I/O Address Space

The iPDS CPU (an MCS-80/85) can address 256 (0 - 0FFH) I/O ports. These ports are assigned as follows:

Port Address		Port Function		
00H -	0FH	Reserved		
10H -	1FH	EMV/PROM Programmer Interface		
20H -	2FH	Reserved		
30H -	3FH	Reserved		
40H -	4FH	Multimodule Chip Select J1		
50H -	5FH	Multimodule Chip Select J2		
60H -	6FH	Multimodule Chip Select J3		
70H -	7FH	Multimodule Chip Select J4		
	80H	8253 Counter 0 (Baud Clock)		
	81H	8253 Counter 1 (Disk Motor-On Timer)		
	82H	8253 Counter 2 (Disk Index Timer)		
	83H	8253 Mode Select		
84H -	8FH	Reserved		
	90H	8251A Serial I/O Data		
	91H	8251A Serial I/O Command Status		
92H -	9FH	Reserved		
A0H -	A1H	Interrupt Controller 8259A Read/Write		
A2H -	AFH	Reserved		
	B0H	FDC 8272 Main Status Register		
	B1H	FDC 8272 Data Register Read/Write		
B2H -	BFH	Reserved		
	C0H	CRT/KYBD 8255A		Port A Data Read/Write
	C1H	CRT/KYBD 8255A		Port B Bit 0 through 7
			PB0	RQFD-B(A) (Disk Request)*
			PB1	RQM0-B(A) (MMIO J1/J2 Request)*
			PB2	RQM1-B(A) (MMIO J3/J4 Request)*
			PB3	Disk Ready
			PB4	MPST0 (MMIO Present J1)
			PB5	MPST1 (MMIO Present J2)
			PB6	MPST2 (MMIO Present J3)
			PB7	MPST3 (MMIO Present J4)
	C2H	CRT/KYBD 8255A		Port C Bit 0 through 7
			PC0	RQFD-A(B) (Disk Request)
			PC1	RQM0-A(B) (MMIO J1/J2 Request)
			PC2	RQM1-A(B) (MMIO J3/J4 Request)
			PC3	KBINT, (Kybd Interrupt)
			PC4	STBA/B (CRT/KYBD Data Strobe)
			PC5	IBF, (Input Buffer Full)
			PC6	ACKA/B (CRT/KYBD Data Acknowledge)
			PC7	OBFA/B (Output Buffer Full)
	C3H	8255A Control Byte		
C4H -	CFH	Reserved		
	D0H	FDC Terminal Count		
D1H -	DFH	Reserved		
E0H -	E2H	Line Printer 8255A		
	E3H	PROM Programming/Emulator Power Control		
E4H -	EFH	Reserved		
	F0H	Boot ROM Disable		
F1H -	FFH	Reserved		

*Signals from the other processor.

In the following descriptions, it is assumed that the user is familiar with I/O programming techniques and with the I/O chips used. Refer to the current *Intel Component Data Catalog* for further information. Refer to Appendix A for the pin assignments and I/O signals.

CRT and Keyboard I/O

This section describes the keyboard characters available and their interpretation by the system. It also describes the graphics characters.

Most of the keyboard characters are sent to ISIS by the CRT/Keyboard controller as the ASCII code corresponding to the character. The following exceptions to this rule may affect user-written programs that read characters input at the keyboard:

1. Triple key operations are undefined. For example, pressing the FUNCTION key, the SHIFT key, and some other key simultaneously does not have a defined effect.

2. The up arrow key generates the code 1EH instead of CTRL-↑. The left arrow key generates the code 1FH instead of CTRL←.

3. The FUNCTION key only has affect with upper case alpha characters and the digits 0-9. The result of these function characters is that the ASCII code for the character with the most significant bit set is sent to the processor (except for FUNCT-R, FUNCT-S, and FUNCT-T). Lower case function characters are converted to the corresponding upper case function character and sent as an upper case function character.

4. The following function characters are processed directly by the CRT/Keyboard controller and are not sent to the processor:

FUNCT-T	Typewriter/Non-typewriter mode switch
FUNCT-S	CRT scrolling speed switch
FUNCT-HOME	Processor keyboard/screen assignment switch
FUNCT-↑	Increase size of lower half of split screen display
FUNCT- ↓	Decrease size of lower half of split screen display
FUNCT-R	Interrupt processor currently assigned

Cursor Addressing and Graphics Mode. The cursor location on the CRT screen can be programmed. To control the cursor location from a program, output an ESC (1BH) followed by another ASCII character as defined below. Use the CO system call two times to output the two bytes (1BH, <ASCII code>). Since the cursor location is relative to a full screen; in a split screen with dual processors, the cursor may not appear on the physical screen.

ESC, A Move the cursor up one line.

ESC, B Move the cursor down one line.

ESC, C Move the cursor to the right one character.

ESC, D Move the cursor to the left one character. If the cursor is at the first character of a line, it is wrapped around to the last character of the previous line.

