contents of this byte is (length/512). The contents of this byte is used by the CRC (cyclic-redundancy-check) routine to determine how much ROM to check.

- Location 3 contains the beginning of an initialization routine that is reached by a 'Long' call during the power-on sequence. For cartridges that are 'IPL-able' (BASIC or assembler program) this routine should set the INT hex 18 vector to point to their entry points. Other types of cartridges (BASIC or whatever) should merely 'return' to the caller. Setting the INT hex 18 vector will enable transfer of control to the cartridge program by the IPL routine.
- This location 6 should be 00.
- CRC bytes: The last two locations of the address space used by the cartridge must be blank. CRC characters will be placed in these bytes when the cartridge is built. See the routine at label "CRC Check", in the BIOS listing for the CRC algorithm.

B. The following conventions will be followed for cartridges that wish to be recognized by DOS 2.1 as containing code associated with DOS command words:

Location	Contents
0	055H
1	0AAH
2	Length
3-5	Jump to Initialize
6	Command Name Length (Offset Y-Offset Z)
Z	First Character in Command Name
Y	Last Character in Command Name
W	Word Pointing to Routine that is Jumped to if "Name" is Typed
X	Next Command Name Length or "00" if No More Command Names
Last 2 Addresses	CRC Bytes

DOS Conventions

- Locations 0 and 1 contain the word hex 55AA. This is used as a test for the presence of the cartridge during the configuration- determination portion of the power-on routines.
- Location 2 contains a length indicator representing the entire address space taken by the ROM on the cartridge. The algorithm for determining the contents of this byte is (length/512). The contents of this byte is used by the CRC routine to determine how much ROM to check.
- Location 3 contains a 'jump' to the initialization code for this ROM. (May just be a 'Far Return')
- Starting at location 6 may be a sequence of command name pointers consisting of 1: Count of length name, 2: Name in ASCII, and 3: Word

containing offset within this segment to the code that is entered when this name is called. There can be as many names as desired, providing that a hex 00 is placed in the count field following the last name pointer. If a cartridge has a routine called 'TEST' at location hex 0FB5 (offset from start of segment that the cartridge is in) that needs to be executed when 'test' is entered as a DOS command the entry at location 6 would be hex 04,54,45,53,54,B5,0F.

- CRC bytes: The last two locations of the address space used by the cartridge must be blank. CRC characters will be placed in these bytes when the cartridge is built. See the routine at label "CRC Check", in the BIOS listing for the CRC algorithm.

C. The following conventions will be followed for cartridges that wish to be recognized by "Cartridge BASIC" as containing interpretable-BASIC Code:

- The cartridge-chip selects must address hex D0000 since the BASIC cartridge addresses hex E0000. When "Cartridge BASIC" is activated, it will check for a second cartridge program at hex D0000. If the second cartridge is present and formatted properly, then the BASIC code is loaded into RAM and run.
- The format for this interpretable-BASIC code must be as follows:

Location	Contents
0	055H
1	0AAH
2	Length
3	0CBH
4	0AAH
5	055H
6	0
7	0FFH if unprotected Basic program or 0FEH if protected Basic program
8	Start of interpretable Basic code
n	0FFH Padding to next 2048 byte boundary
Last 2 Addresses	CRC Bytes

Cartridge Format

1. Locations 0 and 1 contain the word hex 55AA. This is used as a test for the presence of the cartridge during the configuration-determination portion of the power-on routines.
2. Location 2 contains a length indicator representing the entire address space taken by the ROM on the cartridge. The algorithm for determining the contents of this byte is (length/512). The contents of this byte is used by the CRC routine to determine how much ROM to check.
3. Location 3 must be hex 0CB for a 'far return' instruction.

4. Locations 4 and 5 contain the word hex AA55. This is used as a test for the presence of the second cartridge by "Cartridge Basic".
5. Location 6 must be a 0 to follow the DOS conventions.
6. Location 7 can be either hex FF to indicate an unprotected BASIC program, or hex FE to indicate a protected program.
7. Location 8 must be the start of the BASIC program. It must be interpretable Basic and not compiled. Also, at the end of the program PAD to the next 2048 byte boundary with hex 0FF.
8. CRC bytes: The last two locations of the address space used by the cartridge must be blank. CRC characters will be placed in these bytes when the cartridge is built. See the routine at label "CRC Check", in the BIOS listing for the CRC algorithm.

ROM Module

The ROM modules used are 250 ns devices. Typical modules are the Mostek MK37000 and MK38000, the TMM 23256, the SY23128, and other compatible devices.

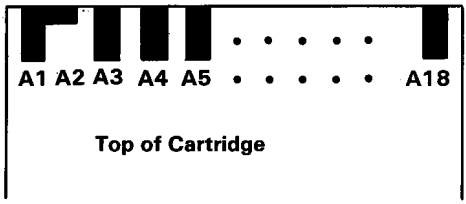
ROM Chip Select	Hex Address Space	Typical Use
CS0	X	Not Used
CS1	X	Not Used
CS2	D0000-D7FFF	Optional Cartridge ROM #2
CS3	D8000-DFFFF	Optional Cartridge ROM #1
CS4	E0000-E7FFF	Standard Cartridge ROM #2
CS5	E8000-EFFFF	Standard Cartridge ROM #1
CS6	F0000-F7FFF	System Board ROM #2
CS7	F8000-FFFFF	System Board ROM #1

ROM Chip Select Table

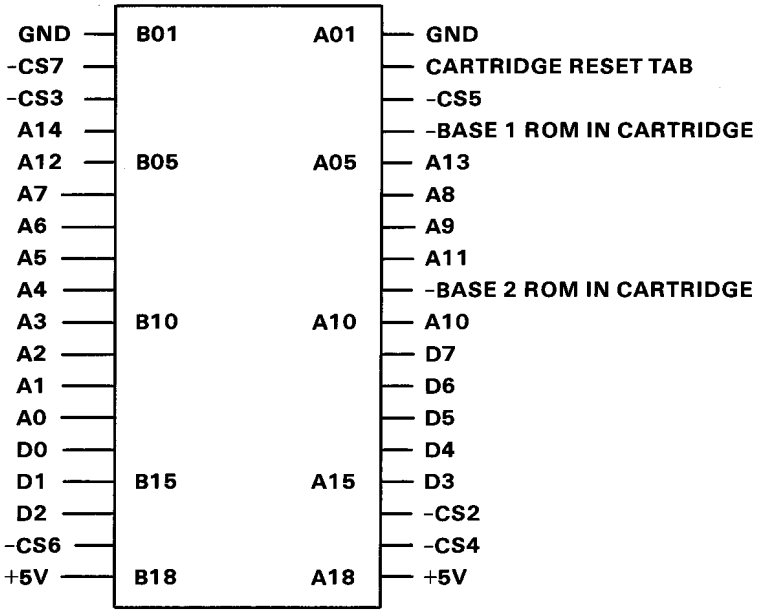
Signal	I/O	Description
A0 - A14	0	Processor Address lines A0 - A14
D0 - D7	I	Processor Data lines

-CS2 THRU -CS7	0	These chip-select lines are used to select ROM modules at different addresses. The addresses for each chip-select are shown in the ROM-chip select-table. -CS6 and -CS7 are used on the system board for BIOS, Power-On-Self-Test (POST) and cassette-basic ROMs. In order to use these chip selects on a cartridge, -BASE 1 ROM IN CARTRIDGE or -BASE 2 ROM IN CARTRIDGE must be pulled 'low'
-BASE 1 ROM IN CARTRIDGE	I	This line when pulled 'low' instructs the system board to de-gate the ROM module from hex F8000 - FFFFF on the system board. This ROM module can then be replaced by a ROM module on the cartridge by using -CS7.
-BASE 2 ROM IN CARTRIDGE	I	This line when pulled 'low' instructs the system board to de-gate the ROM module from hex F0000 - F7FFF on the system board. This ROM module can then be replaced by a ROM module on the cartridge by using -CS6.

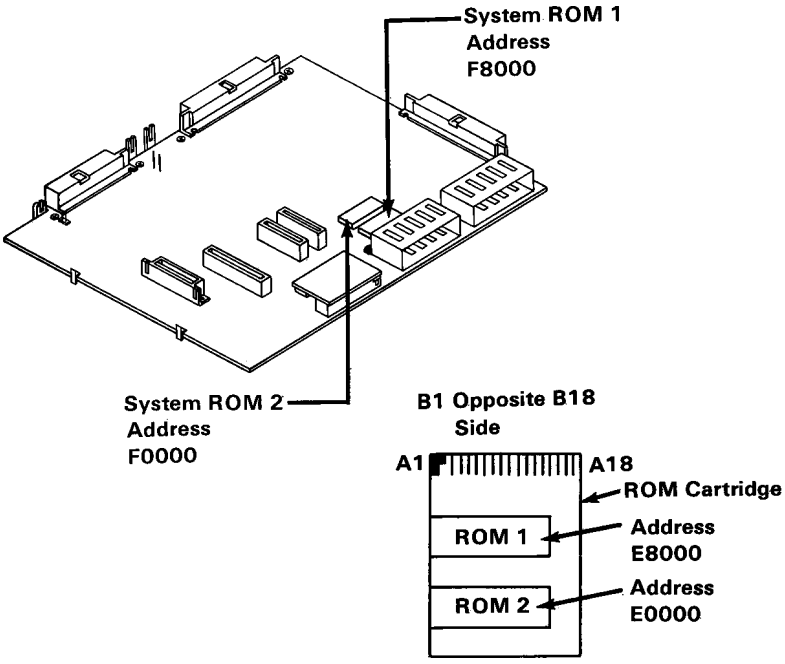
Cartridge Reset Tab	I	This input when 'low' causes a 'reset' to the system. The system will remain 'reset' until this line is brought back 'high'. This tab is usually wired with an L shaped land pattern to the GND at A02 which provides a momentary 'reset' when a cartridge is inserted or removed.



Momentary Reset Land



Connector Specification



Cartridge ROM Locations

Games Interface

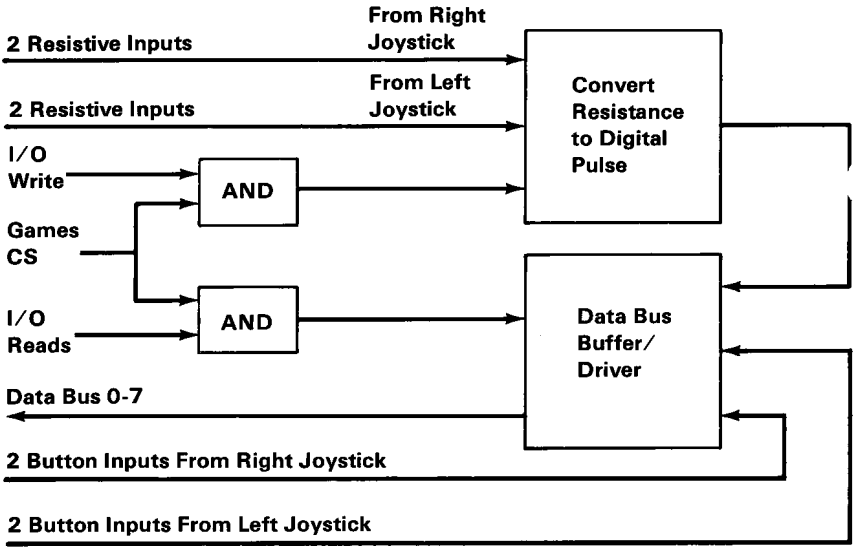
Interface Description

The Game Interface has two connectors located at the rear of the System unit for four paddles (two per connector) or two joysticks. Each connector has four input lines: two digital inputs and two resistive inputs. All the inputs are 'read' with one 'IN' from address hex 201. The interface, plus system software, converts the present resistive value to a relative paddle or joystick-position. On receipt of an output signal, four timing circuits are started. By determining the time required for the circuit to time out (a function of the resistance), the paddle or joystick position can be determined.

The four digital inputs each have a 1K ohm resistor to pull the voltage up to +5V. With no drive on these inputs, a 1 is read. For a 0 reading, the inputs must be pulled to ground.

The four resistive inputs are converted to a digital pulse with a duration proportional to the resistive load, according to the following equation:

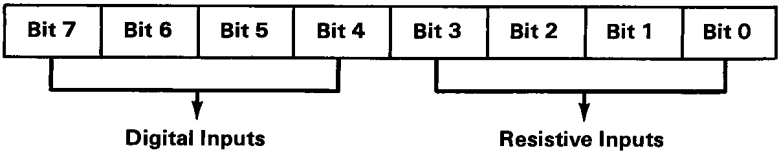
```
Time = 24.2 μs + 0.011 (r) μs
Where r is the resistance in ohms
```



Games Interface Block Diagram

Any program application must first begin the conversion by an 'OUT' to address hex 201. An 'IN' from address hex 201 will show the digital pulse go 'high' and remain 'high' for the duration according to the resistance value. All four bits (Bit 3 through Bit 0) function in the same manner. Each bits digital pulse goes high simultaneously and resets independently according to the input resistance value.

Input from Address Hex 201



Input From Address Hex 201

Joysticks typically have one or two buttons and two variable resistances each. The variable resistances are mechanically linked to have a range from 0 to 100k ohms. One variable resistance indicates the X coordinate and the other variable resistance indicates the Y coordinate. The joysticks are attached to give the following input data:

Joystick B		Joystick A		Joystick B		Joystick A	
Button #2	Button #1	Button #2	Button #1	Coord. Y	Coord. X	Coord. Y	Coord. X
Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0

Joystick Input Data

The game paddles have one button each and one variable resistance each. The variable resistance is mechanically linked to have a range from 0 to 100k ohms. The paddles are attached to give the following input data.

Buttons				Coordinates			
Paddle D	Paddle C	Paddle B	Paddle A	Paddle D	Paddle C	Paddle B	Paddle A
Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0

Paddle Input Data

Pushbuttons

The pushbutton inputs are 'read' by an 'IN' from address hex 201. These values are seen on data bits 7 through 4. These buttons default to an 'open' state and are 'read' as 1. When a button is pressed, it is 'read' as 0.

Note: Software should be aware that these buttons are not debounced in hardware.

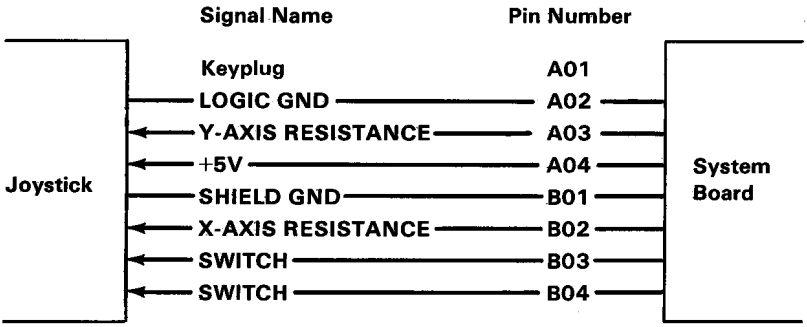
Joystick Positions

The joystick position is indicated by a potentiometer for each coordinate. Each potentiometer has a range from 0 to 100k ohms that varies the time constant for each of the four one-shots. As this time constant is set at different values, the output of the one-shot will be of varying durations.

All four one-shots are fired simultaneously by an 'OUT' to address hex 201. All four one-shot outputs

will go 'true' after the fire pulse and will remain 'high' for varying times depending on where each potentiometer is set.

These four one-shot outputs are 'read' by an 'IN' from address hex 201 and are seen on data bits 3 through 0.



Connector Specification

Notes:

Serial Port (RS232)

The PC*jr* serial port is fully programmable and supports asynchronous communications only. It will add and remove start bits, stop bits, and parity bits. A programmable baud-rate generator allows operation from 50 baud to 4800 baud. Five, six, seven or eight bit characters with 1, 1-1/2, or 2 stop bits are supported. A fully-prioritized interrupt-system controls transmit, receive, line status and data-set interrupts. Diagnostic capabilities provide loopback functions of transmit/receive and input/output signals.

The nucleus of the adapter is a 8250A LSI chip or functional equivalent. Features in addition to those previously listed are:

- Full double-buffering eliminates the need for precise synchronization
- Independent receiver clock input
- Modem control functions: clear to send (CTS), request to send (RTS), data set ready (DSR), data terminal ready (DTR)
- Even, odd, or no-parity-bit generation and detection
- False start bit detection
- Complete status reporting capabilities
- Line-break generation and detection
- Break, parity, overrun, and framing error simulation
- Full prioritized interrupt system controls

All communications protocol is a function of the system ROM and must be loaded before the adapter is operational. All pacing of the interface and control-signal status must be handled by the system software. It should be noted that Asynchronous (Async) receive operations cannot overlap diskette operation since all but the Diskette Interrupt are masked 'off' during diskette operations. If Async receive

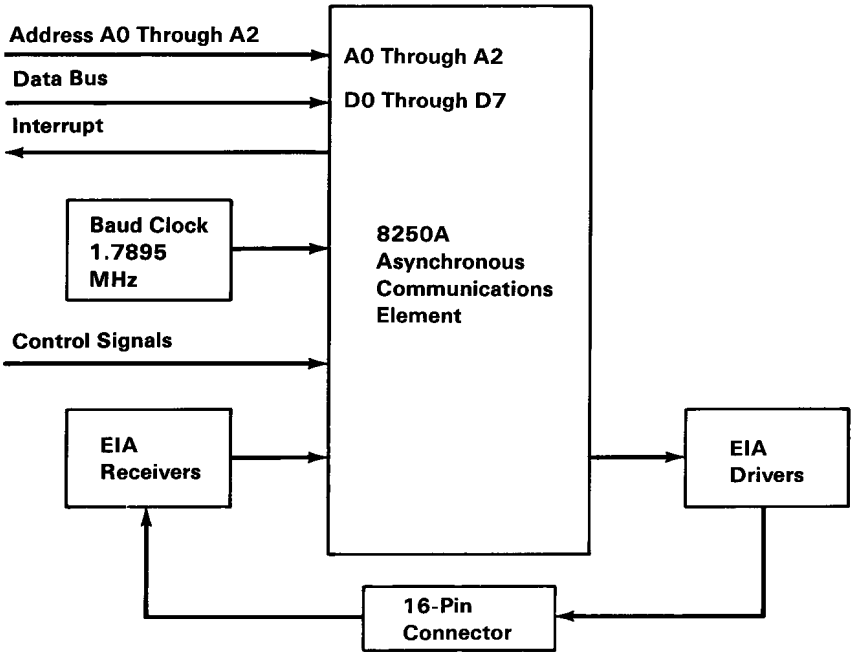
operations are going to be overlapped with keyboard receive operations, the Async Receiver rate cannot exceed 1200 baud. This is due to the processor deserialization of the keyboard. See IBM PC*jr* Cordless Keyboard in this section for more information.

