
01E0 C0 RET NZ
;
; SAVE IT
;
01E1 CD 2A 02 CALL GETDTA
01E4 D3 43 OUT (BPAO),A
01E6 DD CB 07 46 BIT 0,(1X+7)
01E8 C0 RET NZ
01EB 3E 09 LD A,9
01ED D3 13 OUT (PCMD),A
01EF C9 RET
;
; SERIAL PORT STATUS ROUTINE
;
01F0 DD CB 08 4E SERST: BIT 1,(1X+8)
01F4 C0 RET NZ
01F5 ED 78 IN A,(C)
01F7 E6 85 AND BSH
01F9 FE 85 CP BSH
01FB C9 RET
;
; ETX-ETB PROTOCOL ROUTINE
;
01FC DD CB 08 56 EXPROT: BIT 2,(1X+8)
0200 C0 RET NZ
0201 DD 34 05 INC (1X+5)
0204 3E 40 LD A,MAXCNT
0206 DD BE 05 CP (1X+5)
0209 C0 RET NZ
;
020A DD 36 05 00 LD (1X+5),0
020E DD CB 08 5E JR Z,ETBPRO
0212 28 04 LD A,ETX
0214 3E 03 LD A,ETB
0218 3E 17 ETBPRG: LD A,ETB
021A F5 SENDET: PUSH C
021B 0C FQ 01 SERST
021C CD F0 01 CALL NZ,UP3
021F 20 FB JR POP AF
0221 F1 POP AF
0222 0D DEC C
0223 ED 79 OUT (C),A
0225 DD CB 08 CE SET 1,(1X+8)
0229 C9 RET
;
; GET CHARACTER ROUTINE
;
022A DD CB 09 5E GETDTA: BIT 3,(1X+9)
022E 28 02 JR Z,OVRRNT
0230 F1 POP AF
0231 C9 RET
;
; CHECK PAGE HOLD FLAG
;
0232 DD 4E 02 DVRNT: PUSH BC
0233 DD 6E 02 LD L,(1X+2)
0236 DD 66 03 LD H,(1X+3)
0239 7E LD A,(HL)
023A 32 32 0A LD (TEMP),A
;
; LOAD CELL PARAMATER POINTER
;
023C DD 4E 02 IN A,(BUSST)
023E DD 66 03 AND BPAOMK
0239 7E RET NZ
;
; MASK IT
;
023A 32 32 0A LD (TEMP),A
;
; SEND IT
;
023A 32 32 0A LD (TEMP),A
;
; SEND IT TO HOST
;
023A 32 32 0A LD (TEMP),A
;
; CHECK BUFFER STATUS/PROTOCOL
;
023A 32 32 0A LD (TEMP),A
;
; GET NEXT CHARACTER
;
023A 32 32 0A LD (TEMP),A
;
; SEND IT
;
023A 32 32 0A LD (TEMP),A
;
; TURN PORT INTERRUPT ON
;
023A 32 32 0A LD (TEMP),A
;
; TEST FLAG FOR WAIT
;
023A 32 32 0A LD (TEMP),A
;
; NOT READY RETURN
;
023A 32 32 0A LD (TEMP),A
;
; GET PORT STATUS
;
023A 32 32 0A LD (TEMP),A
;
; STRIP USED BITS
;
023A 32 32 0A LD (TEMP),A
;
; ALL THERE?
;
023A 32 32 0A LD (TEMP),A
;
; INCREMENT COUNT
;
023A 32 32 0A LD (TEMP),A
;
; LOAD MAX COUNT
;
023A 32 32 0A LD (TEMP),A
;
; COMPARE WITH COUNT
;
023A 32 32 0A LD (TEMP),A
;
; NOT MAX RETURN
;
023A 32 32 0A LD (TEMP),A
;
; ZERO COUNT
;
023A 32 32 0A LD (TEMP),A
;
; IS PROTOCOL ETX
;
023A 32 32 0A LD (TEMP),A
;
; PUT ETX IN A
;
023A 32 32 0A LD (TEMP),A
;
; PUT ETB IN A
;
023A 32 32 0A LD (TEMP),A
;
; PRINT TO CHD PORT
;
023A 32 32 0A LD (TEMP),A
;
; SET STATUS
;
023A 32 32 0A LD (TEMP),A
;
; WAIT TILL READY
;
023A 32 32 0A LD (TEMP),A
;
; POINT TO PORT
;
023A 32 32 0A LD (TEMP),A
;
; SEND IT
;
023A 32 32 0A LD (TEMP),A
;
; SET WAIT BIT
;
023A 32 32 0A LD (TEMP),A
;
; WAIT FOR NEW PAGE
;
023A 32 32 0A LD (TEMP),A
;
; SAVE PORTS
;
023A 32 32 0A LD (TEMP),A
;
; GET DATA POINTER
;
023A 32 32 0A LD (TEMP),A
;
; GET CHARACTER
;
023A 32 32 0A LD (TEMP),A
;
; TEMPORARY STORE IT
;
023A 32 32 0A LD (TEMP),A
;

```