ESC, E Home cursor and clear the screen.

ESC, H Home the cursor.

ESC, J Erase from the current location of the cursor to the end of the screen.

ESC, K Erase the line containing the cursor from the cursor to the end of the line.

The following sequence moves the cursor to a specified address on the screen. The address is given as an x,y coordinate with an offset of 20H. Thus, to move the cursor to the first character (character 0) on the first line (line 0), the address would be 20H,20H.

ESC, Y, <x>, <y> Move the cursor to the address specified in the third and fourth bytes output. Use the CO system call to output the bytes. Add 20H to the absolute values for <x>, and <y>, since the CRT/Keyboard controller subtracts 20H from the value it receives. This offset is used to be compatible with other products.

In addition, the CRT/Keyboard controller is capable of generating a set of graphics characters that can be displayed on the screen by a user-written program.

The steps to follow in writing a program to output graphics characters to the screen are:

1. Enter graphics mode by outputting the sequence ESC, G. Use the CO system call two times to output the two bytes (1BH, 47H).
2. Move the cursor to the desired location by outputting one of the cursor location control sequences to the CRT screen. Use the CO system call to output the cursor control sequences described previously.
3. Use the CO system call to output the code for the desired graphics symbol or ASCII character. The ASCII codes and the codes for graphics symbols are given in Appendix C.
4. Repeat steps 2 and 3 until the entire graphics display is completed.
5. Exit from graphics mode by outputting the sequence ESC, N. Use the CO system call two times to output the two bytes (1BH, 4EH).

The escape sequences to enter and exit graphics mode are:

ESC, G Enter graphics mode. In graphics mode, any control characters (00H-1EH except 02H, alternate escape, or 1EH, escape) that are output will be displayed as the graphics symbol corresponding to the code. The codes and their corresponding graphic symbols are given in Appendix C. Other characters (20H-7EH) will be displayed as the corresponding ASCII characters.

ESC, N Exit from graphics mode.

Serial I/O

ISIS provides an I/O driver and operating system commands for the serial I/O port. The following information is provided for those who wish to write a customized I/O driver.

The 8253 Programmable Interval Timer is used to provide software control for the baud rate on the serial I/O port.

Intended Baud Rate to be Used	Programmable Baud Rate Generator Nominal Output Frequency (KHz)		
19200	19.2	---	---
9600	9.6	153.6	---
4800	4.8	76.8	307.2
2400	2.4	38.4	153.6
1200	---	19.2	76.8
600	---	9.6	38.4
300	---	4.8	19.2
150	---	2.4	9.6
110	---	1.76	7.04

The 8253 Programmable Timer generates the frequencies shown in the preceding table corresponding to the desired baud rate. The frequency required for a given baud rate depends on the 8251 mode instruction (1X, 16X, or 64X). The maximum allowable frequency deviation is ± 1%.

The I/O address assignment for this baud rate generator, the 8253, is:

80H: Load Counter 0 with value to generate frequency corresponding to desired baud rate on an output instruction, Read Counter 0 on an input instruction

83H: Counter 0 Mode Select

The input frequency to the counter is 1.53846 MHz ± 0.1%. This frequency is the value to be divided by the 1X, 16X, or 64X frequency in the preceding chart to generate the value to load into the counter.

The serial I/O port consists of an 8251A USART and RS-232 receivers and drivers. It provides full duplex asynchronous communication from 110 to 19200 baud using 7 bits plus a parity bit.

The port may be jumpered to use the internal 8253 Programmable Timer or an external timer as a clock. The clock can provide a signal either 16 times or 64 times the actual baud rate. See the Installation Instructions in Appendix A for setting the jumper.

The I/O addresses for the 8251 USART are:

90H: Data I/O

91H: Command/Status

Printer I/O

The printer port for Processor A uses an 8255 Programmable Parallel Interface. The 8085 I/O address assignment is:

8085 Port Address	Function
E2H:	Write Data to 8255 Port A (PA0 - PA7)
E0H:	Read 8255 Port C (PC0-PC3) Write 8255 Port C (PC4-PC7)
E1H:	Write Control Byte

The assignments for the bits in Port A and Port C on the 8255 are shown in the following chart:

Port No.	Function	Mode
PA0	DATA0	Output
PA1	DATA1	Output
PA2	DATA2	Output
PA3	DATA3	Output
PA4	DATA4	Output
PA5	DATA5	Output
PA6	DATA6	Output
PA7	DATA7	Output
PC0	SELECT	Input
PC1	BUSY	Input
PC2	$\overline{\text{ACK}}$	Input
PC3	$\overline{\text{FAULT}}$	Input
PC4	$\overline{\text{STB}}$	Output
PC5	$\overline{\text{PRIME}}$	Output
PC6	N/C	
PC7	N/C	

The 8255 control bytes are the Mode Select Byte and the Bit Set/Reset Byte. The Mode Select Byte is output at 8085 port E1H. A value of 87H (1000 0111) selects the following mode assignment for the 8255:

8255 Port A (PA0 - PA7) Output
8255 Port C (PC0 - PC3) Input
8255 Port C (PC4 - PC7) Output
8255 Port B (PB0 - PB7) Output *
* Used for Plug-in Module Adapter Power On/Off Control

The Bit Set/Reset control byte is output at the 8085 port E0H and provides single bit set/reset control for PC4-PC7 of Port C on the 8255. The following chart gives the value that is output at the 8085 port E0H to set and reset bits PC4-PC7 of Port C.