Programming Note: Due to the read/write cycle-time of the 8250A, it is recommended that back-to-back I/O operations to the 8250A be avoided. A good Programming Technique would be to insert a short 'jump' between every consecutive 8250 I/O instruction. This action will flush the queue and provide 15 clock periods between I/O operations.

> **Note:** This note only applies to programmers using the 8250A directly. It is STRONGLY suggested that the user not communicate directly with the physical hardware, but use the system BIOS instead.

> **Note:** It is important to note that when the IBM PC*jr* has the Internal Modem installed it is logically COM1 and the RS232 serial port is logically COM2 in BIOS, DOS, and BASIC. Without the Internal Modem installed the RS232 serial port is logically addressed as COM1 in BIOS, DOS ,and BASIC even though its address is still hex 2F8 using Interrupt level 3.

The following figure is a Serial Port Block Diagram:



Serial Port Block Diagram

Modes of Operation

The different modes of operation are selected by programming the 8250A asynchronous communications element. This is done by selecting the I/O address (hex 2F8 to 2FF) and 'writing' data out to the card. Address bits A0, A1, and A2 select the different registers that define the modes of operation. Also, the divisor-latch access-bit (bit 7) of the line-control register is used to select certain registers.

I/O Decode (in Hex)	Register Selected	DLAB State
2F8	TX Buffer	DLAB=0 (Write)
2F8	RX Buffer	DLAB=0 (Read)
2F8	Divisor Latch LSB	DLAB=1
2F9	Divisor Latch MSB	DLAB=1
2F9	Interrupt Enable Register	DLAB=0
2FA	Interrupt Identification Registers	(Don't Care)
2FB	Line Control Register	(Don't Care)
2FC	Modem Control Register	(Don't Care)
2FD	Line Status Register	(Don't Care)
2FE	Modem Status Register	(Don't Care)
2FF	Scratch Register	(Don't Care)

I/O Decodes

Address Range hex 2F8 - 2FF

Note: The state of the divisor-latch access-bit (DLAB), which is the most-significant bit of the line-control register, affects the selection of certain 8250A registers. The DLAB must be set 'high' by the system software to access the baud-rate-generator divisor latches.

Interrupts

One interrupt line is provided to the system. This interrupt is IRQ3 and is 'positive active'. To allow the serial port to send interrupts to the system, bit 3 of the modem control register must be set to 1 'high'. At this point, any of the following interrupt types 'enabled' by bits in the interrupt-enable register will cause an interrupt: Receiver-line status, Received Data available, Transmitter-Holding-Register empty, or Modem Status.

Interface Description

The communications adapter provides an EIA RS-232C electrically-compatible interface. One 2 by 8-pin Berg connector is provided to attach to various peripheral devices.

The voltage interface is a serial interface. It supports data and control signals as follows:

Pin A04	Transmit Data
Pin A08	Receive Data
Pin A03	Request to Send
Pin A07	Clear to Send
Pin A06	Data Set Ready
Pin B02-B08	Signal Ground
Pin A05	Carrier Detect
Pin A02	Data Terminal Ready
Pin B01	Shield Ground

The adapter converts these signals to/from TTL levels to EIA voltage levels. These signals are sampled or generated by the communications-control chip. These

signals can then be sensed by the system software to determine the state of the interface or peripheral device.

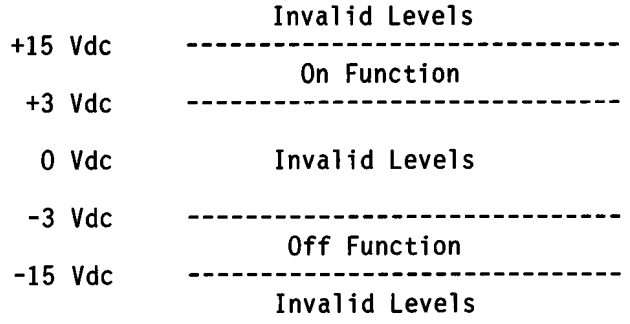
Note: The above nomenclature describes the communications adapter as a DTE (Data Terminal Equipment) device. Suitable adapters must be used to attach other devices such as serial printers.

Note: Ring Indicate is not supported on the PC*jr*.

Voltage Interchange Information

Interchange Voltage	Binary State	Signal Condition	Interface Control Function
Positive Voltage = Negative Voltage =	Binary (0) Binary (1)	= Spacing = Marking	= On = Off

Voltage Interchange Information



The signal will be considered in the 'marking' condition when the voltage on the interchange circuit, measured at the interface point, is more negative than

-3 Vdc with respect to signal ground. The signal will be considered in the 'spacing' condition when the voltage is more positive than +3 Vdc with respect to signal ground. The region between +3 Vdc and -3 Vdc is defined as the transition region, and considered an invalid level. The voltage which is more negative than -15 Vdc or more positive than +15 Vdc will also be considered an invalid level.

During the transmission of data, the 'marking' condition will be used to denote the binary state 1, and the 'spacing' condition will be used to denote the binary state 0.

For interface control circuits, the function is 'on' when the voltage is more positive than +3 Vdc with respect to signal ground and is 'off' when the voltage is more negative than -3 Vdc with respect to signal ground.

For detailed information regarding the INS8250A Communications Controller, refer to "Bibliography".

Output Signals

Output 1 (OUT 1), Pin 34: Output 1 of the 8250A is not supported in PC*jr* hardware.

Output 2 (OUT 2), Pin 31: Output 2 of the 8250A is not supported in PC*jr* hardware.

Accessible Registers

The INS8250A has a number of accessible registers. The system programmer may access or control any of

the INS8250A registers through the processor. These registers are used to control INS8250A operations and to transmit and receive data. For further information regarding accessible registers, refer to "Bibliography".

INS8250A Programmable Baud Rate Generator

The INS8250A contains a programmable baud rate generator that is capable of taking the clock input (1.7895 MHz) and dividing it by any divisor from 1 to (65535). The output frequency of the Baud Rate Generator is 16 x the baud rate [divisor number = (frequency input) / (baud rate x 16)]. Two 8-bit latches store the divisor in a 16-bit binary- format. These divisor latches must be loaded during initialization in order to ensure desired operation of the baud rate generator. Upon loading either of the divisor latches, a 16-bit baud-counter is immediately loaded. This prevents long counts on initial load.

The following figure illustrates the use of the baud rate generator with a frequency of 1.7895 MHz. For baud rates of 4800 and below, the error obtained is minimal.

> **Note:** The maximum operating frequency of the baud generator is 3.1 MHz. In no case should the data rate be greater than 4800 baud.

Desired Baud Rate	Divisor Used to Generate 16x Clock (Decimal)	(Hex)	Percent Error Per Bit Difference Between Desired and Actual
50	2237	8BD	.006
75	1491	5D3	.017
110	1017	1A1	.023
134.5	832	167	.054
150	746	12C	.050
300	373	175	.050
600	186	BA	.218
1200	93	5D	.218
1800	62	3E	.218
2000	56	38	.140
2400	47	2F	.855
3600	31	1F	.218
4800	23	17	1.291

Baud Rate at 1.7895 MHz

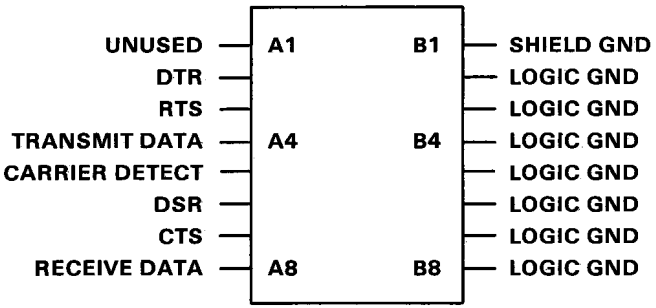
Note: These divisions are different than that used in the IBM Personal Computer. For portability, all initialization should be done through the system BIOS.

Note: Receive rates should not exceed 1200 baud if the receive operation is overlapped with keyboard keystrokes.

The following Assembly language sample program initializes the 8250. The baud rate is set to 1200 baud. It's data word is defined: 8 bits long with 1 stop bit odd parity.

```
BEGIN   PROC    NEAR
        MOV     AL,80H      ;  SET DLAB = 1
        MOV     DX,2FBH     ;  To Line Control Register
        OUT     DX,AL
        JMP     $+2         ;  I/O DELAY
        MOV     DX,2F8H     ;  Point to LSB of Divisor Latch
        MOV     AL,5DH      ;  This is LSB of Divisor
        OUT     DX,AL
        JMP     $+2         ;  I/O DELAY
        MOV     DX,2F9H     ;  Point to MSB of Divisor Latch
        MOV     AL,0        ;  This is MSB of Divisor
        OUT     DX,AL
        JMP     $+2         ;  I/O DELAY
        MOV     DX,2FBH     ;  Line Control Register
        MOV     AL,0BH      ;  8 Bits/Word, 1 Stop Bit,
                            ;  Odd Parity, DLAB = 0
        OUT     DX,AL
        JMP     $+2         ;  I/O DELAY
        MOV     DX,2F8H
        IN      AL,DX       ;  In Case Writing to Port LCR Caused
                            ;  Data Ready to go high
        ENDP
BEGIN
```

Assembly Language Sample Program



Connector Specifications

System Power Supply

The system power supply is a 33 Watt, three voltage-level, two-stage supply. The first stage is an external power transformer that provides a single-fuse protected, extra low, ac-voltage output. The power cord is 3.08 meters (10.16 feet) long. The second stage is an internal, printed-circuit board, which is vertically mounted into the system board. The second stage converts the transformer's ac-output into three dc-output levels.

The amount of power available on the I/O connector for a machine that is fully configured with internal features is 400 mA of +5 Vdc, 0 mA of +12 Vdc and 0 mA of -6 Vdc.

Power is supplied to the system board through a printed-circuit-board edge-connector. The diskette drive is powered through a separate four-pin connector mounted on the front edge of the Power Board. The power for the diskette drive fan is provided by a three-pin Berg-type connector mounted directly below the diskette-drive connector. Power is removed from the system board and diskette drive by a switch mounted on the rear of the Power Board. Both the switch and the transformer connector are accessible from the rear of the system.

Operating Characteristics

Power Supply Input Requirements

Voltage (Vac)			Frequency	Current (Amps)
Nominal	Minimum	Maximum	±.5 Hz	Maximum
120	104	127	60 Hz	.65 at 104 Vac

Voltage ac

D.C Outputs

Vdc Voltage	Current (Amps)		Regulation Tolerance
Nominal	Minimum	Maximum	±%
+5	*1.5	3.6	5
+12	.04	1.2	5
–6	0.0	.025	16

Voltage dc

* There must be a minimum of a 1.5 Amp load on the +5 Vdc output for the -6 Vdc to be present.

Over-Voltage/Over-Current Protection

Input (Transformer)

The following table describes the transformer input protection:

Voltage (Nominal)	Type Protection	Rating (Amps)
120 Vac	Non-resettable Fuse Thermal/Over-Current	5A Slo Blow

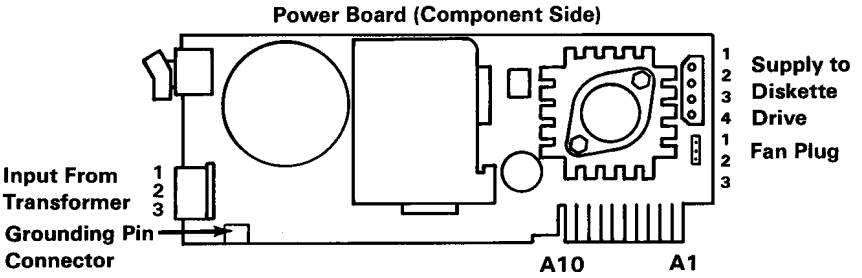
Input Protection

Output (Power Board)

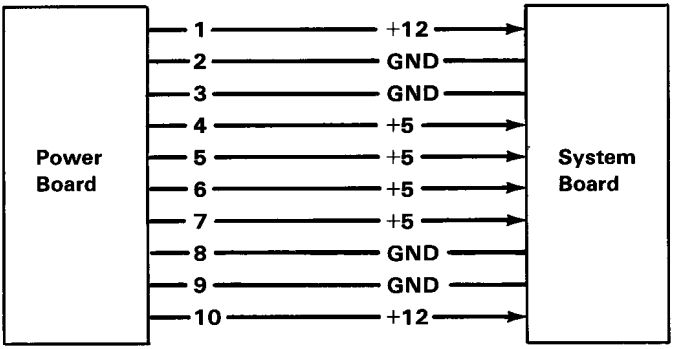
The following table describes the Power Board's output protection:

	Protection Condition	
Output Voltages	**Over-Voltage**	**Over-Current**
+5 Vdc	*6.3 ± .7 Vdc	**3.9 ± .25 Amps
12 Vdc	*14.4 ± 1.4 Vdc	2.2 ± .9 Amps
* Over-Voltage protection is provided by fuse F1. **Resettable by removing the fault condition and removing power for at least 5 seconds and then applying power.		

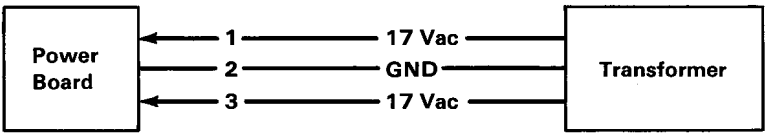
Output Protection



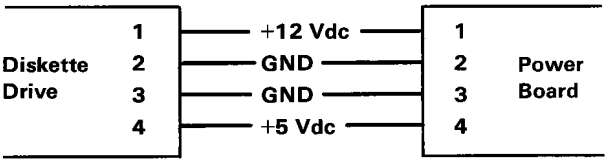
Connector Specifications



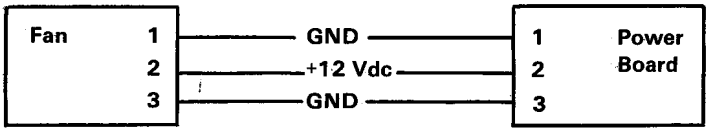
Connector Specifications



Connector Specifications



Connector Specifications



Fan Connector Specifications

Notes:

SECTION 3. SYSTEM OPTIONS

Contents

Notes:

IBM PC*jr* 64KB Memory and Display Expansion

The 64KB Memory and Display Expansion option enables the user to work with the higher density video modes while increasing the system's memory size by 64K bytes to a total of 128K bytes. The memory expansion option plugs into the 44-pin memory expansion connector on the system board. Only one memory expansion is supported.

The Memory Expansion Option does not require the user to reconfigure the system to recognize the additional memory.

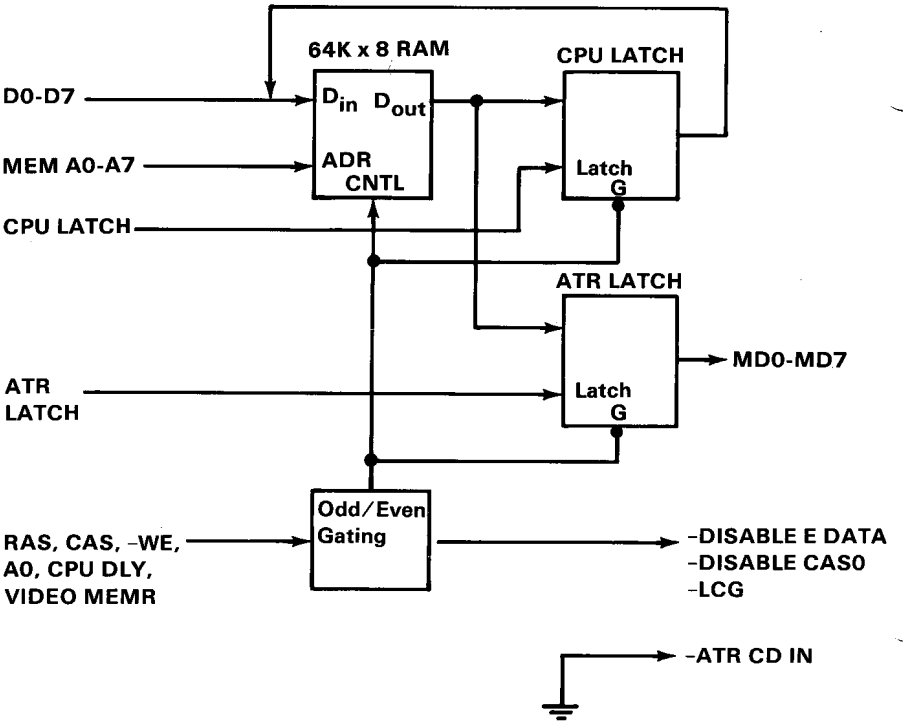
Eight 64K-by-1, 150 ns, dynamic memory modules provide 64K bytes of storage. The memory modules are Motorola's MCM6665AL15, and Texas Instrument's TMS4164-15, or equivalent.

When inserted, the memory expansion option uses the ODD memory space, while the system memory is decoded as the EVEN memory. Thus, when used as video memory, the memory expansion option has the video attributes while the on-board system memory has the video characters. This arrangement provides a higher bandwidth of video characters.

In addition to the eight memory modules, the expansion card has logic to do the EVEN/ODD address decoding, video data multiplexing, and a CARD PRESENT wrap.

Dynamic-refresh timing and address generation are done on the system board and used by the memory expansion option.

The following is a block diagram of the IBM PC*jr* 64KB Memory and Display Expansion.