```

0230 DD CB 09 6E BIT ; IF NO BUFFERING
0241 28 07 JR Z,OVERNR ; RESET BUFFER FLAG
0243 DD CB 07 96 RES 0,(IX+7) ; GET SPACE BLOCKING BYTE
0247 C3 E4 02 JP GDWCNT ; NO SPACE BLOCKING
024A DD CB 09 46 DWRNR: BIT Z,NOBP
024E 28 17 JR ;
0250 CB 7F BIT ; IS IT A MULTI-SPACE
0252 28 13 JR Z,NOBP ; NOT A SPACE DO NOTHING
0254 FE 81 CP BIR ; IS THIS THE LAST?
0256 28 0A JR Z,SPIT ; NO SEND MORE
0258 30 DEC A ; ONE LESS
0259 77 LD (HL),A ; SAVE COUNT IN BUFFER
025A 3E 20 LD A, ; PUT SPACE IN TEMP
025C 32 32 0A LD (TEMP),A ; FALL DONE
025F C3 F2 02 JP GETDON ;
0262 3E 20 SPIT: ; PUT SPACE IN TEMP
0264 32 32 0A LD (TEMP),A ;
0267 FE 0C FEED ; CHECK FOR FORM FEED
0269 20 0A CP NZ,OVERPG ; PAGE HLD ON?
026B DD CB 09 56 BIT 2,(IX+9) ; SET PAGE FLAG
026F 28 04 JR Z,OVERPG ;
0271 DD CB 09 DE SET 3,(IX+9) ;
0275 E5 11 01 FC ; SAVE REGISTER
0276 7C LD DE,-3FFH ; CHECK FOR END OF CELL
0279 E6 03 AND A,H ; STRIP OF HIGH ADDRESS
027A 67 LD H,A ;
027C 027D ADC HL,DE ; RESTORE DATA
027E E1 POP POF ; INC ONE
0280 23 INC HL ; DON'T NEED NEW CELL
0281 20 58 JR NZ,OLDCEL ;
0283 DD E5 ; CALC STACK ADDRESS
0285 E1 LD HL ;
0286 11 15 00 LD DE,OFFSET ;
0289 19 ADD E,(IX+11) ; GET STACK NUMBER
028A DD SE 08 LD HL,DE ; OFFSET ADDRESS
028E 19 ADD A,(HL) ; GET CELL NUMBER
028F ED 52 SBC HL,DE ; GET BACK TO BASE
0291 DD 4E 04 LD C,(IX+4) ; GET RESERVED CELL NUMBER
0294 89 CP ; IF SAME RESET RESERVE FLAG
0295 20 0A JR NZ,OVERRES ;
0297 DD CB 09 4E ; IF HOLD ON DON'T RESET
0298 20 24 JR NZ,HOLDON ;
029D DD CB 06 86 RES 0,(IX+6) ; RESET RESERVED CELL FLAG
02A1 DD CB 09 4E DWRRES: BIT 1,(IX+9) ; IF HOLD DON'T RESET
02A5 20 1A JR NZ,HOLDON ;

```

```

02A7 E5 ; DWRHLD: PUSH HL ; SAVE NUMBERS
02A8 D5 DE ; MOVE CELL NUMBER TO E
02A9 5F E,A ; LOAD MEMORY MAP ADDRESS
02AA 21 E0 09 LD HL,MENMAP ; ADD CELL NUMBER
02AD 19 LD HL,DE ; RESET BIT
02AE 36 00 LD (HL),0 ;
02B0 DD CB 07 46 ; IF FULL SKIP
02B2 28 09 JR BIT 0,(IX+7)
02B4 F3 JR Z,OVRCCK ;
02B6 DD E5 DI ;
02B7 DD E5 PUSH DI ; GIVE PUTDIA A VECTOR
02B9 FD E1 POP POF ;
02BB CD CE 06 CALL CELLC ; GET NEW ADDRESS
02BD FB EI ; GET NUMBERS
02BF D1 DWRCK: POP DE ;
02C0 E1 HL ;
02C1 1C ; HOLDON: INC E
02C2 3E 38 LD E,A,MAPCNT+2
02C4 8B CP ; IF AT END WRAP
02C5 20 02 E NZ,OVRCCL ; START OVER
02C7 1E 00 LD E,0 ; SAVE IT
02C9 DD 73 08 DWRCEL: LD HL,DE ; ADD NEW OFFSET
02CB 19 LD A,(IX+11),E ; GET CELL NUMBER
02CD 7E LD A,(HL) ;
02CE 21 00 10 ; MULTPLY: LD HL,1000H
02D1 11 00 04 ; LOAD CELL SIZE
02D4 C8 3F SRL A ; SHIFT TO CARRY
02D6 30 01 JR NC,NOADD ; IF NO CARRY NO ADD
02D8 19 ADD HL,DE ; ADD
02DA CB 22 LD D ; MULT *2
02DB B7 D ;
02DD 20 F6 JR A,ZERO? ;
02DE DD 75 02 ; OLDCEL: LD (IX+2),L
02E1 DD 74 03 LD (IX+3),H ; SAVE NEW ADDRESS
02E4 AF ; CLEAR A
02E5 F3 ;
02E6 DD BE 0C DI ;
02E9 20 03 0D ; IS LSB ZERO
02EB DD 35 0D JR NZ,OVERDEC ; NO DON'T BORROW
02ED DD 35 0C DEC (IX+13) ; DEC MSB
02F1 FB DWRDEC: DEC (IX+12) ; DEC COUNT
02F2 CA ;
02F3 31 32 0A ; GETDON: POP BC ; RESTORE PORTS
02F6 C9 LD A,(TEMP) ; GET CHARACTER
; ;
; ; INTERRUPT SERVICE ROUTINES
; ; ;
; ; ; HOST COMMAND/DATA

```