8255 Port	Set	Reset
PC4	09H (00001001B)	08H (00001000B)
PC5	0BH (00001011B)	0AH (00001010B)
PC6	0DH (00001101B)	0CH (00001100B)
PC7	0FH (00001111B)	0EH (00001110B)

Multimodule I/O

Up to four 8-bit iSBX Multimodule™ boards are supported with some restrictions. Sixteen bit Multimodules are not supported. Only non-DMA mode I/O is supported. Only limited power consumption boards are supported. See the power supply specification in Appendix A. See the *Intel iSBX® Bus Specification*, order no. 142686, for further information on the multimodule bus interface.

Most of the information required for using Multimodules is contained in the Hardware Reference Manual for each Multimodule. Some of the additional precautions which should be followed when using Multimodules with the iPDS system are as follows:

- Multimodule interrupts are returned to the INTR line of the CPU (both base and optional processor). These interrupts are maskable.

- Multimodule interrupts are sent to both processors in a dual processor system. Mask off the other processors interrupts. The base processor should enable only those interrupts ISIS does not allow. (See the ATTACH call).

- Data cannot be transferred between a Multimodule and a processor unless the proper semaphores are first set up.

- The ATTACH and DETACH commands (chapter 5) must be uset to communicate with Multimodules in the iPDS system.

Communication between the Multimodule Adapter board and wither the base or optional processor board is controlled by an 8255 Programmable Peripheral Interface (PPI) chip on each processor board. The following I/O port address list covers the ports needed to use Multimodules in the iPDS system.

Port Address	Port Function	Comments
40H - 47H	Multimodule Chip Select J1	Functions selected by A processor for MMIO at J1
48 - 4FH	Multimodule Chip Select J1	Functions selected by B processor for MMIO at J1
50H - 57H	Multimodule Chip Select J2	Functions selected by A processor for MMIO at J2
58H - 5FH	Multimodule Chip Select J2	Functions selected by B processor for MMIO at J2
60H - 67H	Multimodule Chip Select J3	Functions selected by A processor for MMIO at J3
68H - 6FH	Multimodule Chip Select J3	Functions selected by B processor for MMIO at J3
70H - 77H	Multimodule Chip Select J4	Functions selected by A processor for MMIO at J4
78H - 7FH	Multimodule Chip Select J4	Functions selected by B processor for MMIO at J4
A0H	Interrupt Controller R/W	Used to set up the 8259A Interrupt Controller for the Control Register. See Table 8-3 for pin assignments.
A1H	Interrupt Controller R/W	Used to set up the 8259A Interrupt Controller for the Data Register. See Table 8-ff for pin assignments.
C1H	PPI Port B, bits 0 through 7	
	PB0 - RQFD	Disk semaphore (Semaphore available=1)
	PB1 - RQM0	MMIO J1/J2 semaphore (Semaphore available =1)
	PB3 - Disk Ready	
	PB4 - MPST0	Multimodule present at J1
	PB5 - MPST1	Multimodule present at J2
	PB6 - MPST2	Multimodule present at J3
	PB7 - MPST3	Multimodule present at J4
C2H	PPI Port c, bits 0 through 2	
	PC0 - RQFD	Set disk request bit and turn disk motor on (Motor ON=0)
	PC1 - RQM0	Enable Multimodules at J1/J2 (MMIO Enable =0)
	PC2 - RQM1	Enable Multimodules at J3/J4 (MMIO Enable=0)

C3H PPI Control Byte, bits 0 through 2

bit 0 - Request the MMIO semaphore (Request semaphore=1)

bit 1 - Request Multimodules J1/J2 (Request MMIO=1)

bit 2 - Request Multimodules J3/J4 (Request MMIO=1)

Table 8-3 shows the pin numbers of the interrupt lines from the MMIO to the 8259A interrupt controller chips.

Table 8-3 Interrupt Line Pin Numbers

MMIO Pin Number	U-5 and U-6 8259A Pin Number
J1 Pin 12	IR0-0 Pin 18
J1 Pin 14	IR1-0 Pin 19
J2 Pin 14	IR2-0 Pin 20
J2 Pin 12	IR3-1 Pin 21
J3 Pin 14	IR0-1 Pin 22
J3 Pin 12	IR1-1 Pin 23
J4 Pin 14	IR2-1 Pin 24
J4 Pin 12	IR3-1 Pin 25