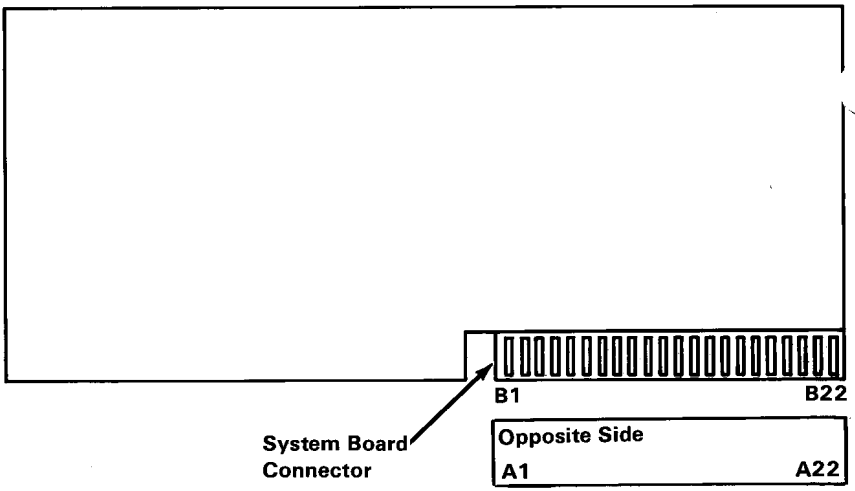
Memory Expansion Block Diagram

Signal	I/O	Description
+RAS	I	+Row Address Strobe. This line is inverted and then becomes the -RAS for the RAM modules.
+A0	I	Microprocessor Address 0. This is used to determine whether the microprocessor access is from the system board RAM (Low) or from the expansion RAM (High).
-DISABLE EDATA	O	When the expansion RAM card is in and the microprocessor is reading an ODD byte of data the expansion card tri-states the latch for EVEN data on the system board using this line.
ATR LATCH	I	This signal indicates that the expansion RAM card should 'latch' up data from the expansion RAM into the attribute latch.
MD0 thru MD7	O	These data lines contain CRT information from the attribute latch and go to the Video Gate Array.
D0 thru D7	I/O	These data lines are from the microprocessor and are bidirectional.
MEM A0 thru A7	I	These are the multiplexed address lines for the dynamic-RAM modules. These lines are multiplexed between row address and column

		address, and also between microprocessor and CRT addresses.
VIDEO MEMR	I	When this signal is 'high' it indicates a MEMR is accessing the system board or expansion RAM is being accessed. This line along with A0 determines if the expansion RAM microprocessor latch should 'gate' its data onto the D0 thru D7 Bus.
CPU DLY	I	This line when 'high' indicates that a microprocessor RAM cycle is occurring. It is used to gate 'off' the expansion RAM CAS or used with A0 to generate the -DISABLE CAS 0 signal.
-DISABLE CAS 0	O	This line is used to disable the system board CAS0 when a system microprocessor 'write' is occurring to the expansion RAM. This line keeps the 'write' from occurring to the system board RAM.
+CAS	I	Column Address Strobe. This line instructs the expansion RAM to 'latch' up the address on the MEM A0 thru A7 address lines.

-LCG	O	This line is used to instruct the system board that attributes or ODD graphics data should be 'read' from the expansion RAM card for use by the Video Gate Array.
GATE	I	This line is 'wrapped' and becomes the -LCG output.
-WE	I	This line instructs the memory that the cycle is a microprocessor 'write' cycle.
CPU LATCH	I	This line instructs the expansion RAM card to 'latch' the data from the expansion RAM into the microprocessor latch.
-ATR CD IN	O	This line is a wrap of the ground line on the expansion RAM card. It pulls 'down' an 8255 input so that the microprocessor can tell if this card is installed or not.

The following is the connector specifications for the IBM PC*jr* 64KB Memory and Display Expansion.



64KB Memory and Display Expansion

Connector Pin	Signal Name	Signal Name	Connector Pin
A01	+RAS	VIDEO MEMR	B01
A02	A0	CPU DLY	B02
A03	-DISABLE EDATA	-DISABLE CAS 0	B03
A04	ATR LATCH	+CAS	B04
A05	MD4	-LCG	B05
A06	MD5	GATE	B06
A07	MD6	Ground	B07
A08	MD7	Ground	B08
A09	MD0	Ground	B09
A10	MD1	-WE	B10
A11	MD2	CPU LATCH	B11
A12	MD3	-ATR CD IN	B12
A13	GND	GND	B13
A14	VCC	VCC	B14
A15	D7	D6	B15
A16	D5	D4	B16
A17	D3	D2	B17
A18	D1	D0	B18
A19	MEM A6	MEM A7	B19
A20	MEM A4	MEM A5	B20
A21	MEM A2	MEM A3	B21
A22	MEM A0	MEM A1	B22

Connector Specifications

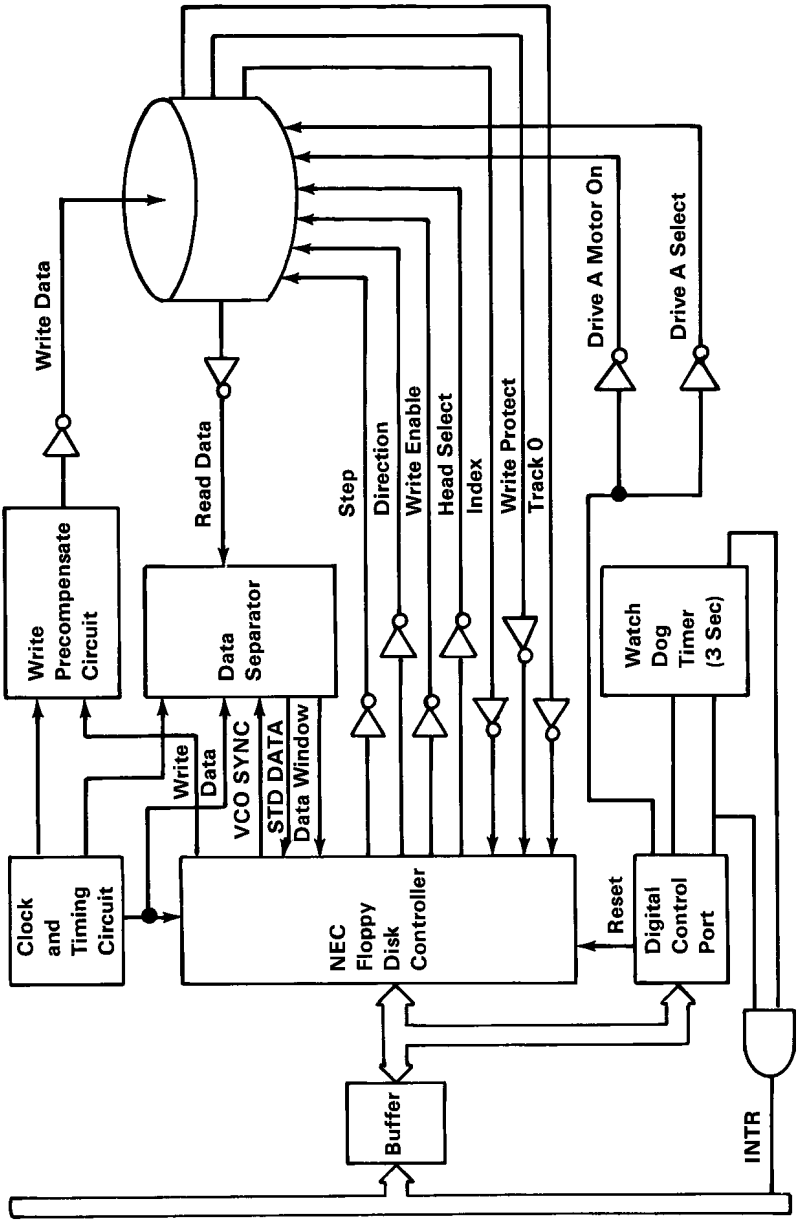
Notes:

IBM PC*jr* Diskette Drive Adapter

The diskette drive adapter resides in a dedicated connector on the IBM PC*jr* system board. It is attached to the single diskette drive through a flat, internal, 60-conductor, signal cable.

The general purpose adapter is designed for a double-density , Modified Frequency Modulation (MFM)-coded, diskette drive and uses write precompensation with an analog phase-lock loop for clock and data recovery. The adapter uses the NEC μPD765 or compatible controller, so the μPD765 characteristics of the diskette drive can be programmed. In addition, the attachment supports the diskette drive's write-protect feature. The adapter is buffered on the I/O bus and uses the system ROM BIOS for transferring record data. An interrupt level is also used to indicate an error status condition that requires processor attention.

A block diagram of the diskette drive adapter follows.



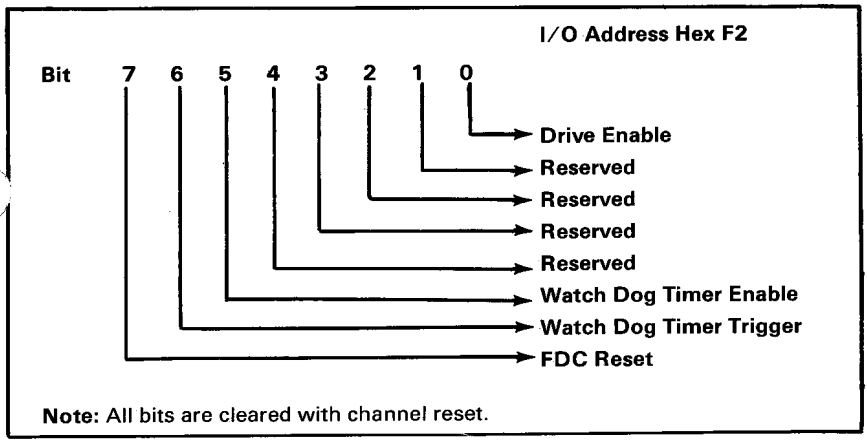
Diskette Drive Adapter Block Diagram

Functional Description

From a programming point of view, the diskette drive adapter consists of a 4-bit digital output register (DOR) in parallel with a NEC μPD765 or equivalent floppy disk controller (FDC).

Digital Output Register

The digital output register (DOR) is an output-only register used to control the drive motor and selection. All bits are cleared by the I/O interface reset line. The bits have the following functions:



Digital Output Register

Bit 0 This bit controls the motor and enable lines to the drive. When 'high' (1), this bit will turn 'on' the drive motor and 'enable' the drive. When 'low' (0), this bit will turn 'off' the drive motor and 'disable' the drive.

Bits 1-4 These bits are reserved.

Bit 5 When 'high' (1), this bit 'enables' the WatchDog Timer function and interrupt. When 'low' (0), this bit 'disables' the WatchDog Timer and interrupt.

Bit 6 This bit controls the start of a watchdog timer cycle. Two output commands are required to operate the trigger. A 1 and then a 0 must be written in succession to 'strobe' the trigger.

Bit 7 This bit is the hardware 'reset' for the floppy diskette controller chip. When 'low' (0), this bit holds the FDC in its 'reset' state. When 'high' (1), this bit releases the 'reset' state on the FDC.

WatchDog Timer

The WatchDog Timer (WDT) is a one to three-second timer connected to interrupt request line 6 (IRQ6) of the 8259. This timer breaks the program out of data transfer loops in the event of a hardware malfunction. The WatchDog Timer starts its cycle when 'triggered.'

Floppy Disk Controller (FDC)

The floppy disk controller (FDC) contains two registers that can be accessed by the system microprocessor: a status register and a data register. The 8-bit main-status register contains the status information of the FDC and can be accessed at any time. The 8-bit data register consists of several registers in a stack with only one register presented to the data bus at a time. The data register stores data, commands, parameters, and provides floppy disk drive (FDD) status information. Data bytes are read from or written to the data register in order to program or obtain results after

a particular command. The main status register can only be read and is used to facilitate the transfer of data between the system microprocessor and FDC.

FDC Register	I/O Address
Data Register	hex F5
Main Status Register	hex F4

Programming Summary

The FDC is set up with the following Parameters during system power up:

Parameter	Power-up Condition
Sector Size	hex 02 for 512 Byte Sectors
Sector Count	9
Head Unload	hex 0F - Has no effect on system operation.
Head Step Rate	hex D - This gives a step rate of 6 milliseconds.
Head Load Time	hex 1 Minimum head load time.
Format Gap	hex 50
Write Gap	hex 2A
Non-DMA Mode	hex 1
Fill byte for Format	hex F6

FDC Power-up Parameters Settings

The IBM PC*jr* Diskette Drive Adapter and BIOS use and support the following FDC commands:

- Specify
- Recalibrate
- Seek
- Sense interrupt status
- Sense Drive status
- Read data
- Write data
- Format a track

Note: Please refer to the Diskette section of the BIOS listing for details of how these commands are used.

The following FDC hardware functions are not implemented or supported by the IBM PC*jr* Diskette Drive Adapter.

- DMA data transfer
- FDC interrupt
- Drive polling and overlapped seek
- FM data incoding
- Unit select status bits

2 Heads (1 per side) 40 Cylinders (Tracks)/Side 9 Sectors/Track 512 Bytes/Sector Modified Frequency Modulation (MFM)

Diskette Format

Constant	Value
Head Load	Not Applicable
Head Settle	21 Milliseconds
Motor Start	500 Milliseconds

Drive Constants

Comments

1. Head loads when diskette is clamped.
2. Following access, wait Head Settle time before RD/WR.
3. Drive motor should be 'off' when not in use. Wait Motor Start time before RD/WR.
4. All system interrupts except IRQ6 must be 'disabled' during diskette data transfer in order to prevent data under-run or over-run conditions from occurring.

System I/O Channel Interface

All signals are TTL-compatible:

Most-Positive Up-Level	+ 5.5 Vdc
Least-Positive Up-Level	+ 2.7 Vdc
Most-Positive Down-Level	+ 0.5 Vdc
Least-Positive Down-Level	- 0.5 Vdc

The following lines are used by this adapter:

+D0 thru 7 (Bidirectional, Load: 1 74LS, Driver: 74LS 3-state)

	These eight lines form a bus through which all commands, status, and data are transferred. Bit 0 is the low-order bit.
+A0 thru 3	(Adapter Input, Load: 1 74LS)
	These four lines form an address bus by which a register is selected to receive or supply the byte transferred through lines D0-7. Bit 0 is the low-order bit.
-IOW	(Adapter Input, Load: 1 74LS)
	The content of lines D0-7 is stored in the register addressed by lines A0-3 at the trailing edge of this signal.
-IOR	(Adapter Input, Load: 1 74LS)
	The content of the register addressed by lines A0-3 is 'gated' onto lines D0-7 when this line is 'active.'
-RESET	(Adapter Input, Load: 1 74LS)
	A down level 'aborts' any operation in process and 'clears' the digital output register (DOR).
+IRQ6	(Adapter Output, Driver: 74LS 3-state)
	This line is made 'active' when the WatchDog timer times out.
-DISKETTE CARD INSTALLED	(Adapter Output, Driver: Gnd.)
	This line is pulled 'up' on the System Board and is wired to input port bit PC2 on port hex 62 of the

8255. This line is used by the program to determine if the diskette drive adapter is installed.

-Diskette CS (Adapter Input, Load: 1 74LS)

This line is shared with the modem CS line and is 'low' whenever the microprocessor is doing IOR or IOW to either the diskette adapter or the modem. This line should be conditioned with A9 being 'low' to generate a DISKETTE CS.

A9 (Adapter Input, Load: 1 74LS)

This line is the microprocessor address line 9. When this line is 'low' and -DISKETTE CS is 'low', IOR and IOW are used by the diskette adapter.

DRQ 0 (adapter Output, Driver: NEC μpd 765)

This output would indicate to a DMA device on the external I/O Channel that the diskette controller wants to 'receive' or 'transmit' a byte of data to or from memory.

DACK 0 (Adapter input, Load: NEC μpd 765)

This line should come from an external DMA and should indicate that a byte is being transferred from/to the Floppy Disk Controller to/from memory.

Drive Interface

All signals are TTL-compatible:

Most Positive Up Level	+ 5.5 Vdc
Least Positive Up Level	+ 2.4 Vdc
Most Positive Down Level	+ 0.4 Vdc
Least Positive Down Level	- 0.5 Vdc

All adapter outputs are driven by active collector gates. The drive should not provide termination networks to Vcc (except Drive Select which has a 2,000 ohm resistor to Vcc).

Each attachment input is terminated with a 2,000 ohm resistor to Vcc.

Adapter Outputs

-Drive Select (Driver: MC3487)

This line is used to 'degate' all drivers to the adapter and receivers from the adapter (except Motor Enable) when the line is not 'active.'

-Motor Enable (Driver: 74LS04)

The drive must control its spindle motor to 'start' when the line becomes 'active' and 'stop' when the line becomes 'inactive.'

-Step (Driver: MC3487)

The selected drive must move the read/write head one cylinder in or

out as instructed by the Direction line for each pulse present on this line.

-Direction (Driver: MC3487)

For each recognized pulse of the step line the read/write head should move one cylinder toward the spindle if this line is active, and away from the spindle if not-active.

-Write Data (Driver: 74LS04)

For each 'inactive' to 'active' transition of this line while Write Enable is 'active', the selected drive must cause a flux change to be stored on the diskette.

-Write Enable (Driver: MC3487)

The drive must 'disable' write current in the head unless this line is 'active.'

-HEAD SELECT 1 (Driver: MC3487)

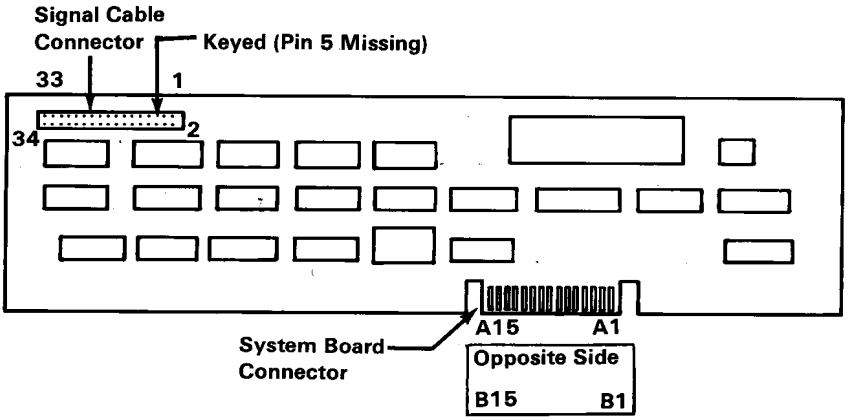
This interface signal defines which side of a two-sided diskette is used for data recording or retrieval. A 'high' level on this line selects the R/W head on the side 1 surface of the diskette. When switching from side 0 to side 1 and conversely, a 100 μs delay is required before any 'read' or 'write' operation can be initiated.