```

03A7 3A 35 0A      LD      A,(CTR2)
03A8 D3 02          RUIT
03A9 C3 29 07      JP
03AC          :
03AF 3E B0          STCTR: LD      A,OR04
03B0 D3 03          DOUT  JP      (CTMODE),A
03B3 C3 29 07      :
03B6 FD 21 00 08  DSPSA: LD      IY,PTRHSA
03B8 18 0A          JR
03BC          :
03C0 FD 21 00 08  DSPSB: LD      IY,PTRSB
03C2 18 04          JR
03C5          :
03C6 78           DSPOL: LD      A,B
03C7 EA 0F          AND    0FH
03C9 FE 00          CP
03CB 20 25          JR
03CD FD CB 09 CE    SET
03D0          :
03D1 FD 7E 0C      DSPHLD: LD      A,(IY+12)
03D2 FD 77 0E      LD      (IY+14),A
03D3 FD 7E 0D      LD      A,(IY+13)
03D4 FD 77 0F      LD      (IY+15),A
03D5 FD 7E 02      LD      A,(IY+2)
03D6 FD 77 10      LD      (IY+16),A
03D7 FD 7E 03      LD      A,(IY+3)
03D8 FD 77 11      LD      (IY+17),A
03D9 FD 7E 08      LD      A,(IY+11)
03DA FD 77 12      LD      (IY+18),A
03DB C3 29 07      JP
03DE          :
03E1 FE 01          NNT1: CP
03E2 20 0A          JR
03E3 FD CB 09 BE    RES
03E4 CD CC 04          CALL
03E5 C3 29 07      JP
03E8          :
03EA FE 04          NNT2: CP
03EB 20 07          JR
03EC FD CB 09 D6    SET
03ED C3 29 07      JP
03F0          :
03F2 FE 01          NNT3: CP
03F3 20 08          JR
03F4 FD CB 09 96    RES
03F5 FD CB 09 9E    RES
03F6 FD CB 09 9E    RES
03F7 C3 29 07      JP
0400          :
0401 FE 05          NNT4: CP
0402 20 0B          JR
0403 FD CB 09 96    RES
0404 FD CB 09 9E    RES
0405 C3 29 07      JP
0408          :
0409 FE 07          NNT5: CP
040A 20 07          JR
040B FD CB 09 C6    SET
040C C3 29 07      JP
040D          :
040E FE 08          NNT6: CP
040F 20 07          JR
0410 FD CB 09 C6    SET
0411 C3 29 07      JP
0412          :
0413 FE 09          NNT7: CP
0414 20 07          JR
0415 FD CB 09 86    RES
0416 C3 29 07      JP
0417          :
0418 FE 0A          NNT8: CP
0419 20 07          JR
041A FD CB 09 86    RES
041B C3 29 07      JP
041C          :
041D FE 0B          NNT9: CP
041E 20 07          JR
041F FD CB 09 C6    SET
0420 C3 29 07      JP
0421          :
0422 FE 0C          NNT10: CP
0423 20 07          JR
0424 FD CB 09 C6    SET
0425 C3 29 07      JP
0426          :
0427 FE 0D          NNT11: CP
0428 20 07          JR
0429 FD CB 09 C6    SET
042A C3 29 07      JP
042B          :
042C FE 0E          NNT12: CP
042D 20 07          JR
042E FD CB 09 C6    SET
042F C3 29 07      JP
0430          :
0431 FE 0F          NNT13: CP
0432 20 07          JR
0433 FD CB 09 86    RES
0434 C3 29 07      JP
0435          :
0436 FE 02          NNT14: CP
0437 C2 6C 04          JR
0438 FD CB 09 4E    BIT
0439 CA 29 07      JP
043A          :
043B FE 02          NNT15: CP
043C C2 6C 04          JR
043D FD CB 09 4E    BIT
043E CA 29 07      JP
043F          :
0440 FE 03          NNT16: CP
0441 C2 9C 04          JR
0442 FD 7E 0A          LD
0443 FD 77 08          LD
0444 FD 7E 12          LD
0445 FD 77 0E          LD
0446 FD 77 0C          LD
0447 FD 7E 0F          LD
0448 FD 7E 0F          LD
0449 FD 77 0D          LD
044A FD 77 0D          LD
044B FD 77 0D          LD
044C FD 77 0D          LD
044D FD 77 0D          LD
044E FD 77 0D          LD
044F FD 77 0D          LD
0450 FD 77 0D          LD
0451 FD 77 10          LD
0452 FD 77 02          LD
0453 FD 77 02          LD
0454 FD 7E 11          LD
0455 FD 7E 03          LD
0456 FD 77 03          LD
0457 FD 77 03          LD
0458 FD 77 03          LD
0459 FD 77 03          LD
045A FD 77 03          LD
045B FD 77 03          LD
045C FD 77 03          LD
045D FD 77 03          LD
045E FD 77 03          LD
045F FD 77 03          LD
0460 FD 77 03          LD
0461 FD 77 03          LD
0462 FD 77 03          LD
0463 FD 77 03          LD
0464 FD 77 03          LD
0465 FD 77 03          LD
0466 FD 77 03          LD
0467 FD 77 03          LD
0468 FD 77 03          LD
0469 FD 77 03          LD
0470 FD 77 03          LD
0471 FD 77 03          LD
0472 FD 77 03          LD
0473 FD 77 03          LD
0474 FD 77 03          LD
0475 FD 77 03          LD
0476 FD 77 03          LD
0477 FD 77 03          LD
0478 FD 77 03          LD
0479 FD 77 03          LD
0480 FD 77 03          LD
0481 FD 77 03          LD
0482 FD 77 03          LD
0483 FD 77 03          LD
0484 FD 77 03          LD
0485 FD 77 03          LD
0486 FD 77 03          LD
0487 FD 77 03          LD
0488 FD 77 03          LD
0489 FD 77 03          LD
0490 FD 77 03          LD
0491 FD 77 03          LD
0492 FD 77 03          LD
0493 FD 77 03          LD
0494 FD 77 03          LD
0495 FD 77 03          LD
0496 FD 77 03          LD
0497 FD 77 03          LD
0498 FD 77 03          LD
0499 FD 77 03          LD
0500 FD 77 03          LD

```



```

0499 C3 29 07 JP VECT1
049C FE 0A CP I0
049E C2 AB 04 NZ,NXT10
04A1 FD CB 09 EE SET S,(IY+9)
04A5 C3 29 07 JP VECT1
04A8 FE 08 CP I1
04AA C2 B4 04 NZ,NXT11
04AD FD CB 09 AE RES S,(IY+9)
04B1 C3 29 07 JP VECT1
04B4 FE 06 CP I2
04B6 C2 29 07 NZ,VECT1
04B9 FD CB 09 DE SET S,(IY+9)
04BD C3 29 07 JP VECT1
MEMMAP BIT RESET ROUTINE
04C0 FD E5 BITRES: PUSH IY
04C2 E1 POP HL
04C3 11 15 00 LD DE,OFFSET
04C6 19 ADD HL,DE
04C7 FD 5E 08 LD E,(IY+11)
04CA FD 4E 04 LD C,(IY+4)
04CD FD 7E 04 LD A,(IY+10)
04D0 BB CP Z
04D1 C8 RET HL,DE
04D2 19 ADD A,(HL)
04D3 7E LD C
04D4 B9 CP NZ,QVRBIT
04D5 20 04 JR 0,(IY+6)
04D7 FD CB 06 86 QVRBIT: PUSH DE
04D8 D5 RES DE
04DC E5 LD E,A
04DD 5F LD HL,MEMMAP
04DE 21 E0 09 LD HL,DE
04E1 19 LD (HL),0
04E2 36 00 LD POP
04E4 E1 LD HL,DE
04E5 D1 POP DE
04E6 1C INC E
04E7 3E 38 LD A,MOPCNT+2
04E9 BB CP E
04EA 20 E1 JR NZ,NXTBIT
04EC 1E 00 LD E,0
04EE 18 DD JR NXTBIT
PROTOCOL FUNCTIONS
SERIAL A
04F0 FD 21 00 08 PROSA: LD LD
04F4 21 50 08 LD LD IY,PTRNSA
HL,PTNSAH
;LOAD PARAMETER POINTER

```

```

04F7 0E 21 LD C,SACMD
04F9 18 09 JR PROTST
SERIAL B
04A9 FD 21 A0 08 PROSB: LD LD
04AF 21 F0 08 LD HL,PTRSBH
0502 0E 31 LD C,SECMD
0504 78 PROTST: LD LD
0505 E6 0F AND A,B
0507 FE 00 CP 0
0509 20 07 JR NZ,PNXT1
050B 3E 37 LD A,SREN
050D ED 79 LD (C),A
050F C3 29 07 JP VECT1
0512 FE 01 PNXT1: CP 1
0514 20 07 JR NZ,PNXT2
0516 3E 3F LD A,SREN+8
0518 ED 79 LD (C),A
051A C3 29 07 JP VECT1
051D FE 02 PNXT2: CP 2
051F 20 04 JR NZ,PNXT3
0521 3E 05 LD A,S
0523 FD 77 08 FINT1: LD LD
0526 E5 LD A,11H
0527 FD E1 POP HL
0529 FD 77 08 LD (IY+8),A
052C C3 29 07 JP VECT1
052F FE 03 PNXT3: CP 3
0531 20 04 JR NZ,PNXT4
0533 3E 09 LD A,9
0535 18 EC JR FINT1
0537 FE 04 PNXT4: CP 4
0539 20 04 JR NZ,PNXT5
053B 3E 11 LD A,11H
053D 18 E4 JR FINT1
053F FE 05 PNXT5: CP 5
0541 C2 29 07 JR NZ,VECT1
0544 AF XOR A
0545 18 DC JR FINT1
INTERUPT HOST TO SERIAL A
0547 FD 21 00 08 INHSA: LD LD
054B 0E 01 LD C,1
054D 18 0E JR INTRBS
;LOAD PARAMETER POINTER
;LOAD BUFFER MASK
;GET CHARACTER

```