Adapter Inputs

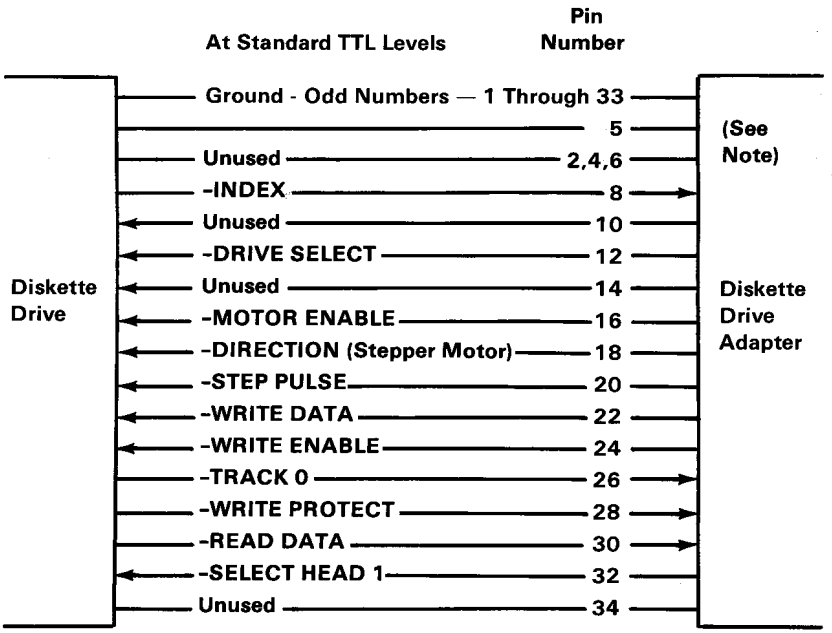
-Index	The selected drive must supply one pulse per diskette revolution on this line.
-Write Protect	The selected drive must make this line 'active' if a write-protected diskette is mounted in the drive.
-Track 0	The selected drive must make this line 'active' if the read/write head is over track 0.
-Read Data	The selected drive must supply a pulse on this line for each flux change encountered on the diskette.

Voltage and Current Requirements

The diskette drive adapter requires a voltage supply of +5 Vdc +/- 5% and draws a nominal current of 525 mA and a maximum current of 700 mA.

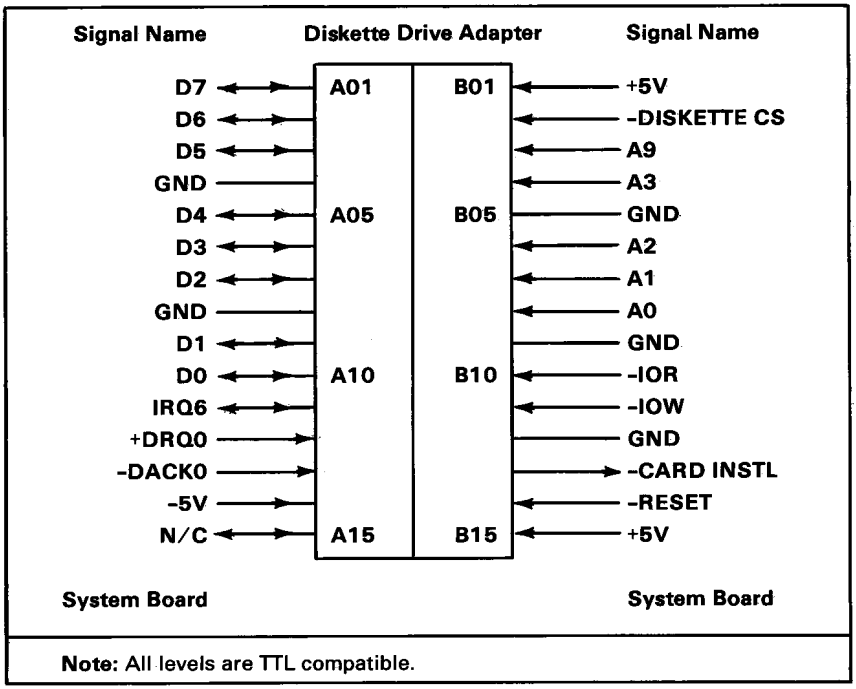


Diskette Drive Adapter



Note: Pin 5 is missing to match the key plug on the signal cable.

Connector Specifications (Part 1 of 2)



Note: All levels are TTL compatible.

Connector Specifications (Part 2 of 2)

IBM PC*jr* Diskette Drive

The system unit has space and power for one diskette drive. The drive is double-sided with 40 tracks for each side, is fully self-contained, and consists of a spindle-drive system, a read- positioning system, and a read/write/erase system.

Functional Description

The diskette drive uses modified frequency modulation (MFM) to read and write digital-data, with a track-to-track access time of 6 milliseconds.

To load a diskette, the operator rotates the load lever at the front of the diskette drive clockwise and inserts the diskette into the slot. Plastic guides in the slot ensure the diskette is in the correct position. Closing the load lever centers the diskette and clamps it to the drive hub. This same action also loads the Read/Write heads against the surfaces of the diskette. The load lever is mechanically interlocked to prevent closing of the lever if a diskette is not installed.

The head-positioning system moves the magnetic head to come in contact with the desired track of the diskette. Operator intervention is not required during normal operation. If the diskette is write-protected, a write-protect sensor 'disables' the drive's circuitry, and an appropriate signal is sent to the interface.

Data is read from the diskette by the data-recovery circuitry, which consists of a low-level read-amplifier, differentiator, zero-crossing detector, and digitizing circuits. All data decoding is done by the adapter card.

The IBM PC*jr* Diskette Drive is equipped with a media cooling fan, which gets its power from the power supply board.

The diskette drive also has the following sensor systems:

- The track 00 sensor, senses when the head/carriage assembly is at track 00.
- The index sensor, which consists of an LED light source and phototransistor. This sensor is positioned so that when an index hole is detected, a digital signal is generated.
- The write-protect sensor 'disables' the diskette drive's electronics whenever it senses a write-protect tab on the diskette.

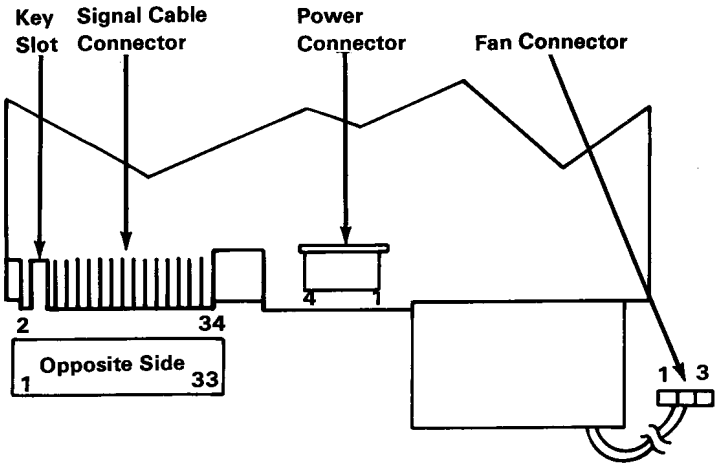
The drive requires power within the following specifications:

Specification	+5 Vdc Input	+12 Vdc Input
Nominal Supply	+5 Vdc	+12 Vdc
Ripple (0 to 50 kHz)	100 mV	100 mV
Tolerance (Including Ripple)	±5%	±5%
Standby Current (Nominal)	600 mA	400 mA
Standby Current (Worst Case)	700 mA	500 mA
Operating Current (Nominal)	600 mA	900 mA
Operating Current (Worst Case)	700 mA	2400 mA

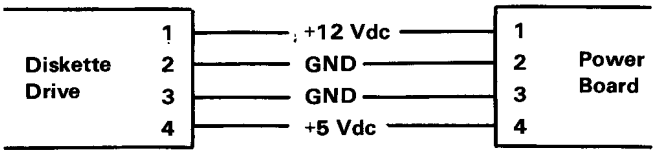
Diskette Drive Power Specifications

For interface information refer to "Diskette Drive Adapter" in this section.

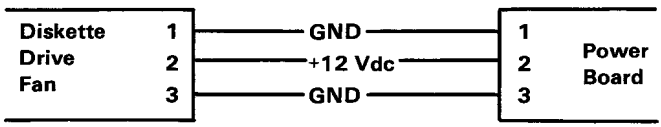
For mechanical and electrical specifications see Appendix D.



Diskette Drive Connectors



Connector Specifications (Part 1 of 2)

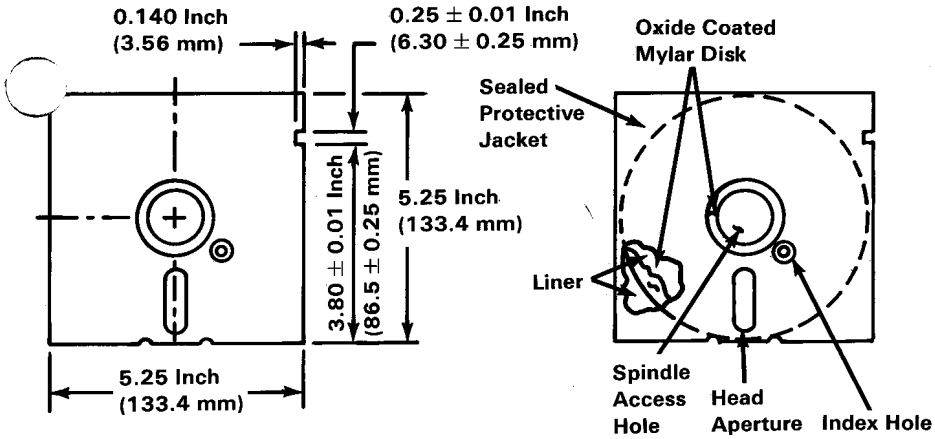


Connector Specifications (Part 2 of 2)

Notes:

Diskette

The IBM PC*jr* Diskette Drive uses a standard 133.4 mm (5.25 in.) diskette. For programming considerations, single-sided, double-density, soft-sectored diskettes are used for single-sided drives. Double-sided drives use double-sided, double-density, soft-sectored diskettes. The figure below is a simplified drawing of the diskette used with the diskette drive. This recording medium is a flexible magnetic disk enclosed in a protective jacket. The protected disk, free to rotate within the jacket, is continuously cleaned by the soft fabric lining of the jacket during normal operation. Read/write/erase head access is through an opening in the jacket. Openings for the drive hub and diskette index hole are also provided.



Recording Medium

Notes:

IBM PC*jr* Internal Modem

The IBM PC*jr* Internal Modem is a 65 mm (2.5 inch) by 190 mm (7.5 inch) adapter that plugs into the PC*jr* system board modem connector. The modem connector is an extension of the system I/O bus. All system control signals and voltage requirements are provided through a 2 by 15 position card-edge tab with 0.254 cm (0.100-inch) spacing on the modem adapter.

Functional Description

The Internal Modem consists of two major parts: (1) the INS8250A Asynchronous Communication Element, and (2) the Smart 103 Modem. Therefore, the programming must be considered in two parts. The INS8250A communications protocol is a function of the system ROM BIOS, and is discussed later in this section. All 'pacing' of the interface and control-signal status must be handled by the system software. After the INS8250A is initialized, the modem is controlled by ASCII characters transmitted by the INS8250A.

Key features of the INS8250A used in the modem adapter are:

- Adds or deletes start bits, stop bits, and parity bits to or from the serial data stream
- Full double-buffering eliminates the need for precise synchronization
- Independently-controlled transmit, receive, line status, and data-set interrupts
- Programmable baud-rate-generator allows division of the baud clock by 373 (hex 175) for a 300-bps transmission-speed or 1017 (hex 3F9) for a 110-bps transmission-speed to generate the internal 16 x clock

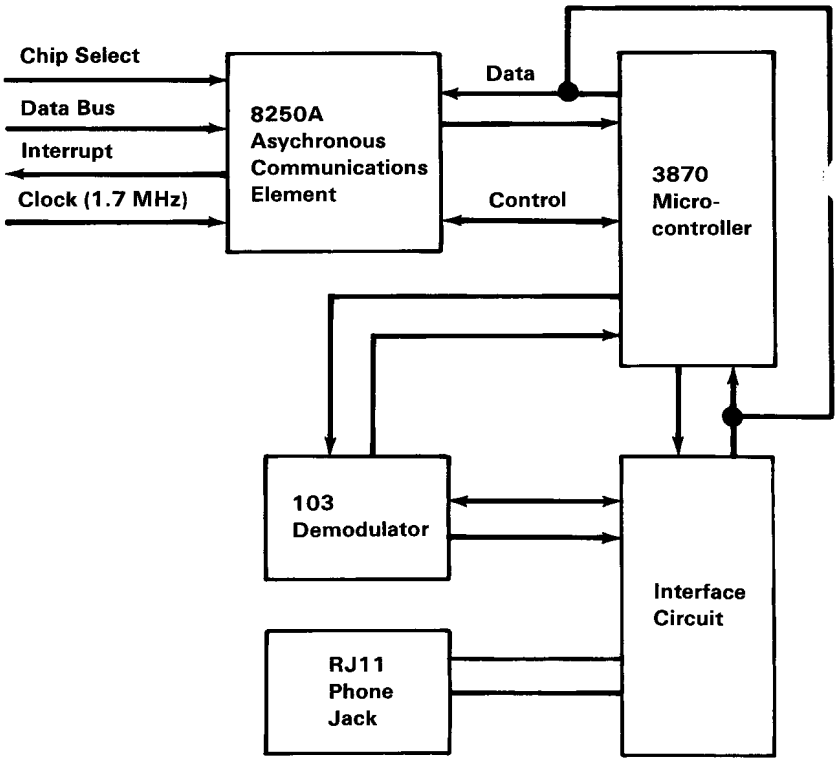
- Modem-control functions: Clear to Send (CTS), Data Set Ready (DSR), Data Terminal Ready (DTR), Ring Indicator (RI), and Data Carrier Detect (DCD)
- Fully-programmable serial-interface

  characteristics:
  - 7, or 8-bit characters
  - Even, odd, or no-parity bit generation and detection
  - 1 stop-bit generation
  - Baud-rate generation
- False-start bit detection
- Complete status reporting capabilities
- Line-break generation and detection
- Internal-diagnostic capabilities
  - Loopback controls for communications-link fault-isolation
  - Break, parity, overrun, framing-error simulation
- Fully prioritized-interrupt system-controls

Key features of the Smart 103 Modem used on the IBM PC*jr* Internal Modem are:

- Direct connection to a telephone company line through an FCC Part-68-approved permissive connection
- Compatible to Bell Series 100 originate/answer for modulation and handshaking
- All functions controlled by ASCII characters and INS8250A modem-control lines
- Uses modular phone-jack (USOC RJ11)
- Data rate is either 300 or 110 bits-per-second
- Auto/manual originate
- Auto/manual answer
- Communication mode is full duplex on two-wire, switched-network channels

- Auto dialer; either DTMF ([dual-tone modulated-frequency] touch-tone) or pulse-dialing (rotary dial) by software command
- Tandem dialing
- Call-progress reporting
- Dial-tone, ring-back tone, and busy-tone detection



IBM PC*jr* Internal Modem Block Diagram

Modem Design Parameters

The following tables describe the design parameters of the Smart 103 Modem.

Dialer Type:	Two modes 1. Forced Touch-Tone (DTMF) dialing 2. Forced pulse dialing
Tandem Dialing:	The ASCII character P (hex 50 or 70) in the dial string causes a delay of up to 10 seconds while the modem is searching for another dial tone. A time out will cause the modem to hang up and post status. The ASCII character W (hex 57 or 77) in the dial string causes a 5-second dead wait before continuing to dial. Multiple ASCII W's will cause multiple waits.
Pulse Dialing:	Rate: 10 + 1, –0 pulses per second Duty Cycle: 60% make, 40% break Interdigit Delay: 800 ms ± 50 ms
DTMF Dialing:	Tone Duration: 85 ms ± 10 ms Intertone Duration: 80 ms ± 10 ms

Dialer Parameters (Part 1 of 2)

Tone Pair Frequencies:		
ASCII Digit Code	**Frequency (Hz)**	
0	941	1336
1	697	1209
2	697	1336
3	697	1477
4	770	1209
5	770	1336
6	770	1477
7	852	1209
8	852	1336
9	852	1477
*	941	1209
#	941	1477

Dialer Parameters (Part 2 of 2)

Time Out Duration:	A data call will time out if an answer tone is not detected within 45 seconds of the last digit dialed.

Failed Call Time Out Parameter

Modulation:	Conforms to Bell 103/113 specification using binary phase-coherent frequency shift keying (FSK).

Modulation Parameter

Mode	Originating End	Answering End
Transmit	1070 Space 1270 Mark	2025 Space 2225 Mark
Receive	2025 Hz Space 2225 Hz Mark	1070 Hz Space 1270 Hz Mark

Transmitter/Receiver Frequency Parameters

Receive Sensitivity	More negative or equal to –42 dBm.

Receive Sensitivity Parameters

Transmitter Level	Fixed at –10 dBm as per FCC Part 68 Permissive connection.

Transmitter Level Parameter

Programming Considerations

The modem and the IBM PC*jr* system can communicate commands or data between each other. Any commands sent to the modem from the IBM PC*jr* are stripped from the data stream and executed but are not transmitted to the receiving station. The data is transparent to the modem. The modem is capable of causing hardware interrupts as the result of certain conditions, and in response to queries for its status.

Commands to the modem are a sequence of characters preceded by a single command character. The command character tells the modem that the following character sequence, until a carriage return, is a command. The carriage return completes the command sequence and causes the modem to execute the commands. The command character (represented by [cc] in the following text) is programmable (with the **NEW** command) to any ASCII character (hex 00 thru 7F). The default for the command character is Ctrl N (ASCII hex 0E).

Commands can occur anywhere in the data stream if properly formatted but are not to be executed by the modem until a carriage return is received.

Multiple commands are allowed if separated by commas and preceded by a single command character.

Command Format

The following is the command format that all commands must follow.

[cc][command word][delimiter][arguments] [,more][CR]

where:

[cc]	is the single ASCII command character.
[command word]	is the command word or the first letter of the command word.
[delimiter]	is always a space when separating an argument and command word. Any spaces thereafter are ignored until the modem sees a comma, an argument or a carriage return.
[arguments]	is a variable that is replaced by any character allowed by the command definition.
[,more]	is any additional commands preceded by a comma.
[CR]	is a carriage return that completes the command sequence and causes the modem to execute the commands.

The following are two examples of command format.

[cc] COUNT 5 [CR]
sample test **[cc] VOICE, D** (408)
555-1234,**QUERY [CR]**

Format Guidelines

1. Commands can occur anywhere in the data stream if properly formatted but are not be executed by the modem until a carriage return is received.
2. Multiple commands are allowed if separated by commas and preceded by a single command-character.
3. Only the first character of the command word is significant. All remaining characters are ignored up to the first space following the command word. In other words, the **DIAL** command and DUMMY are treated identically.