```

;
; INTERRUPT HOST TO SERIAL B
054F FD 21 A0 08 INTSH: LD IY, PTRSHB ;LOAD PARAMETER POINTER
0553 0E 02 LD C, 2 ;LOAD BUFFER MASK
0555 18 06 JR INTSH ;GET CHARACTER
;
; INTERRUPT HOST TO PARALLEL B
0557 FD 21 90 09 INTHP: LD IY, PTRHPB ;LOAD PARAMETER TABLE
0558 0E 04 LD C, 4 ;LOAD BUFFER MASK
;
; INTERRUPT HOST TO ALL I/O PORTS
055D 3A 36 0A INTBUS: LD A, (BUFST) ;TELL HOST YOUR BUSY
0560 B1 00 OR C ;SET BIT
0561 D3 44 (BRUFST), A ;TELL HOST
0563 32 36 0A LD (BUFST), A ;SAVE IT
;
; GET DATA
0566 DB 45 IN A, (INDTA) ;GET DATA
0568 FD CB 07 46 BIT O, (Y+7) ;IF BUFFER FULL
056C C2 29 07 JP NZ, VECT1 ;STORE IT IN BUFFER
056F CD 23 06 CALL PUDTA ;STORE IT IN BUFFER
0572 C3 29 07 JP VECT1
;
; INTERRUPT SERIAL A TO HOST
0575 F5 AF INTSAH: PUSH AF ;
0576 C5 BC PUSH BC ;
0577 E5 DE PUSH DE ;
0578 E5 HL PUSH HL ;
0579 0E 21 C, 9ACMD ;LOAD PORT NUMBER
057B FD 21 50 08 LD IY, PTRSAH ;LOAD PARAMETER POINTER
057E CD ED 05 CALL SERIN ;GET CHARACTER
0582 D4 23 06 NC, PUDTA ;STORE DATA IN BUFFER
0585 CD A1 05 SBUF ;DO PROTOCOLS IF BUFFER FULL
0588 C3 3E 07 JP VECT4 ;CHECK OTHER VECTORS
;
; INTERRUPT SERIAL B TO HOST
058B F5 AF INTSBH: PUSH AF ;
058C C5 BC PUSH BC ;
058D D5 DE PUSH DE ;
058E E5 HL PUSH HL ;
058F 0E 31 C, SB CMD ;LOAD PORT NUMBER
0591 FD 21 F0 08 LD IY, PTRSBH ;LOAD PARAMETER POINTER
0594 CD ED 05 CALL SERIN ;GET CHARACTER
0598 D4 23 06 NC, PUDTA ;STORE DATA IN BUFFER
059B CD A1 05 SBUF ;DO PROTOCOLS IF BUFFER FULL
059E C3 37 07 JP VECT3 ;CHECK OTHER VECTORS
;
; SERIAL BUFFERS STATUS CHECK
05A1 FD CB 07 46 SBUF: BIT 0, (Y+7) ;IF BUFFERS FULL
05A5 28 1B JR Z, DVDRSR ; DO NOTHING
;
; SERIAL INPUT ROUTINE
05A7 3E 36 LD A, SEREN-1 ;
05A9 ED 79 OUT (C), A ;
05AB FD CB 08 56 D3, (Y+8) ;XON/XOFF ON?
05AD C8 RET Z ;NO SEND
05AF 08 RET ;CHECK STATUS
05B2 E6 81 AND A, (C) ;
05B4 FE 81 CP B1H ;
05B6 28 FB JR Z, *-6 ;
05B8 00 DEC C ;
05B9 3E 13 XOFF ;POINT AT DATA PORT
05BB ED 79 LD IY, XOFF ;
05BD FD CB 08 CE SET 1, (Y+8) ;SEND XOFF
05C1 C9 RET ;SET WAIT BIT
;
; DOWDRSR: LD A, SEREN ;RESET DSR
05C2 3E 37 OUT (C), A ;SEND IT
05C4 ED 79 RET ;
05C6 C9 RET ;
;
; INTERRUPT PARALLEL A TO HOST
05C7 F5 AF INTPAH: PUSH AF ;
05C9 C5 BC PUSH BC ;
05CB D5 DE PUSH DE ;
05CD HL ;
05CE IY, PTRPAH ;LOAD PARAMETER POINTER
05D0 FD 21 40 09 PAM: LD A, B ;DISABLE PORT INTERRUPT
05D2 3E 08 LD (PCMD), A ;
05D4 D3 13 OUT 0, (Y+7) ;IF BUFFERS FULL
05D6 FD CB 07 46 BIT 0, (Y+7) ;NOT OK RETURN
05D8 DB 10 IN A, (PADATA) ;GET CHARACTER
05DA 05DC CALL PUDTA ;CHECK IF BUFFER FULL
05DC FD CB 07 46 BIT O, (Y+7) ;
05DE C2 30 07 JP A, 9 ;ENABLE PORT INTERRUPT
05E0 3E 09 LD A, 9 ;
05E2 D3 13 NZ, VECT2 ;
05E4 C3 30 07 JP VECT2 ;
;
; SERIAL INPUT ROUTINE
05ED 3E 35 LD A, SEREN-2 ;SET DSR FOR BUSY
05EF ED 79 OUT (C), A ;TELL PERIPHERAL
05F1 0D DEC C ;POINT AT PORT
05F2 ED 78 INC A, (C) ;GET DATA
05F4 0C INC C ;POINT AT CMD
05F6 FD CB 07 46 BIT 0, (Y+7) ;IF BUFFER FULL WASTE DATA
05F8 37 SCF ;SET CARRY
05FA C0 RET NZ ;
05FB B7 DR ;
05FC C9 RET A ;CLEAR CARRY
;
; INTERRUPT PROTOCOL SERVICE ROUTINE
05FD FD CB 88 56 PROTO: BIT 2, (Y-72) ;IS PROTO XON/XOFF?
0601 28 14 JR Z, CKYACK ;

```

```

0603 FE 11 CP XON ;IS IT XON
0605 20 06 JR NZ,DXOFF ;RESET WAIT FLAG
0607 FD CB B8 BE RES 1,(1Y-72) ;CLEAR CARRY
0608 B7 OR A ;SET WAIT FLAG
060C C9 RET ;CLEAR CARRY

060D FE 13 ;IS IT XOFF
060F 20 10 JR NZ,NDOACT ;SET WAIT FLAG
0611 FD CB B8 CE SET 1,(1Y-72) ;CLEAR CARRY
0615 B7 OR A ;SET WAIT FLAG
0616 C9 RET ;CLEAR CARRY

0617 FE 06 ;IS IT ACK
0619 20 06 JR NZ,NDOACT ;NO ACTION NEEDED
061B FD CB B8 BE RES 1,(1Y-72) ;RESET WAIT FLAG
061F B7 OR A ;CLEAR CARRY
0620 C9 RET ;SET CARRY

0621 37 ;NOACT: SET
0622 C9 RET ;SET CARRY

;
;
; PUT DATA
0623 DD CB 08 46 ;PUTDTA: BIT 0,(1Y+8)
0627 28 04 JR Z,NOPRO ;CHECK FOR PROTOCOL
0629 FD FD 05 CALL PROTO ;DO PROTOCOL
062C D0 RET NC ;NO CHARACTER

062D C5 ;NOPRO: PUSH BC
062E FD 6E 00 LD L,(1Y+0) ;SAVE PORT
0631 FD 66 01 LD H,(1Y+1) ;LOAD DELT POINTERS

0634 32 33 0A LD (INTMP),A ;TEMP STORE
0637 CB 7F JR 7,A ;CHECK FOR PARITY BIT
0639 28 04 JR Z,SPCK ;CHECK FOR SPACK
063B FD CB 09 B6 RES 0,(1Y+9) ;CANCEL SPACK COMPRESSION

063F FD CB 09 46 ;SPCK: BIT 0,(1Y+9)
0643 CA BC 05 JP Z,NOPACK ;CHECK FOR SPACK COMPRESSION
0646 28 0F CP ;IS CHARACTER A SPACE?
0648 28 0F JR Z,ITISSP ;YES - PAK IT
064E 28 3C BIT 4,(1Y+9) ;SPACE PRESENT
0650 FD CB 09 A6 ADJNT: RES 4,(1Y+9) ;RESET SPACK PRESENT FLAG
0654 CD B6 06 CALL ADVANC ;INC COUNTER
0657 18 33 JR NOPACK

0659 3A 34 0A ;ITISSP: LD A,(BUFS1)
065C 49 XOR C ;CLEAR BIT FOR THIS BUFFER
065D 32 36 0A LD (BUFS1),A ;SAVE IT
0660 D3 44 OUT (BBUFS1),A ;TELL HOST

0662 FD CB 09 66 ;BIT 4,(1Y+9)
0666 20 0E JR NZ,MORSP ;SPACE COUNT?

```

```

0669 36 B1 LD (HL),B1 ;START SPACE COUNT
066A FD CB 09 E6 SET 4,(1Y+9) ;SET SPACE PRESENT BIT
066E 7D 13 LD (1Y+19),L ;SAVE SPACE ADDRESS
0671 FD 74 14 LD (1Y+20),H ;SAVE SPACE ADDRESS
0674 18 1D JR ALDND ;COUNT FULL?

0676 FD 6E 13 ;MORSP: LD L,(1Y+19)
0679 FD 66 14 LD H,(1Y+20) ;GET SPACE ADDRESS
067C 7E LD A,(HL) ;GET SPACE COUNT
067D 3C INC A ;ADD ONE TO SPACE COUNT
067E FE FE CP ;COUNT FULL?
0680 20 07 JR NZ,DVRC ;RESET SPACE BPRESENT FLAG
0682 FD CB 09 A6 RES 4,(1Y+9) ;RESET SPACE BPRESENT FLAG
0686 CD B6 06 CALL ADVANC ;ADVANCE CHARACTER COUNT

0689 77 ;DVRC: LD (HL),A ;SAVE COUNT
068A C1 POP BC ;RESTORE PORTS
068B C9 RET ;RESTORE PORTS

068C 3A 33 0A ;NOPACK: LD A,(INTMP)
068F 77 LD (HL),A ;GET CHARACTER
0690 CD B6 06 CALL ADVANC ;SAVE CHARACTER
0693 28 06 BIT 5,(1Y+9) ;INC COUNT
0697 28 06 JR Z,DVNB ;IF NO BUFFERING
0699 FD CB 07 C6 SET 0,(1Y+7) ;SET BUFFER FULL
069E C9 POP BC ;RESTORE BC
RET

069F CD BE 06 ;DVNB: CALL NXTADD ;GET NXTADD
06A2 FD 75 00 LD (1Y+0),L ;SAVE ADDRESS
06A5 FD 74 01 LD (1Y+1),H ;SAVE ADDRESS
06A8 C1 POP BC ;RESTORE PORTS
06A9 FD CB 09 4E JR 1,(1Y+9) ;RESTORE PORTS
06AD CB 0E RET (1Y+14) ;IF HOLD ON INC HOLD COUNT
06AE CD 0E INC (1Y+14)
06B1 CD 0E RET NZ ;RESTORE PORTS
06B2 FD 34 0F INC (1Y+15) ;RESTORE BC
RET

;
;
; ADVANCE CHARACTER COUNT
06B6 FD 34 0C ;ADVANC: INC (1Y+12) ;INC LSB
06B9 CD 0D RET NZ ;INC MSB
06BA FD 34 0D INC (1Y+13) ;INC MSB
06BD C9 RET

;
;
; CALC NEXT BUFFER CHARACTER ADDRESS
06BE E5 ;NXTADD: PUSH HL ;SAVE REGISTER
06BF 11 01 FC LD DE,-3FFH ;CHECK FOR END OF CELL
06C2 7C LD A,H ;STRIP OF HIGH ADDRESS
06C3 E6 03 AND 3 ;STRIP OF HIGH ADDRESS
06C5 67 LD LD H,A ;STRIP OF HIGH ADDRESS
06C6 ED 5A ADC HL,DE ;STRIP OF HIGH ADDRESS

```

```

06C8 E1 POP HL ;RESTORE DATA
06C9 23 INC TMC ;INC TIME
06CA CC CE 06 CALL Z,CELLCK ;NEED NEW CELL
06CD C9 RET

;
;
; ROUTINE LOOKS FOR FREE CELLS AND CALC'S ADDRESS
06CE 21 E6 09 ;CELLCK: LD HL,MEMMAP+6
06D1 01 36 00 LD LD ;LOAD MEMORY MAP POINTER
06D4 AF B1 XOR A ;LOAD POINTER
06D5 ED B1 CPTR A ;CLEAR A
06D7 CA EE 06 JP Z,ONECEL ;LOOK FOR FREE CELL
; FOUND FREE CELL