4. The modem does not discriminate between upper-case and lower-case characters.
5. There are three ways to send the current command-character as data to a receiving station:
   a. Consecutively sending it twice:
      [cc][cc]
      This would send the character a single time.
   b. Change the command character (with the **NEW** command) to another ASCII character and then transmit the previous command-character.
   c. Place the modem in the Transparent mode and then transmit the character.

Commands

The commands that are used with the integrated modem are listed on the following pages in alphabetical order.

Each of the commands has its syntax described according to the following conventions:

1. Words in capital letters are keywords. Only the first letter of the keyword is required, the others are optional.

2. You must supply any arguments which are in lower-case letters. Valid characters for arguments are defined as:

   - m - ASCII decimal digits 0 to 9, *, #, I, P, and W
   - n - ASCII hexadecimal digits 0 to F
   - o - ASCII hexadecimal digits 0 to 9
   - p - any ASCII character

3. All arguments are examined for validity. If extra characters are used in an argument, the extra characters are ignored. If the argument is invalid, the command is ignored.

4. An ellipsis (...) indicates an item may be repeated as many times as you wish.

5. All command lines must begin with a command character. The default command-character is (CONTROL N).

6. Multiple commands separated by commas can follow a single command-character.

An example of the **DIAL** command is given below:

Command format - **DIAL m...m**

Command line - DIAL 1 800 555 1234

If an invalid argument or no argument is given, the command is not executed. Also, a question mark (?) is given as the error response and the command line is aborted.
The commands are as follows:

Format: **ANSWER**

A

Purpose: To logically take the phone off the hook and force **ANSWER** mode. This is logically like a manual answer.

Format: **Break n**

Purpose: To send a space or break character for a duration equal to a multiple of 100 ms (**n** x 100 ms).

Format: **COUNT n**

C n

Where **n** is the number of complete rings in the range of hex 0 to hex F.

When answering an incoming call, the modem answers the phone after **n** complete incoming rings, where **n** is any value from hex 0 to F.

A value of zero specifies that the modem not answer an incoming call, but still carry out any instructions from the host.

When dialing, the modem waits **n** + 3 complete ringbacks before cancelling the call.

If **n** exceeds 4, the 45-second abort timer cancels an outgoing call with an "UNSUCCESSFUL" response, as more than seven ringbacks exceeds 45 seconds.

Purpose: Sets the ring count when the modem is answering an incoming call or dialing a call.

Default: 0

Format: **DIAL m...m**

D m...m

Where **m...m** is a dial string of ASCII decimal digits 0 through 9, *, #, I, P, and W. A maximum of 33 characters are allowed in the dial string. The first character of the string defaults to P (a 10-second delay while searching for the dial tone). W causes the modem to delay five seconds, then continue dialing.

W or P must start a string, can also occur anywhere within a string, and causes the digits to be tone dialed.

The characters * and # represent the two extra buttons on a push-button phone, but may be used for other things.

I causes the next digits to be pulse dialed. The I stays in effect until a (P,), (W,),or end of command. The modem then searches for line busy, ringing, or incoming carriers while posting the status.

Purpose: To cause the modem to dial.

Default: P (10-second timeout). (If this command is used without an argument, the last number dialed is redialed once.)

Format: **FORMAT n**

F n

Where **n** is one of the following:

n	Parity	Data Length	Stop Bit
0	Mark	7	1
1	Space	7	1
2	Odd	7	1
3	Even	7	1
4	None	8	1
5-7	Reserved		

The 8250A line control register (LCR) must specify the same format as defined in the **FORMAT n** command to 'enable' data/command communication.

Do not combine this command with any other commands except the **SPEED** command on a single command line.

> **Note:** If programming in BASIC, this command must be used in addition to specifying the same parity and data length in the BASIC 'open' statement.

Purpose: To change the parity and number of stop-bits being transmitted at either end, to a new format.

Default: 3

Format: **HANGUP**

H

Purpose: To perform a clean disconnect and go on-hook. Logically the same as manually hanging up.

Format: **INITIALIZE**

I

This command is executed in 10 seconds and is the same as a cold start. An "OK" response is not returned after execution and the integrity test code in the **QUERY** command is set.

Purpose: Places the modem in the power-up default-state.

Format: **LONG RESPONSE o**

L o

Where **o** is one of the following:

o	Mode	Responses
0	Verbose	"BUSY" "CONNECTED" "NO ANSWER" "NO DIAL TONE" "OK" "RING" "UNSUCCESSFUL" "?" (Question Mark)
1	Terse (Hex code)	30 31 32 33 34 35 36 37

Note: The dial string is not echoed in the terse mode.

Purpose: Modifies message feedback. Information is posted in the status area.

Default: 0 (Verbose mode)

Format: **MODEM**

M

Purpose: Forces the modem into the data state where the carrier is placed on the telephone line and proper connection-protocols are followed.

This command is equivalent to **ANSWER** if the data state started as autoanswer.

Format: **NEW p**

N p

where **p** is any ASCII character.(hex 0E)

Purpose: Changes the command character to an ASCII character.

Default: Ctrl N (ASCII hex 0E)

Format: **ORIGINATE**

O

Purpose: Logically takes the phone off-hook and forces the ORIGINATE mode. Logically equivalent to manual originate.

Format: **PICKUP**

P

Purpose: Logically takes the phone off-hook and puts the modem in the voice state.

Format: **QUERY**

Q

Purpose: To query the modem for its status information.

Possible characters returned by the modem are as follows:

Responses	Meaning
H0 or H1	Hook status: H0 = on-hook, H1 = off-hook.
S0 to SF	Current ringcount setting in hex.
B	Line busy.
D	Line dead: no dial-tone found or no ring/no busy timeout after dialing.
L	Successful dial and handshake.
N	Dial not recorded: dial tone present after dialing.
X	No answer: ringcount plus 3 exceeded.
T0	Integrity test passed.
T1	Integrity test failed.

The first group of characters is always returned for a **QUERY** command. The second group of characters is returned only after a dialing sequence has been started or a change has occurred in the dialing status. The third group of characters is returned when a **TEST** command has occurred. All characters except the first group are erased by being read and do not appear in response to the next **QUERY** unless the

condition has recurred in the interim. The **QUERY** response overrides any incoming data from the telephone line.

Format: **RETRY**

R

Purpose: When placed after a **DIAL** command, it causes the modem to execute up to 10 redials at a rate of one per 40 seconds. The redials are triggered by a busy detection after dialing.

Format: **SPEED o**

S o

Where **o** is one of the following:

o	**bps**
0 -	110
1 -	300
2 -	Reserved

> **Note:** Do not combine this command with other commands except the **FORMAT** command on a single command line.

The **SPEED** command must be issued before the 8250A baud rate is changed.

> **Note:** If programming in BASIC, this command must be used in addition to specifying the same bps rate in the BASIC 'open' statement.

Purpose: Sets the baud rate.

Default: 1 (300 bps)

Format: **TRANSPARENT n...n**

T n...n

Where **n...n** is the number of bytes to transmit in the range of hex 0 to hex FFFF.

Purpose: Places the modem in the transparent mode for the next **n...n** bytes.

The modem does not look for command sequences but instead transmits every character it receives.

The argument can be up to four ASCII-coded hex digits long. This provides a range of 65,536 bytes.

If an argument is not included with the **TRANSPARENT** command, the command is ignored because it has no default.

The transparent mode is terminated when:

1. **n...n** characters have been transmitted.
2. Loss of carrier timeout.
3. INS8250A OUT 1 pin goes 'active.' (The INS8250A -OUT 1 signal should remain 'active' until the transparent mode is requested again.)

The modem exits the transparent mode before processing the next complete character from the host.

To re-enter the transparent mode, the sequence is:

1. The INS8250A -OUT 1 pin changes to, or remains in the 'inactive' state.

2. The command string containing the **TRANSPARENT** command is issued.

   An argument of 0 causes a permanent transparent mode which can be exited by the INS8250A -OUT 1 pin going 'active.'

Format: **VOICE**

V

Purpose: Forces the modem to the voice state where no tones or carriers are placed or searched for on the telephone line.

This state is used for voice communication, when the modem is an autodialer or answering device only. It is also necessary to be in the voice state to transmit DTMF tone-pairs.

This command 'disables' the autoanswer function.

The status responses are:

1. If a busy signal is detected "BUSY OK".

2. Any other condition "OK...(16 dots)....CONNECTED".

Format: **WAIT**

W

Purpose: Causes the modem to take no action, including autoanswer, until the next command is received from the host. All commands following the **WAIT** command in a single command-line are ignored.

Format: **XMIT m...m**

X m...m

Purpose: Instructs the modem to transmit the DTMF tone-pairs found in the argument string **m...m.** This is only valid in the voice state. Delays between digits can be caused by inserting W's in the string.

Each W causes a five-second delay.

Format: **ZTEST o**

Z o

Where **o** is one of the following:

o	**Test**
0 -	Hardware Integrity Test
1 -	Analog Loop Back Test

Purpose: Places the modem in the test mode specified by the argument.

For modes other than the integrity test, the modem stays in the test mode until any other command is received.

For the integrity test, the test is performed, status posted, and then the modem returns to service immediately. The integrity test takes eight to 10 seconds to execute and its completion is signaled by an "OK" message.

All commands following the **ZTEST** command in a single command-line are ignored.

Responses

Autoanswer

If -DTR is 'active', the modem goes off-hook and proper connection protocols including the two-second billing delay are followed. If connection is made, the modem sends "CONNECTED" to the host and posts the status in the status area.

Editing/Changing Command Lines

Corrections to the command line can be performed by aborting current-command lines and typing a new line or by entering the correct command later on in the current-command line.

The last command entered on a single command-line supersedes any previously entered command that performs an opposite function.

A Control X or backspace received by the modem immediately aborts the entire command line.

Opposite Commands

The command line is scanned after its completion (after [CR] is entered). Commands which cause an action during the scan (for example, **DIAL**) are not candidates for opposite treatment. Only commands which 'preset' a static condition can be opposites.

They include:

Count (n)	two entries, latest are used
Format (n)	two entries, latest are used
New (p)	two entries, latest are used
Speed (n)	two entries, latest are used
Transparent n..n	two entries, latest are used
Modem - Voice	these are opposites only when on-hook

Note: Answer and originate are not opposites; each of these causes an action when scanned.

Status Conditions

The modem sends the host messages as defined in the **LONG RESPONSE** command for dialing success or failure. Hardware interrupts for carrier loss and detecting incoming rings are provided on the 8250A.

Dialing and Loss of Carrier

The dialing process begins with the modem searching for a dial tone if it is not in the blind dialing mode. If a dial tone is not detected, the modem hangs up, the appropriate status characters are posted, and the "NO DIAL TONE" message is returned to the host.

If a dial tone is found, the modem continues to dial. When a P is encountered in the dial string, the modem

delays for up to 10 seconds to search for another dial tone and returns the "NO DIAL TONE" message to the host if a dial tone is not detected. When a W is encountered in the dial string, the modem delays for five seconds before continuing to dial. Consecutive W's are allowed in a dial string.

Anytime a P or W is not followed with an I in a dial string, the next digits are tone-dialed. When an I follows a P or W, all following digits are pulse-dialed until a P, W, or end of command ([CR]) is detected.

The modem ignores any character except 0 through 9, *, #, I, P, or W while dialing. This allows the user to place parentheses and dashes in the dial string for greater legibility.

The modem checks the telephone line again after it has dialed the digits in the dial string. If a dial tone is found immediately, the dialed digits are not recorded and the modem posts this to the status characters, hangs up, and sends the "UNSUCCESSFUL" message to the host. If the line is busy, this is also posted to the status characters and the modem hangs up and returns the "BUSY" message to the host. If the line is ringing, the modem begins counting the number of rings. If this count exceeds the value of **COUNT** + 3, the modem hangs up and takes the same actions as above. If no answer tone is detected within 45 seconds after completion of dialing, the modem hangs up and takes the same actions as above.

Finally, if the call is answered, the modem either looks for a carrier and begins the handshake sequence (if it is in the data or modem state) or remains silent (if it is in the voice state). In the voice state, the modem looks for busy, and transmits a response (1) when the line is

found not busy, or (2) if it is found busy, in which case it also hangs up and possibly dials again. In voice state, ringback count and abort time out are not used.

If, during the process of establishing the data link after dialing, the modem receives any character from the host or - DTR goes 'inactive', the modem aborts the call with a clean disconnect, clears the balance of the command line, and sends an "OK" message. Also, the modem does not carry out the instruction sent from the host, even if the character is a command character.

In the data state, the modem transmits a message after successful completion of the handshake, or after it has determined that the handshake failed. An unsuccessful handshake is evidenced by absence of carrier at the proper time.

If a carrier drops out for more than two seconds in the data state, the modem begins a timeout lasting approximately 17 seconds. At the end of the timeout, the modem hangs up. Any command received during the 17 seconds resets the timer.

The modem does not automatically reestablish the connection if the carrier returns after this dropout interval. This allows the user or software to intercede by commanding the modem to go into the voice state, to hang up immediately, or to take some other action. The data connection may also be terminated by a **HANGUP** command while carriers are still present. A voice connection is always terminated by a **HANGUP** command.

Default State

Upon power up or after an **INITIALIZE** command is given, the modem returns to the default state as follows:

- A verification of hardware integrity is performed and the result posted to the status characters.
- The remaining status characters cleared.
- The modem is placed in the data state awaiting a dialing request or incoming ring.
- The Transparent mode is cleared.
- All loopback modes are cleared.
- The wait mode is cleared.
- The command character is set to Control-N.
- The data format is set to 7 data bits, even parity, and one stop bit.
- Ringcount is set to 0 (auto answer 'disabled')
- The modem is set to on-hook.
- The message mode is set to verbose.

Programming Examples

Call progress reporting is done in two modes, verbose messages or terse messages as defined in **LONG RESPONSE** command to the Serial In (SIN) pin of the 8250A. The power-up default is the verbose messages mode, and these messages from the modem are in capital letters. Also, in call progress reporting, the status area is updated.

The following examples are representative of real-time call-progress reporting. The italicized entries are user entries.

Example 1:

OK *[cc]Dial 555-1234 [CR]*
NO DIAL TONE
OK

In this example, no dial tone is detected within the time out period.

Example 2:

OK
[cc]Dial 555-1234 [CR]

5551234.....
RING .....CONNECTED OK

In this example, a modem answer tone is detected.

Example 3:

OK
[cc]Dial 1(301)555-1234 [CR]
13015551234..... BUSY
OK

In this example, busy is detected.

Example 4:

OK
[cc]Dial 555-1234 [CR]
5551234.....
RING.....
RING.....
RING.....NO ANSWER
OK

In this example, ring count is exceeded before ringing stops.

Example 5:

OK
[cc]Dial 555-1234 [CR]
5551234.....
RING.....
.....
.....UNSUCCESSFUL
OK

In this example, a failed-call time-out occurred because an answer tone was not detected within the allotted time.

Example 6:

OK
[cc]Dial 99P555-1234 [CR]
99.....
.....NO DIAL TONE
OK

In this example, the second dial-tone is not detected within the time out period.

Example 7:

OK
[cc]Dial 99P421-7229 [CR]
99.....BUSY
OK

In this example, busy is detected within the time-out period.

Example 8:

OK
[cc]Dial 99WW555-1234 [CR]
99.....
.....
.....
.....
4217229....
RING.....CONNECTED OK

In this example, the access code is dialed and two dead waits are performed. Then, the second number is dialed and a modem answers.

Example 9:

OK
[cc]Dial 555-1234, Retry [CR]
5551234.....BUSY
5551234.....BUSY
5551234.....CONNECTED OK

In this example, the modem dials a number with auto redial.
The first two times, the number is busy.
The third time, a modem answers.

Modes of Operation

The different modes of operation are selected by programming the 8250A Asynchronous Communication Element. This is done by selecting the I/O address (hex 3F8 to 3FF) and writing data out to the card.

The 8250A is externally programmed to provide asynchronous, ASCII, 10 bit character length including start, stop, and parity on the serial-output pin (SOUT, pin 11). The data rate is 110 or 300 bits-per-second. The commands can be either upper-case or lower-case characters. See the command, **Format [n]**, earlier in this section for additional information.

For further information refer to "Bibliography."

Hex Address	Register Selected	Input/ Output	Mode 1	Mode 2	Notes
3F8	Transmit Buffer	Write	XX	XX	*
3F8	Receive Buffer	Read	XX	XX	*
3F8	Divisor Latch LSB	Write	75	F9	**
3F9	Divisor Latch MSB	Write	01	03	**
3F9	Interrupt Enable	Write	0F	0F	*
3FA	Interrupt Identification	Read	XX	XX	
3FB	Line Control	Write	1A	03	
3FC	Modem Control	Write	01	01	
3FD	Line Status	Read	XX	XX	
3FE	Modem Status	Read	XX	XX	
3FF	Scratch Pad	Write	XX	XX	

*DLAB = 0 (Bit 7 in line control Register).
**DLAB = 1 (Bit 7 in line control Register).
Mode 1 - 300 BPS - 7 Data Bits, 1 Stop Bit, Even Parity.
Mode 2 - 110 BPS - 8 Data Bits, 1 Stop Bit, No Parity.

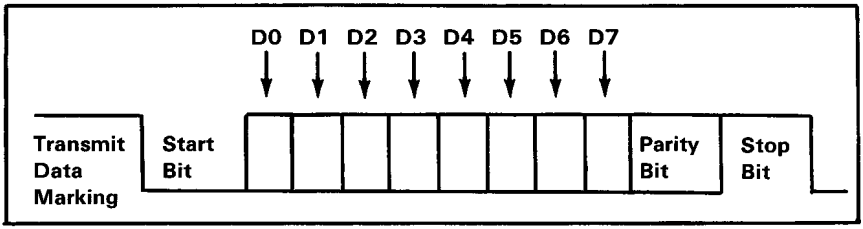
8250A Register Description

Interrupts

One interrupt line is provided to the system. This interrupt is IRQ4 and is 'positive active.' The interrupt enable register must be properly programmed to allow interrupts.

Data Format

The data format is as follows:



Transmitter Output and Receiver Input Data Format

Data bit 0 is the first bit to be transmitted or received. The attachment automatically inserts the start bit, the correct parity-bit if programmed to do so, and the stop bit.