;
; NO CELLS FREE
06DA FD CB 06 46 BIT 0,(1Y+6) ;NO CELLS FREE
06DB 3E FF SET A,OFFH ;SET FLAG FOR RETURN
06DE FD CB 07 C6 SET 0,(1Y+7) ;SET BUFFER FULL FLAG
06E0 C0 RET NZ ;BUFFER FULL+RESERVE IN USE
06E4 ;
;
06E5 FD CB 06 C6 SET 0,(1Y+6) ;SET RESERVE CELL FLAG
06E9 FD 7E 04 LD A,(1Y+4) ;GET RESERVE CELL NUMBER
06EC 18 06 JIR DVRSUB

;
;
; ADJUST POINTER
06EE 2B DEC HL ;ADJUST POINTER
06EF CB C6 SET 0,(HL) ;SET CELL AS BUSY

;
;
; FIGURE CELL ADDRESS
06F1 3E 3B LD A,MAPCNT+5 ;FIGURE CELL ADDRESS
06F3 91 SUB C ;
06F4 4F LD D,C+1 ;CELL NUMBER IN C
06F5 FD E5 PUSH IV ;CALC STACK ADDRESS
06F7 E1 POP HL ;
06F8 11 15 00 LD DE,OFFSET ;ADD OFFSET TO BASE
06FB 19 LD HL,DE ;
06FC FD 5E 04 LD E,(1Y+10) ;ADD STACK POINTER
06FF 1C INC E ;ADVANCE POINTER
0700 3E 3B LD A,MAPCNT+2 ;WRAP IF END OF STACK
0702 BB CP JR E,0 ;NO DON'T WRAP
0703 20 02 LD E,0 ;WRAP
0705 1E 00 LD D,E ;
0707 19 LD HL,DE ;PUT CELL NUMBER IN STACK
0708 FD 71 LD HL,C ;SAVE POINTER
0709 FD 73 04 LD LD ;
;
; GET READY TO MULTIPLY
070C 21 00 10 ;DELADD: LD HL,1000H ;GET READY TO MULTIPLY
070F 11 00 04 LD DE,400H ;LOAD CELL SIZE
0712 79 LD A,C ;PUT CELL NUMBER IN A
0713 CB 3F SRL A ;SHIFT TO CARRY
0715 30 01 LD A,NC,NADD ;IF NO CARRY NO ADD
0717 19 LD HL,DE ;ADD
0718 CB 22 DADD D ;MULT X2
071A B7 DR A ;A ZERO?
071B 87 JR NZ,CELA ;
071D FD 75 00 LD LD ;SAVE ADDRESS
0720 FD 74 01 LD LD ;
0723 AF XOR XDR ;SET FLAG AS FOUND

```

```

0724 FD CB 07 86 ;RESTORE BUFFER FULL FLAG
072B C9 RET

;
; INTERRUPT VECTOR CHECK
0729 DB 12 VECT1: IN A,(PSTAT) ;CHECK STAUS ON PAR A
072B E6 08 AND BH ;
072D C2 CB 05 JP NZ,PAH ;
0730 DB 31 IN A,(SBCMD) ;CHECK STATUS ON SER B
0732 E6 02 AND 2 ;
0734 C2 8F 05 JP NZ,SBH ;
0737 DB 21 IN A,(SACMD) ;CHECK STATUS ON SER A
0739 E6 02 AND 2 ;
073B C2 79 05 JP NZ,SAH ;
073E E1 POP HL ;RESTORE REGISTERS
073F D1 POP DE ;
0740 C1 POP BC ;
0741 F1 POP AF ;
0742 FB EI ;
0743 C9 RET

;
;
; BAUD RATE TABLE
0744 17 04 ;TABLE: DEFW B110 ;110 BAUD
0746 80 01 DEFW B300 ;300 BAUD
0748 C0 00 DEFW B600 ;600 BAUD
074A 60 00 DEFW B1200 ;1200 BAUD
074C 30 00 DEFW B2400 ;2400 BAUD
074E 18 00 DEFW B4800 ;4800 BAUD
0750 0C 00 DEFW B9600 ;9600 BAUD
0752 06 00 DEFW B19200 ;19200 BAUD

;
;
; DEFAULT I/O PORT PARAMETER TABLE
0754 00 2B ;PTRIBL: DEFW 2800H ;HOST TO SERIAL A
0756 00 2B DEFW 2800H ;
0758 00 DEFW 0 ;
;
; SERIAL A TO HOST
0759 00 2C DEFW 2C00H ;SERIAL A TO HOST
075B 00 2C DEFW 2C00H ;
075D 01 DEFW 1 ;
;
; HOST TO SERIAL B
075E 00 30 DEFW 3000H ;HOST TO SERIAL B
0760 00 30 DEFW 3000H ;
0762 02 DEFW 2 ;
;
; SERIAL B TO HOST
0763 00 34 DEFW 3400H ;SERIAL B TO HOST
0765 00 34 DEFW 3400H ;
0767 03 DEFW 3 ;
;
; PARALLEL A TO HOST
0768 00 3B DEFW 3B00H ;PARALLEL A TO HOST
076A 00 3B DEFW 3B00H ;
076C 04 DEFW 4 ;

```