Interfaces

8250A to Modem Interface

The following describes the 8250A to 103 modem interface:

Signal	Description

INS8250A -OUT 1	The 'inactive' state enables entry into the transparent mode using the **UNLISTEN** command. The 'active' state 'disables' the transparent mode.
-OUT 2	No connection.
SOUT	Serial output from the 8250A.
-RTS	-Request To Send No connection.
-DTR	-Data Terminal Ready 1. To accept a command, -DTR must be 'active.' 2. If -DTR goes 'inactive', the modem does a clean disconnect sequence. 3. In auto-answer mode, the modem does not go off-hook, but RI on the 8250A will be toggled if the ringing signal is present.
SIN	Serial input to the 8250A.
-RI	The ring indicator pulses with an incoming ring voltage.
-CTS	-Clear To Send

	This line is wired 'active' on the modem adapter.
-DSR	-Data Set Ready This line is wired 'active' on the modem adapter.
-RLSD	-Received Line Signal Detect When 'low', this line indicates the data carrier has been detected. If the carrier drops out for longer than two seconds, this line goes 'inactive' and starts the timeout timer.
-RESET, +XRESET	These lines are used to reset or initialize the modem logic upon power-up. These lines are synchronized to the falling edge of the clock. Its duration upon power up is 26.5 ms -RESET is 'active low'. +XRESET is 'active high.'
A0,A1,A2,A9	Address bits 0 to 3 and bit 9. These bits are used with -MODEM CS to select a register on the modem card.
-MODEM CS **DISKETTE CS**	This line is 'active' for addresses hex 0F0 thru 0FF and 3F8 thru 3FF. It is gated with A9 in the 8250A to exclusively decode hex 3F8 thru 3FF.

D0 thru D7	Data bits 0 thru 7: These eight lines form a bus through which all data is transferred. Bit 0 is the least significant bit (LSB).
-IOR	The content of the register addresses by line A0 thru A2 is gated onto lines D0 thru D7 when this line is 'active', -MODEM CS is 'active', and A9 is 'high.'
-IOW	The content of lines D0 thru S7 is stored in the register addressed by A0 thru A2 at the leading edge of this signal when -MODEM CS is 'active', and A9 is 'high.'
BAUDCLK	This is a 1.7895 MHz clock signal used to drive the Baud Rate Generator.
+MODEM INTR	This line is connected to the +IQRP4 on the 8259A Interrupt Controller.
-CARD INSTALL	This line indicates to the system BIOS that an IBM PC*jr* Internal Modem is installed in the feature location.

Telephone Company Interface

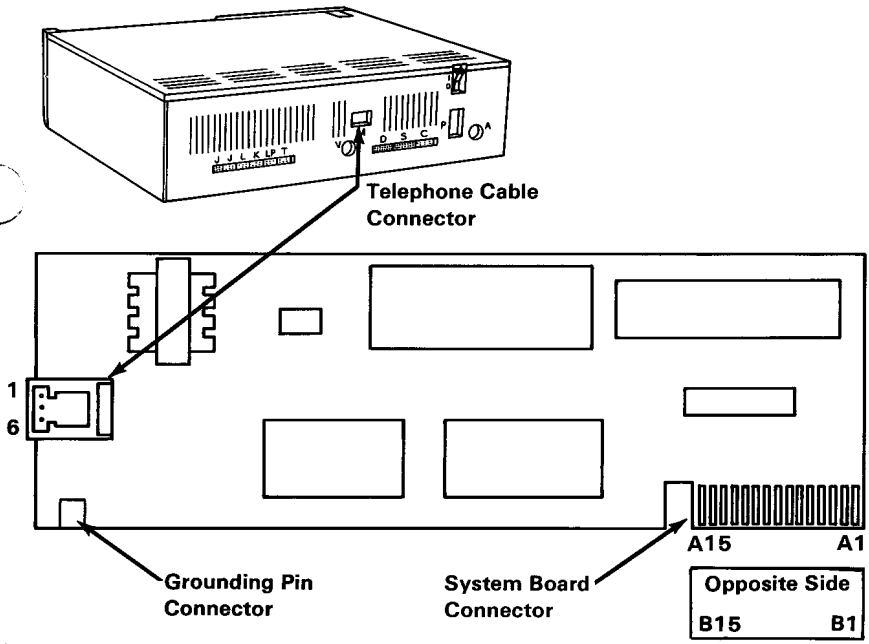
The telephone company interface is a 600 Ohm, balanced, two-wire telephone-interface design that meets the FCC Part 68 rules. A 2.13 meter (7 foot) modular telephone cord is included with the modem adapter.

Line-status detection of dial tone, ringback tone, busy, and incoming ring is provided along with automated routines which react to detected conditions.

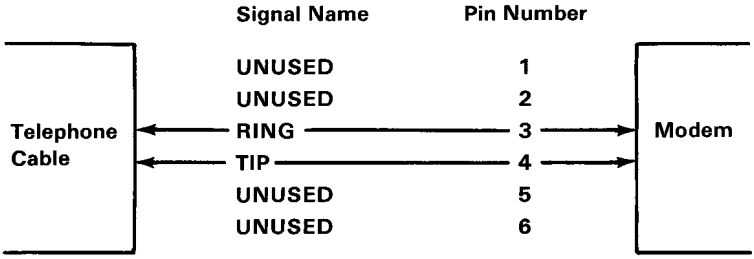
The modem card has one USOC RJ11 jack.

System I/O Channel

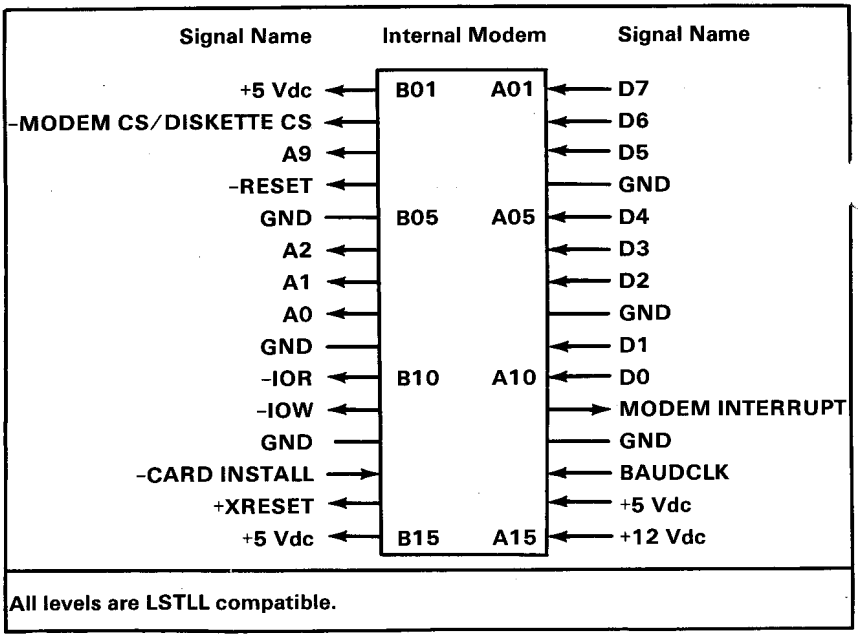
The following shows pin assignments for the system board modem connector. Pins A1 to A15 are on the component side.



Internal Modem Connectors



Connector Specifications (Part 1 of 2)



Connector Specifications (Part 2 of 2)

IBM PC*jr* Attachable Joystick

The Attachable Joystick is an input device intended to provide the user with two-dimensional positioning-control. Two pushbutton switches on the joystick give the user additional input capability.

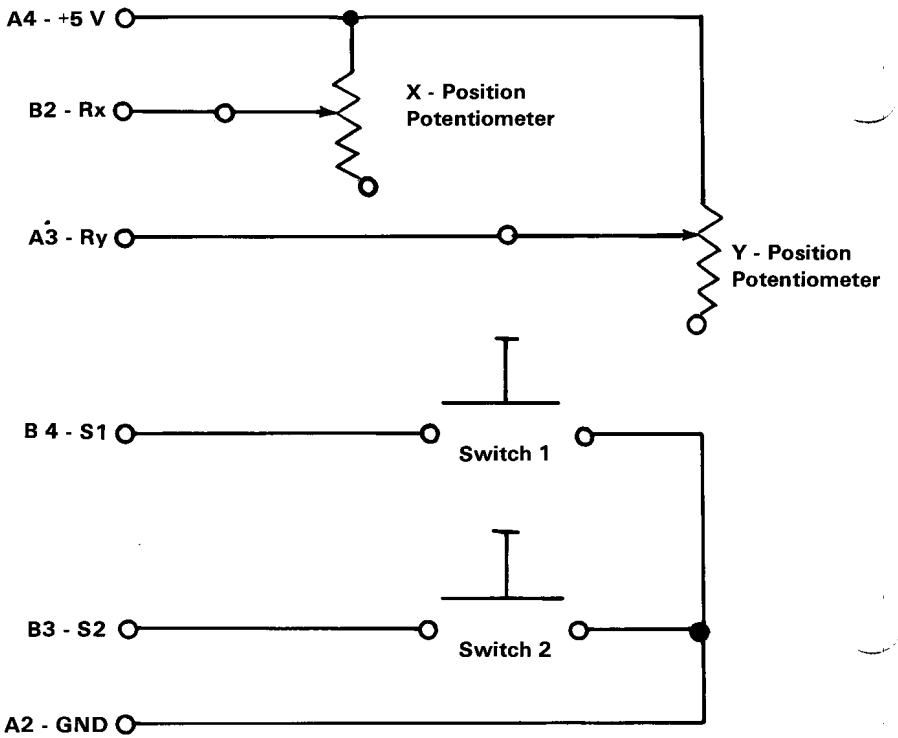
Hardware Description

Two modes of operation of the joystick are available. In the "Spring Return" mode the control stick returns to the center position when released. The "Free Floating" mode allows smooth, force free operation with the control stick remaining in position when released. Selection of these modes can be made for each axis independently. Two controls are provided for individual adjustment to the electrical center of each axis.

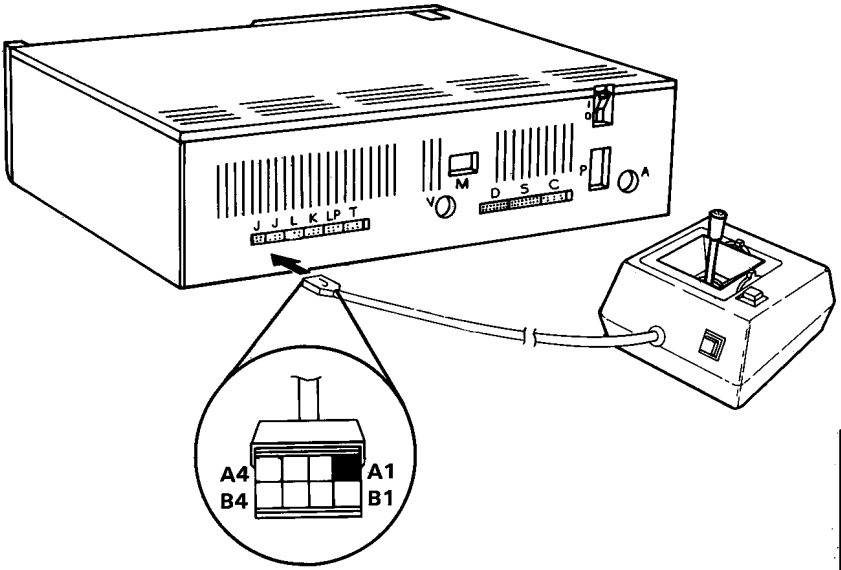
Functional Description

Positional information is derived from two potentiometers Rx and Ry. The resistance of these potentiometers will vary from 0 to 100K ohms nominally as the position of the control stick moves from left to right (X-axis) and from top to bottom (Y-axis). A linear taper is used on the potentiometers so that a linear relationship exists between angular displacement of the stick and the resulting resistance. Electrical centering for each axis is accomplished with the controls by mechanically rotating the body of the potentiometer. Adjustment in this manner has the effect of varying the minimum and maximum resistance relative to the extremes of the angular displacement. The two pushbuttons provided on the joystick are single-pole, single-throw, normally-open pushbuttons.

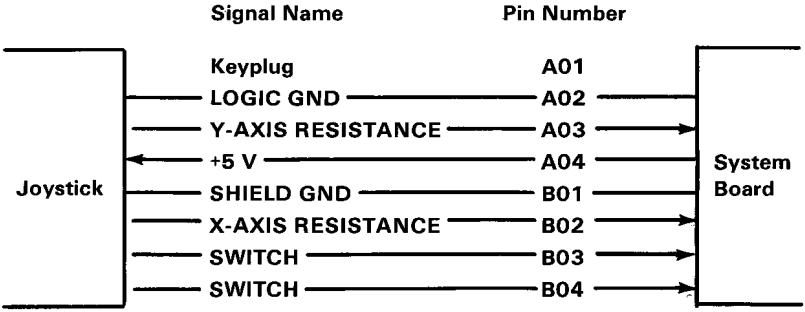
The following are the logic diagram and specifications for the two Attachable Joystick connectors.



Attachable Joystick Logic Diagram



Attachable Joystick Connector



Connector Specifications

Notes:

IBM Color Display

The IBM Color Display is a Red/Green/Blue/Intensity (RGBI)-Direct-Drive display, that is independently housed and powered.

Hardware Description

The IBM Color Display's signal cable is approximately 1.5 meters (5 feet) in length. This signal cable must be attached to the IBM PC*jr* with the IBM PC*jr* Adapter Cable for the IBM Color Display which provides a direct-drive connection from the IBM PC*jr*

A second cable provides ac power to the display from a standard wall outlet. The display has its own power control and indicator. The display will accept either 120-volt 60-Hz power or 220-volt 50-Hz power. The power supply in the display automatically switches to match the applied power.

The display has a 340 mm (13 in.) CRT. The CRT and analog circuits are packaged in an enclosure so the display may be placed separately from the system unit. Front panel controls and indicators include: Power-On control, Power-On indicator, Brightness and Contrast controls. Two additional rear-panel controls are the Vertical Hold and Vertical-Size controls.

Operating Characteristics

Screen

- High contrast (black) screen.
- Displays up to 16 colors.
- Characters defined in an 8-high by 8-wide matrix.

Video Signal

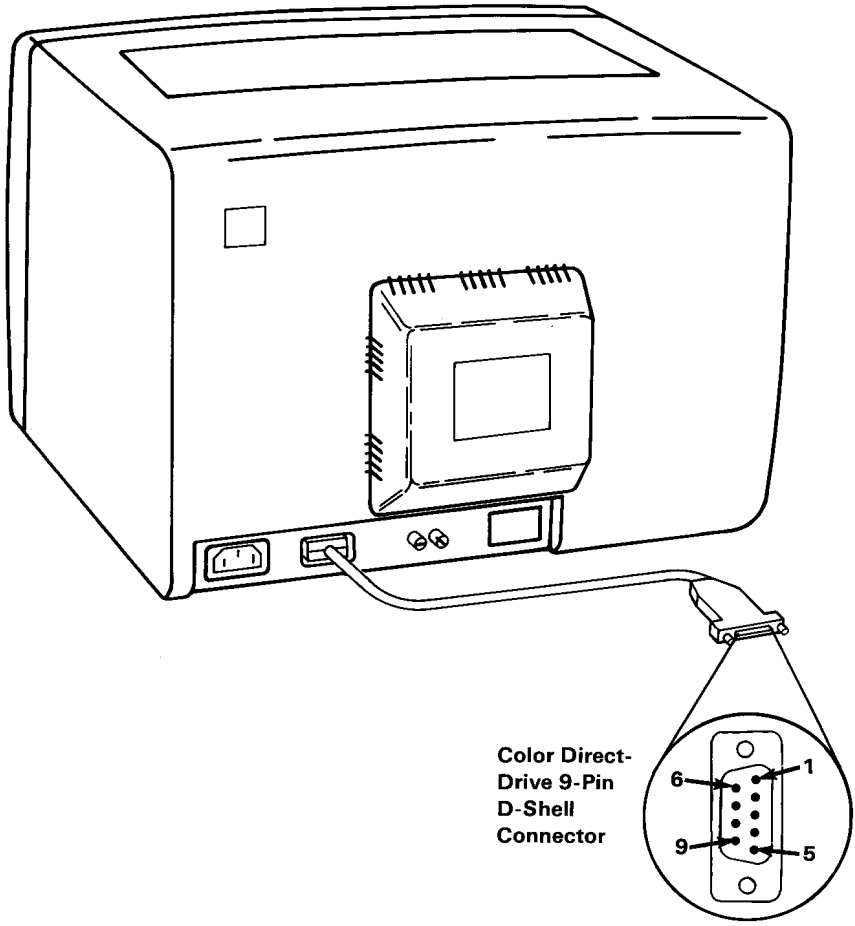
- Maximum video bandwidth of 14 MHz.
- Red, green, and blue video-signals, vertical sync, horizontial sync, and intensity are all independent. All input signals are TTL compatible.

Vertical Drive

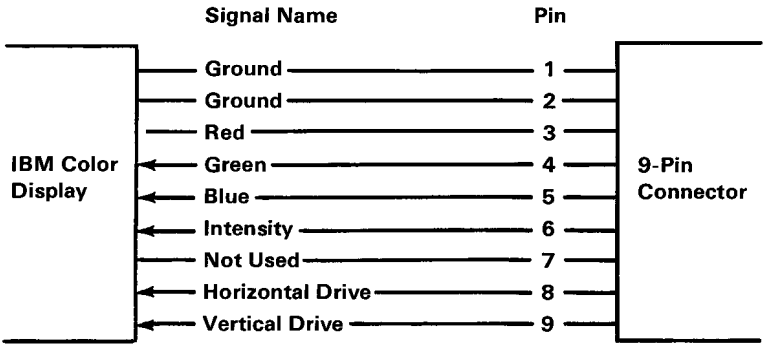
- Screen refreshed at 60 Hz with 200 vertical lines of resolution.

Horizontal Drive

- The horizontal drive frequency is 15.75 kHz.