```

076D 00 3C ;
076F 00 3C DEFM 3C00H ;
0771 05 DEFM 3C00H ;
0772 06 07 08 09 ;
0776 0A 0B DEFB 6,7,8,9,10,11 ;
;
;
;
;
0778 0C 00 ; IOPAR:
077A 80 01 DEFM B9600 ;BAUD A
077C DEFM B300 ;BAUD B
END

```

\*\*\*\*\* S Y M B O L I C R E F E R E N C E T A B L E \*\*\*\*\*

ACK	= 0006	ADCNT	= 0650	ADVANC	= 06B6	ALLDON	= 0693
B110	= 0417	B1200	= 0660	B19200	= 0006	B2400	= 0030
B300	= 0180	B4800	= 0018	B600	= 00C0	B9600	= 00C0
BAUD1	= 0000	BAUD2	= 0001	BAUIDL	= 0357	BAUDSA	= 0351
BAUDSR	= 0355	BUFEST	= 0044	BIANDL	= 0004	BIANDTA	= 0045
BITRES	= 04C0	BPAD	= 0043	BRADL	= 0004	BSAD	= 0041
BSADNK	= 0001	BSRD	= 0042	BSRDK	= 0002	BSIPCK	= 01B5
BTABLE	= 0744	BUFPRE	= 017F	BURPBR	= 010B	BUFSIT	= 0436
BTABLE E	= 0193	BUFTHB	= 0165	BUFTPB	= 016A	BUFTSA	= 0130
BUFTHA	= 014D	BUSST	= 0040	CELA	= 0713	CELADD	= 070C
BUFTSB	= 064E	CKXACK	= 0617	CKXOFF	= 06D0	CMD	= 0700
CELLCK	= 0315	CMDPTR	= 0430	CMDEEC	= 0319	CTMODE	= 0003
CMDBYT	= 0002	CTRD1	= 0434	CTRD2	= 0435	CTRHFB	= 099C
CTR	= 080C	CTRH5A	= 08AC	CTRLD	= 03B6	CTRLD1	= 0394
CTRHSA	= 094C	CTRS9A	= 085C	CTRSBH	= 08FC	CTRSBT	= 036B
DSPHLD	= 03D1	DSPDOL	= 03C6	DSPPR	= 03C2	DSPSA	= 03B6
DSPSR	= 03BC	ETB	= 0017	ETPRD	= 021B	ETX	= 0003
EXPRT	= 01FC	FFED	= 000C	FINI	= 02A3	GDWONT	= 0003
GETDON	= 02F2	GETDTR	= 022A	GDCTR	= 0342	HOLDON	= 02E4
INTRBUS	= 055D	INTH8	= 0537	INTSHA	= 0547	INTHSB	= 02C1
INTPAH	= 05C7	INTSAH	= 0579	INTSBH	= 05BB	INTTMP	= 054F
IOPAR	= 0778	ITISSP	= 0659	LDDP1	= 0044	LPEND	= 004C
MAPCNT	= 0036	MAXCNT	= 0040	MEMMAP	= 09E0	MMLDOP	= 0048
MORSP	= 0676	MULTPY	= 02CE	NADD	= 071B	NBRKA	= 0114
NBRKB	= 012B	NDACT	= 0621	NOADD	= 02D9	NBRACK	= 068C
NDPRD	= 062D	NDSP	= 0267	NULL	= 0000	NXT1	= 03F2
NXT10	= 04AB	NXT11	= 04B4	NXT2	= 0400	NXT3	= 0408
NXT10	= 0448	NXT11	= 0425	NXT6	= 0450	NXT7	= 0458
NXT4	= 041A	NXT9	= 0425	NXTA	= 02D4	NXTADD	= 06BE
NXT8	= 044C	NXT9	= 049C	DLJCEL	= 02DE	UNJCEL	= 06EE
NXTBIT	= 04CD	DFFSET	= 0015	DLDCEL	= 02D4	UNJCEL	= 06EE
DVBD1T	= 04DB	DVRCCK	= 02BF	DVRCCL	= 02C9	DVRLD	= 0689
DVBD1C	= 02EE	DVRC1R	= 05C2	DVRFRE	= 0191	DVRRLD	= 02A7
DVRNB	= 069F	DVRNBR	= 024A	DVRRDT	= 0232	DVRRG	= 0275
DVRRAP	= 0707	DVRES	= 02A1	DVRSUB	= 06F4	PADATA	= 0010
FAH	= 05CB	PBDATA	= 0011	PCMD	= 0013	PADATA	= 0010
FNXT2	= 051D	PBDATA	= 0011	PCMD	= 0013	PADATA	= 0010
PROSA	= 04F0	PROSR	= 052F	PNTY4	= 0537	PNTY5	= 053F
PSIAT	= 0012	PROSR	= 04FB	PROTD	= 05FD	PROTST	= 0504
PTRH8B	= 0912	PTRH8B	= 0990	PTRSHA	= 08D0	PTRHSB	= 05B0
PUTPAH	= 0940	PTRSHH	= 0850	PTRSBH	= 08F0	PTRTBL	= 0754
PUTDTA	= 0623	PWJINT	= 003B	SACMD	= 0021	SADATA	= 0020
SAH	= 0579	SBCMD	= 0031	SBDATA	= 0030	SBH	= 058F
SRUFF	= 05A1	SENDET	= 001B	SEREN	= 0037	SERIN	= 05ED
SERMF6	= 004E	SERST	= 01F0	SETPR	= 037D	SPCK	= 063F
SP1T	= 0262	SET	= 01CC	STACK	= 0FA0	START	= 0000
STCTR	= 03AF	STSOUL	= 00D6	TEMP	= 0A32	UP3	= 021C
STCTR	= 0729	VECT1	= 0730	VECT3	= 0737	VECT4	= 073E
XOFF	= 0013	XCN	= 0011	ZCNT	= 0237	ZEND	= 0000

0000 ASSEMBLY ERRORS