Color-Display Connector



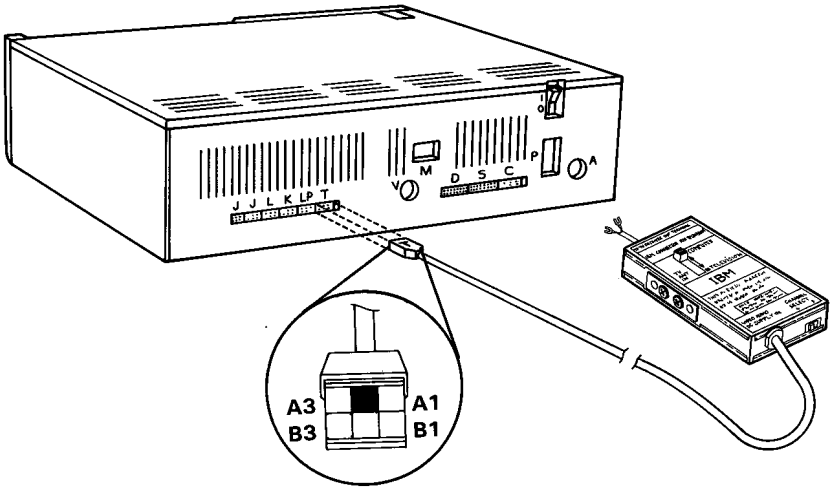
Connector Specifications

Notes:

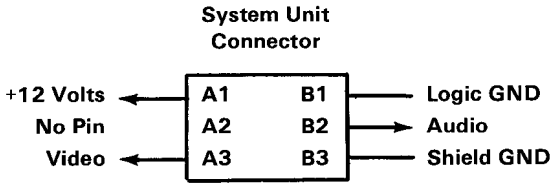
IBM Connector for Television

The Connector for Television is a sealed Radio Frequency (RF) Modulator that imposes the composite video and audio signals onto the RF carrier-wave supplied by the modulator. The connector unit has two two-position switches. One switch selects between the computer's signal or the standard-TV signal from an antenna as the input to the TV. The other switch selects either channel 3's or channel 4's carrier-wave frequency for input to the TV. This allows users to select the weaker TV channel for their area reducing the amount of interference with the computer's input signal. Signal input from the computer is provided by a five-conductor cable with a six-pin IBM PC*jr*-dedicated connector. Two spade-lug terminals provide for TV-antenna-cable connection. One twin-lead flat-type TV-cable provides input to the TV.

The following is the connector specifications for the IBM Connector for Television.



Connector for TV Connector



Connector Specifications

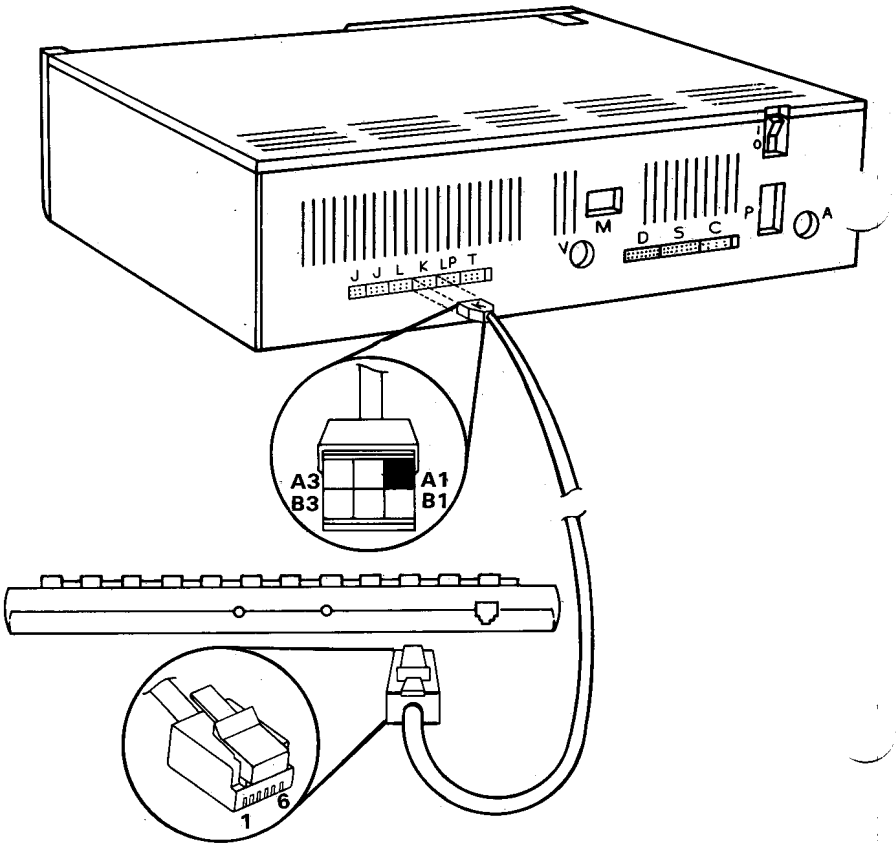
IBM PC*jr* Keyboard Cord

The IBM PC*jr* Cordless Keyboard can be attached to the PC*jr* using the optional Keyboard Cord. The Keyboard Cord is a 1.8 meter (6 foot), two twisted-pair cable, with a six-position RJ11-type connector for the keyboard and a six-position Berg-type connector for the system unit.

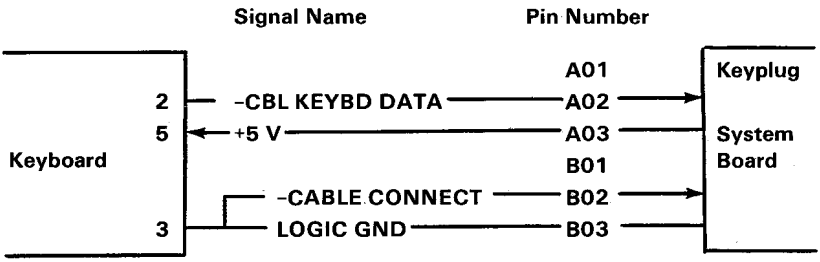
The Keyboard Cord option should be used in an environment that is unfavorable for use of the infra-red link. For instance, brightly lit high-intensity light areas, or multiple IBM PC*jr* areas where keyboards can conflict with one another.

Insertion of the cord's keyboard connector into the keyboard actuates switches internal to the keyboard. The switches 'deactivate' the IR transmitter by removing the power supplied by the keyboard's batteries. The system unit's infra-red (IR) receiver circuit is 'disabled' by the -CABLE CONNECT signal, supplied when the system-unit end of the cord is connected.

The following figures show the connector specifications for the Keyboard Cord.



Keyboard Cord Connectors

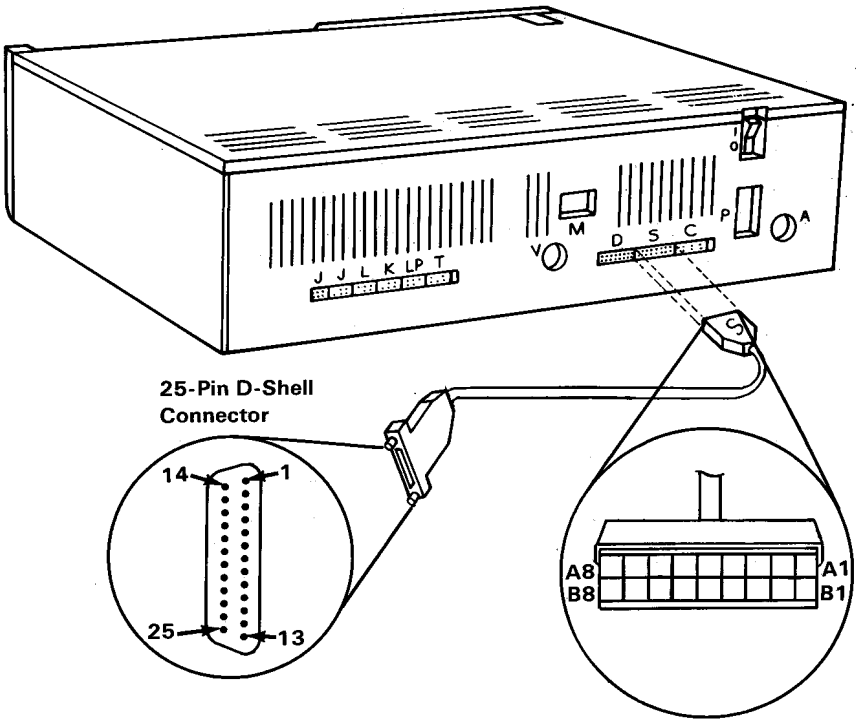


Connector Specifications

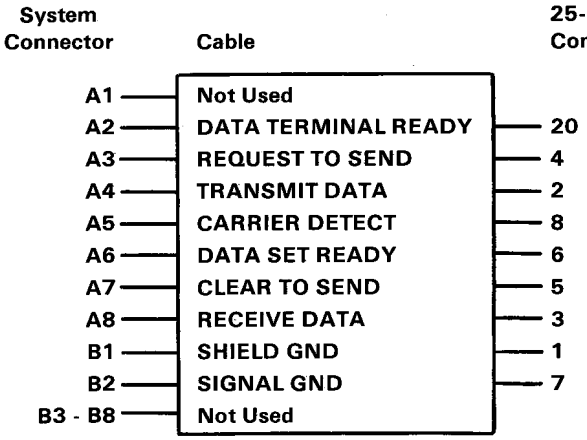
IBM PC*jr* Adapter Cable for Serial Devices

The Adapter Cable for Serial Devices is a 72 mm (3-inch) long, nine-conductor cable terminated with a 16-position Berg-type connector and a 25-pin "D"-shell connector. This cable allows serial devices that terminate with a standard EIA-RS232C 25-pin "D"-shell connector to be connected to the IBM PC*jr*.

The following figures show the connector specifications for the Adapter Cable for Serial Devices.



Adapter Cable for Serial Devices



Connector Specifications

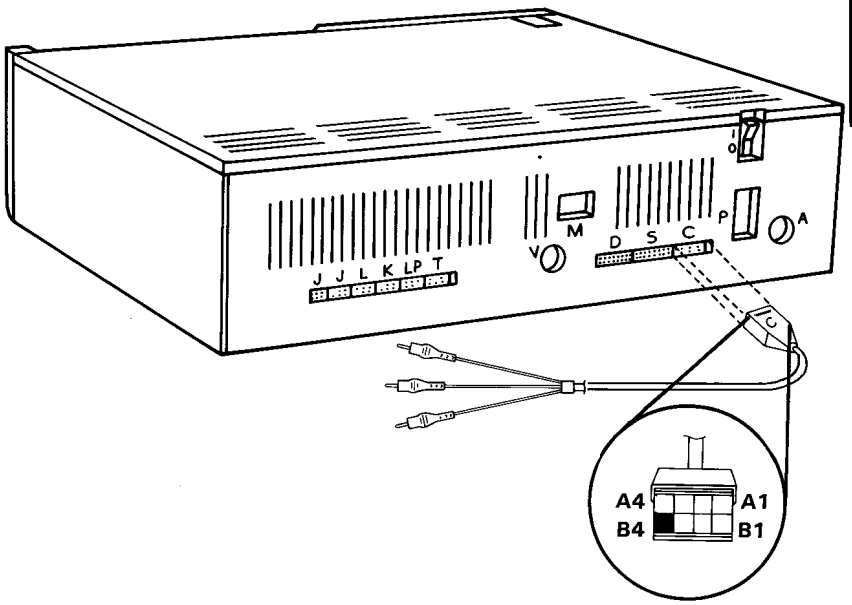
IBM PC*jr* Adapter Cable for Cassette

This option is an adapter cable that allows connection of a cassette recorder to the IBM PC*jr* cassette connector.

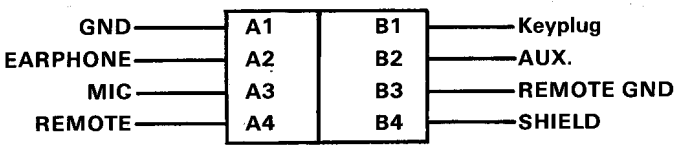
The cassette recorder to be connected must use the following type connectors:

- Belden Style-51 miniture phone-plug (Auxiliary)
- Belden Style-51 miniture phone-plug (Earphone)
- Belden Style-56 subminiture phone-plug (Remote)

The following figures show the connector specifications for the Adapter Cable for Cassette.



Adapter Cable for Cassette Connectors



Connector Specifications (System End)
(Part 1 of 2)

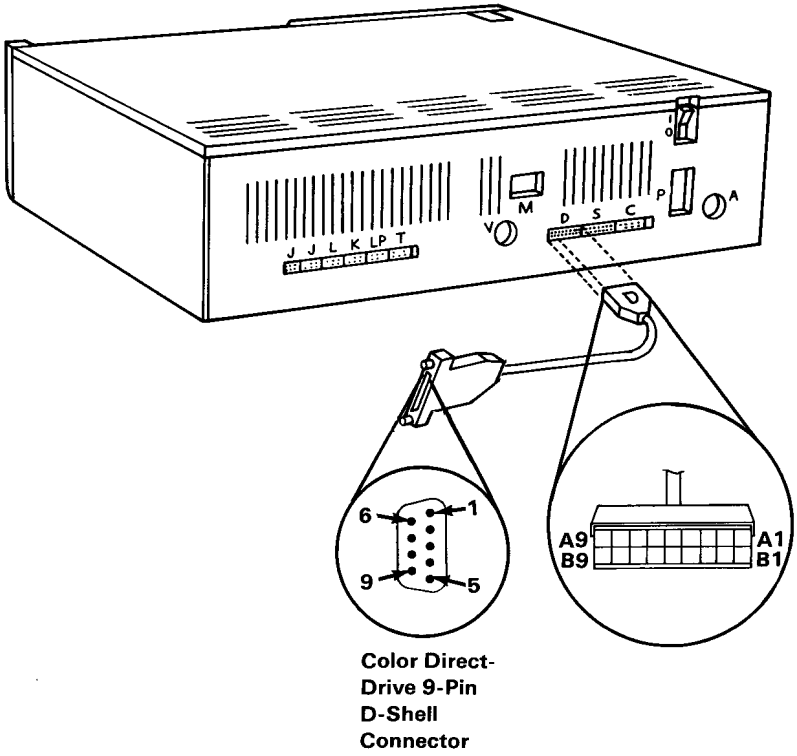
Cassette Connector		System Connector Pin
Aux. (Red)	Signal	B2
	Gnd	A1
Ear (Black)	Signal	A2
	Gnd	A1
Remote (Gray)	Signal	A4
	Gnd	B3

Connector Specifications (Recorder End)
(Part 2 of 2)

IBM PC*jr* Adapter Cable for the IBM Color Display

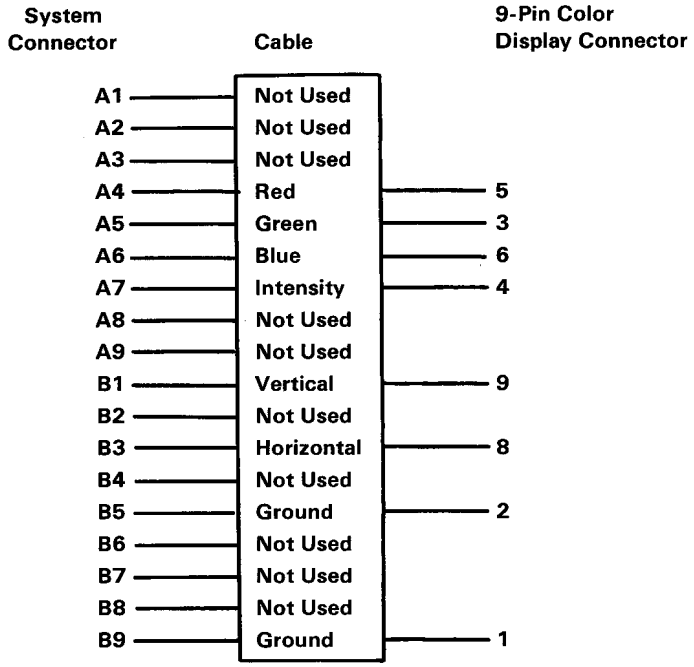
This adapter cable allows the IBM Color Display to be connected to the IBM PC*jr*.

The following figures show the connector specifications for the adapter cable for the IBM Color Display.



Color Direct-Drive 9-Pin D-Shell Connector

Adapter Cable for IBM Color Display Connectors



Connector Specifications

IBM PC*jr* Parallel Printer Attachment

The Parallel Printer Attachment is provided to attach various I/O devices that accept eight bits of parallel data at standard TTL-logic levels. The card measures 76mm (3 inches) high by 244mm (9.6 inches) long.

The Parallel Printer Attachment attaches as a feature to the right-hand side of the system unit. It connects to the 60-pin Input/Output (I/O) connector where power and system-input signals are received. A parallel printer attaches to the Parallel Printer Attachment through a 25-pin female "D"-shell connector located on the rear edge of the attachment, where a cable and shield can be attached. The logic design is compatible with the IBM Personal Computer printer adapter.

The attachment card has 12 TTL buffer-output points which are latched and can be 'written' and 'read' under program control using the processor 'IN' or 'Out' instructions. The attachment card also has five steady-state input-points that may be 'read' using the processors' 'IN' instructions.

In addition, one input can also be used to create a processor interrupt. This interrupt can be 'enabled' and 'disabled' under program control. 'Reset' from the power-on circuit is also **ORed** with a program-output point allowing a device to receive a power-on 'reset' when the processor is 'reset.'

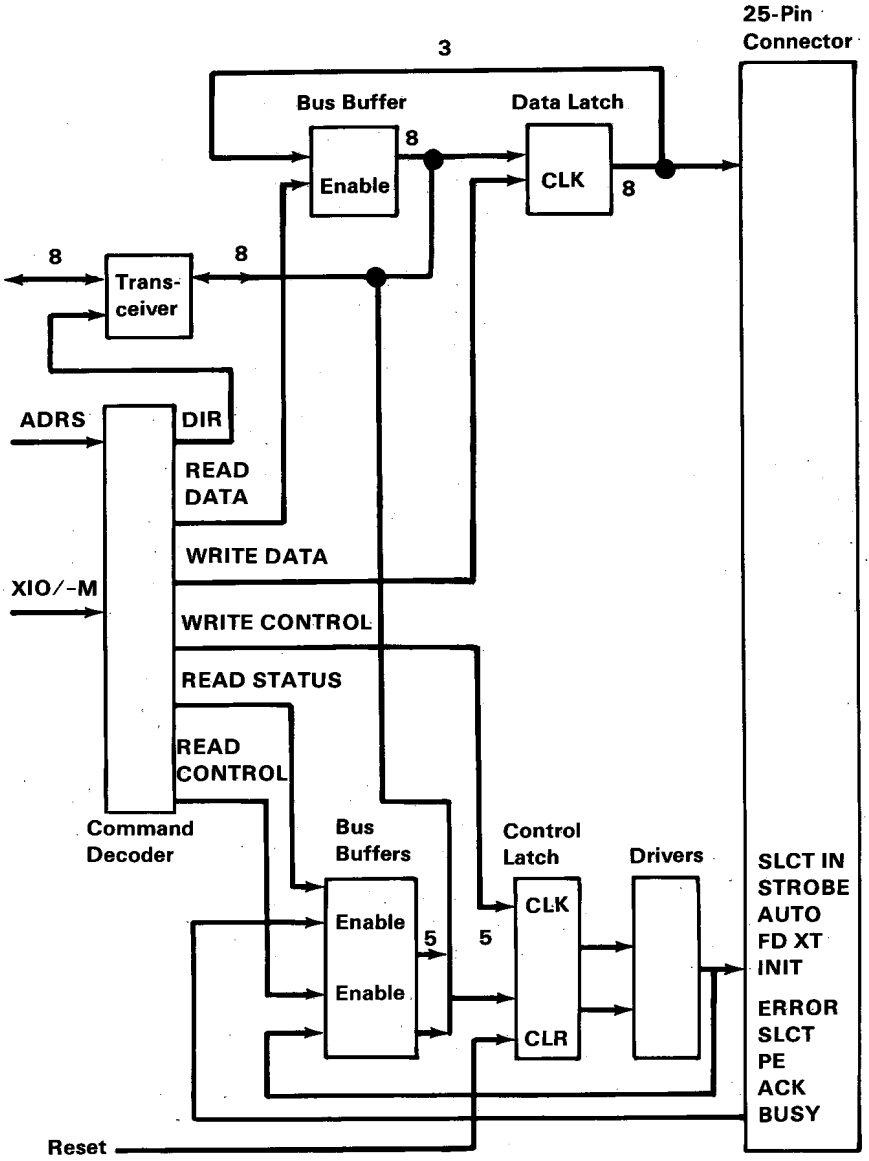
When the Parallel Printer Attachment is used to attach a printer, data or printer commands are loaded into an 8-bit latched output-port, then the strobe line is 'activated' to 'write' data to the printer. The program can then 'read' the input ports for printer

status indicating when the next character can be written or it may use the interrupt line to indicate **not busy** to the software.

The output ports can also be 'read' at the card's interface for diagnostic-loop functions. This allows fault-isolation determination between the printer attachment and the attached printer.

Description

During a system I/O 'read' or 'write', with the proper address selection, data may be 'written' to or 'read' from the Parallel Printer Attachment. The data and Control Registers must be manipulated by the system software to be consistent with the attaching hardware. The following is a block diagram of the Parallel Printer Attachment card.



Parallel Printer Interface Block Diagram

System Interface

The Parallel Printer Attachment reserves addresses hex 378, through hex 37F. **IO/-M** must also be 'active high' when addressing the Parallel Printer Attachment.

A card selected signal (**-CARD SLCTD**) is provided to the system I/O when the above addresses are used, and the **IO/-M** bit is 'active high.'

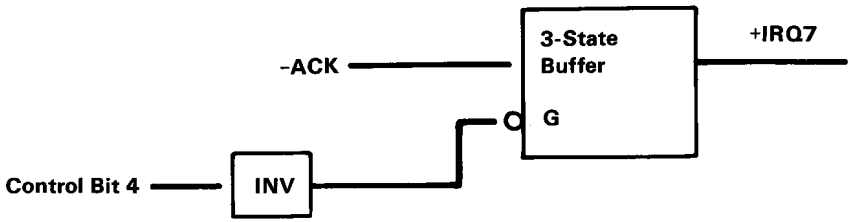
Specific commands are decoded from A0, A1, RD, and WR per the following table. Input A2 is not used.

Addresses (hex)	Operation	Comments
378	'Read'	Read Data Latch
379	'Read'	Read Status
37A	'Read'	Read Control Latch
37B	'Read'	Unused
37B	'Write'	Write Data Latch
379	'Write'	Unused
37A	'Write'	Write Control Latch
37B	'Write'	Unused

All data transfers take place over the 8-bit I/O data-bus with timing provided by the 8088 microprocessor. (IOR, IOW, IO/-M)

An interrupt is provided to the system through the I/O connector of the Parallel Printer Attachment. This

interrupt is 'positive active', Interrupt Level 7 (+IRQ7). Bit 4 of the control latch must be 'written high' to allow interrupts. When the -ACKnowledge signal ('low active' signal goes 'high') the I/O device causes a level 7 interrupt. See the following figure.



+IQR7/-ACK Logic Diagram

Programming Considerations

The Parallel Printer Attachment can serve as a general purpose peripherial driver. This section describes a configuration which supports attachment to the IBM Graphics Printer.

Command Definition

For the parallel-printer application, the following bit definitions apply.

Data Latch - Address hex 378

A 'write' to this address causes data to be latched onto the printer data bits. A 'read' from this address presents the contents of the data latch to the processor.

MSB	7	6	5	4	3	2	1	0	LSB
	Data	Data	Data	Data	Data	Data	Data	Data	
	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0	

Data Latch Format

Printer Status - Address hex 379, hex 7D, Input Only

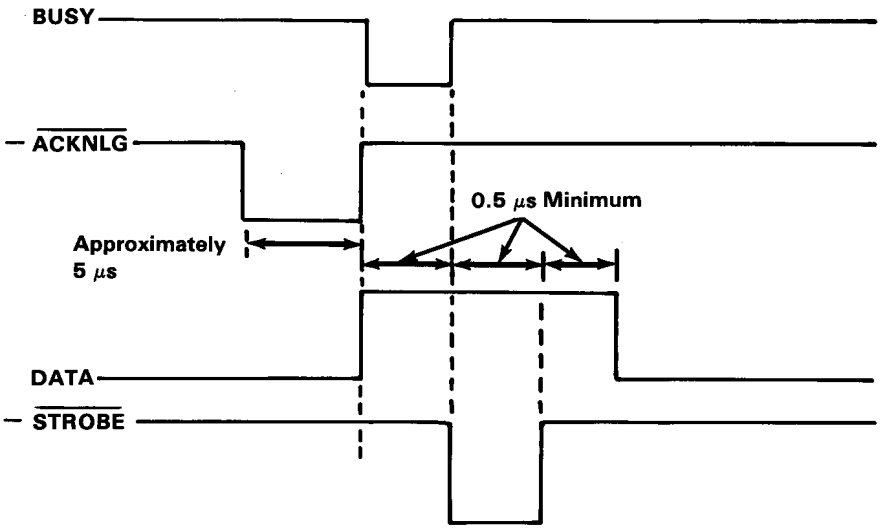
This port provides real-time feedback and status to the system from the printer.

Bit	Signal Name	Description
MSB 7	–BUSY	When this signal is at a low level, the printer is busy and cannot accept data. It can become low during data entry, off-line printing, head translation, or error state.
6	–ACK	When port B is read, this bit will represent the current state of the printer ACK signal. A low level means that a character has been received and the printer is ready to accept another. Normally, this signal will be low for approximately 5 microseconds before BUSY goes away.
5	–PE	A low level indicates that the printer has detected an end of form.
4	+SLCT	A high level indicates that the printer is selected.
3	–ERROR	A low level indicates that the printer has encounted an error condition.
2 Through 0 LSB		Unused.

Printer Status

Printer Control - Address hex 37A

This port contains printer control signals. A 'write' latches control bits to the printer; a 'read' presents the contents of the latches to the processor. See the following timing diagram:



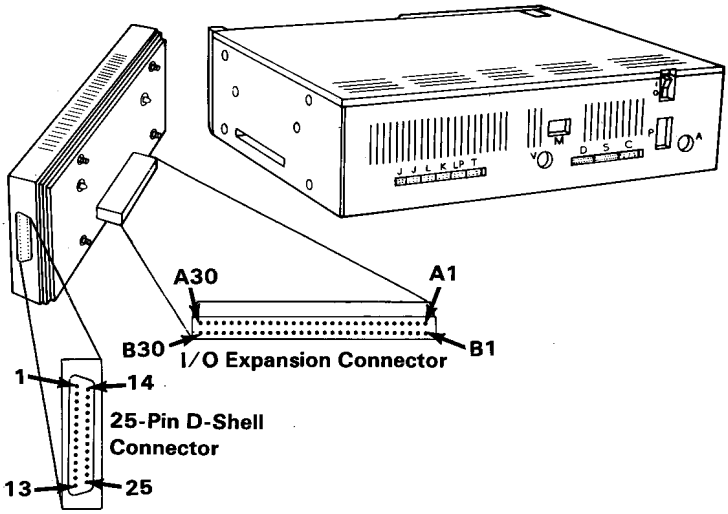
Parallel Interface Timing Diagram

The following figure describes the printer control signals.

Bit	Signal Name	Description
MSB 7 Through 5		Unused.
4	+INTERRUPT ENABLE	A high level in this bit position will allow an interrupt to occur when –ACK goes high.
3	SLCT IN	A low level in this bit position selects the printer.
2	INIT	A low level will initialize the printer (50 microseconds minimum).
1	AUTO FD XT	A low level will cause the printer to line feed anytime a line is printed.
LSB 0	STROBE	A 5 microsecond (minimum) low active pulse clocks data into the printer. Valid data must be present for 5 microseconds (minimum) before and after the STROBE pulse.

Printer Control Signal

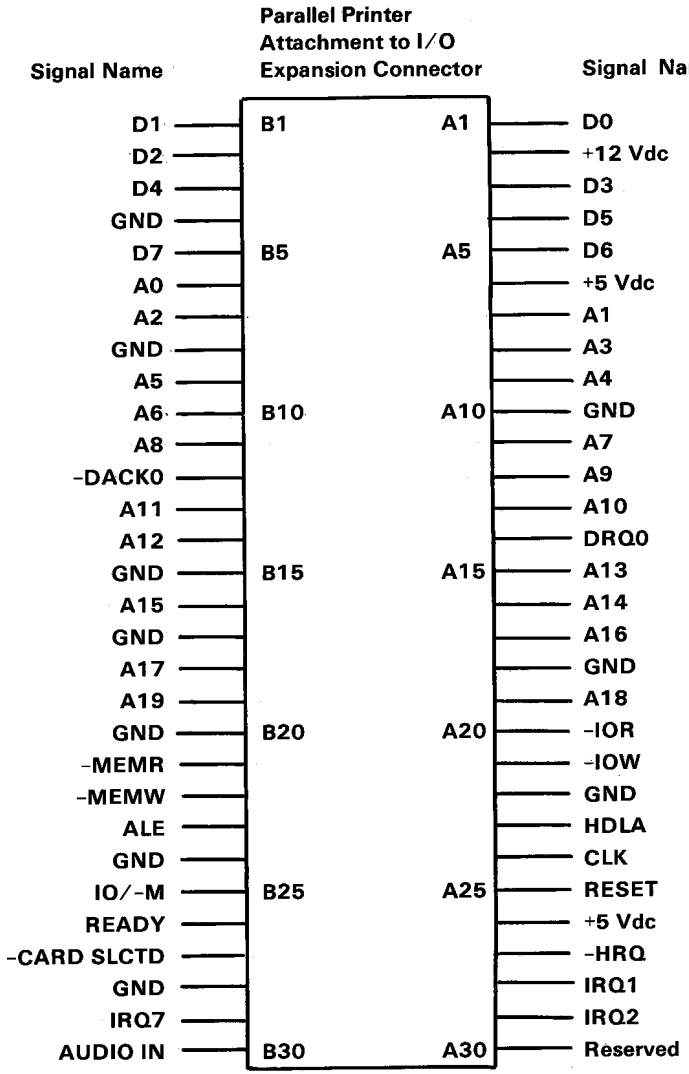
The following are the connector specifications for the IBM PC*jr* Parallel Printer Attachment.



Parallel Printer Attachment Connectors

25-Pin "D"-Shell Connector				
Pin	**Signal**	**I_{OL} Max**	**I_{OH} Max**	**Source**
1	–STROBE	14 ma	–.6 ma	Attachment Card
2 Through 9	DATA BIT 0 Through DATA BIT 7	24 ma	–2.6 ma	Attachment Card
10	–ACK	74LS Input	74LS Input	Printer
11	BUSY	74LS Input	74LS Input	Printer
12	PE	74LS Input	74LS Input	Printer
13	SLCT	74LS Input	74LS Input	Printer
14	–AUTO FD XT	14 ma	.6 ma	Attachment Card
15	–ERROR	74LS Input	74LS Input	Printer
16	–INIT PRINTER	14 ma	.6 ma	Printer
17	–SELECT INPUT	14 ma	.6 ma	Attachment Card
18 Through 25	GND	N/A	N/A	

Connector Specifications (Part 1 of 2)



Connector Specifications (Part 2 of 2)

Notes:

IBM Graphics Printer

The IBM Graphics Printer is a self-powered, stand-alone, tabletop unit which attaches to the system unit through a 6-foot parallel-signal cable, and obtains 120 Vac power from a standard wall outlet through a seperate cable. It is an 80 CPS (characters per second), bidirectional, wire-matrix device that can print in a compressed mode of 132 characters per line, in a standard mode of 80 characters per line, in a double width-compressed mode of 66 characters per line, and in a double width mode of 40 characters per line. It can also print double-size and double-strike characters. It prints the standard ASCII, 96-character, uppercase and lowercase character sets and also has a set of 64 special block characters. It has an extended character set for international languages, subscript, superscript, an underline mode, and programmable graphics. The Graphics printer accepts commands that set the line-feed control desired for the application.

It attaches to the system unit through the IBM PC*jr* Parallel Printer Attachment. The cable is a 25-conducter, shielded cable with a 25-pin "D"-shell connector at the system unit end, and a 36-pin connector at the printer end.

Printer Specifications

Print Method: Serial-impact dot matrix

Print Speed: 80 CPS

Print Direction: Bidirectional with logic seeking

Number of Pins in Head: 9

Line Spacing: 1/16 inch (4.23 mm) or programmable

Matrix Characteristics: 9 by 9

Character Set: Full 96-character ASCII with descenders plus 9 international characters/symbols

Graphic Characters: See "Additional Printer Specifications"

Printing Sizes:

Normal	10 characters-per-inch with a maximum of 80 characters-per-line
Double Width	5 characters-per-inch with a maximum of 40 characters per line
Compressed	16.5 characters-per-inch with a maximum of 132 characters per line
Double Width-Compressed	8.25 characters-per-inch with a maximum of 66 characters per line
Subscript	10 characters-per-inch with a maximum of 80 characters per line
Superscript	10 characters-per-inch with a maximum of 80 characters per line

Media Handling: Adjustable sprocket-pin-feed with 4-inch (101.6 mm) to 10-inch (254 mm) width paper, one original plus two carbon copies (total thickness not to exceed 0.012 inch (0.3 mm)), minimum paper thickness of 0.0025 inch (0.064 mm)

Interface: Parallel 8-bit data and control lines

Inked Ribbon: Black, cartridge type with a life expectancy of 3 million characters

Environmental Conditions: Operating temperature is 5 to 35 degrees centigrade (41 to 95 degrees Fahrenheit), operating humidity is 10 to 80% non-condensing

Power Requirements: 120 Vac, 60 Hz, 1 A maximum with a power consumption of 100 VA maximum

Physical Characteristics:

Height	107 mm (4.2 inches)
Width	374 mm (14.7 inches
Depth	305 mm (12 inches)
Weight	5.5 kg (12 pounds)

Additional Printer Specifications

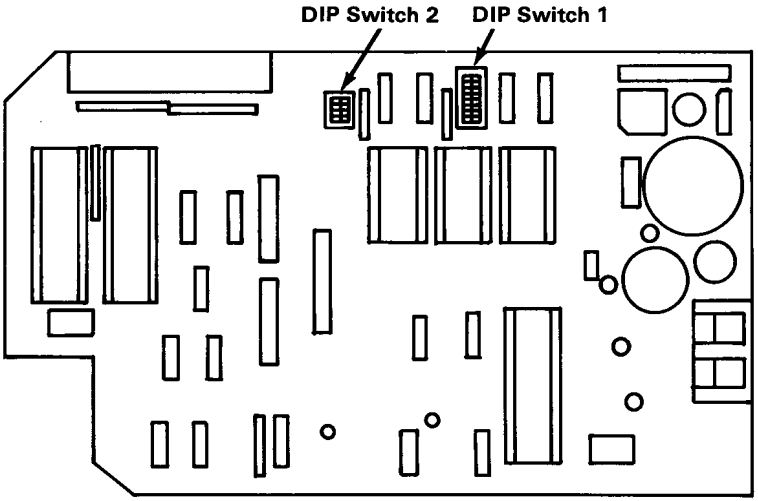
Printing Characteristics

Extra Character Set

Set 1	Additional ASCII numbers 160 to 175 contain European characters. Numbers 176 to 223 contain graphic characters. Numbers 224 to 239 contain selected Greek-characters. Numbers 240 to 255 contain math and extra symbols.
Set 2	The differences in Set 2 are ASCII numbers 3,4,5,6, and 21. ASCII numbers 128 to 175 contain European characters.
Graphics	There are 20 block characters and programmable graphics.

DIP Switch Settings

There are two Dual-Inline-Package (DIP) switches on the control circuit-board. In order to satisfy the user's specific requirements, desired control modes are selected by the DIP switches. The functions of these switches and their preset conditions at the time of shipment are shown in the following figures.



Location of DIP Switches

Switch Number	Function	On	Off	Factory Position
1-1	Not Applicable	—	—	On
1-2	CR	Print Only	Print and Line Feed	On
1-3	Buffer Full	Print Only	Print and Line Feed	Off
1-4	Cancel Code	Invalid	Valid	Off
1-5	Not Applicable	—	—	On
1-6	Error Buzzer	Sound	No Sound	On
1-7	Character Generator	Set 2	Set 1	Off
1-8	SLCT IN Signal	Fixed Internally	Not Fixed Internally	On

Functions and Conditions of DIP Switch 1

Switch Number	Function	On	Off	Factory Position
2-1	Form Length	12 Inches	11 Inches	Off
2-2	Line Spacing	1/8 Inch	1/6 Inch	Off
2-3	Auto Feed XT Signal	Fixed Internally	Not Fixed Internally	Off
2-4	1 Inch Skip Over Perforation	Valid	Invalid	Off

Functions and Conditions of DIP Switch 2

Parallel Interface Description

Specifications

Data Transfer Rate	1000 cycles-per-second (cps)-(maximum)
Synchronization	By externally-supplied STROBE pulses
Signal Exchange	-ACKNLG or BUSY signals
Logic level	Input data and all interface-control signals are compatible with the Transistor-Transistor Logic (TTL) level.
Connector	Plug 57-30360 (Amphenol)

Connector-pin assignments and descriptions of respective interface-signals are provided in the following figures.

Data Transfer Sequence

The following figure shows the Parallel Interface Timing.