February 1983

This document is a technical reference manual for the PCP-11 Programmable Communications Processors. Hardware features and configuration options are presented in sufficient detail so that the user may install and program these devices. Detailed operating specifications for the programmable elements of the PCP-11 are contained in the appropriate manufacturer's data sheets. This manual should be used in conjunction with operating system and applications software documentation.

PCP-11

TECHNICAL REFERENCE MANUAL

Version 1.2

Infosphere, Inc 4730 SW Macadam Portland, OR 97201

WARRANTY

All Infosphere hardware products are warranted against defective materials and workmanship for a period of one year. This warranty is limited to repair or replacement by Infosphere upon return of the product by the customer, freight prepaid, to Infosphere's factory or its duly authorized agent. Infosphere shall, in no event, be responsible for defects resulting from modification or repairs to the product by persons other than Infosphere's agents.

It is Infosphere's policy to improve products as new techniques and components become available. Infosphere therefore reserves the right to change specifications and pricing at any time.

PCP-11 Technical Manual

Table Of Contents

PCP-11 Technical Manual

Table of Figures

Chapter 1

Introduction

1.1 PCP-11 Features

The members of the PCP-11 family are single board peripheral processors intended for use with the DEC LSI-11, PDP-11/03, and PDP-11/23 family of mini-computers. The PCP-11S is an extended configuration designed specifically for developing serial communications applications. Contained upon a 'dual height' circuit board, the PCP-11 provides:

* A programmable 4 MHz Z80A microprocessor (MPU);

* 16 Kbytes of dynamic RAM for down-loadable control store and data buffering;

* 3 EPROM sockets, capable of supporting single supply JEDEC byte-wide memories (a mix of 2716s, 2732s, or 2764s, up to 24 Kbytes) for non-volatile control programs and data constants. The two higher order sockets will also accomodate static RAMs.

* A bidirectional 16 bit I/O port-mapped interface to the host LSI-11 computer, with interrupt driven handshaking. Data transfers to and from the PCP-11 are performed on a programmed I/O basis; the PCP-11 does not support DMA data transfers;

* Dual serial channels, with RS-232 compatible RS-423 signal levels. One channel supports full modem control, with a 25 pin connector, and asynch or synch operating modes. The second channel supports asynch transmit and receive, and utilizes a standard 10 pin DLV-11J type connector. A 'pregnant' cable sub-assembly externally performs level shifting and local signal loopback for the serial channels;

* A four channel counter/timer circuit, usable as a programmable baud rate generator (max rate 9.6K x16) and/or real time clock;

* An 8 bit bidirectional parallel expansion port, typically used to implement auto-dialing features;

* An 8 bit I/O port mapped DIP switch register typically used for end-user configuration of applications software options;

* On the PCP-11S, an EPROM-based resident operating system and programming language called PCP-Sphere. PCP-Sphere provides a proprietary stack-oriented operating environment that gives the user the interactive facilities of an extensible high level interpretive compiler, an optional resident assembler, and the basic utilities necessary to download and debug pre-compiled object modules and to execute pre-compiled applications modules;

* A host resident support program that allows the host to emulate the 'console' of the PCP-11S. This program simplifies the downloading and debugging of programs by supporting interactive communications and file transfers with the PCP-11S.

The PCP-11S may be augmented by various options which include:

* A PCP-Sphere extension set, including resident interactive assembler and 'console' redirection capabilities, contained in EPROM.

1.2 PCP-11 Design Benefits

The PCP-11 provides significant communications augmentation capability to LSI-11 based systems. Operating as a loosely coupled slave co-processor, protocol management and data buffering functions may be off loaded from the host, increasing throughput and reducing memory demands. This seperation allows the system designer to treat a communications link as a logical entity, without becoming mired in the details of the protocol or operating system interface.

Because applications programs may be down-loaded from the host LSI-11, they can be changed 'on the fly', or conveniently updated or upgraded in the field. A PCP-11S implementation can evolve in protocol complexity without changing the characteristics of the host environment. Functional migration of control software from host to a PCP-11S allows long term optimization of system resource usage.

The PCP-11's serial level shifting circuitry is contained in an external pod which allows custom applications to select the interfacing characteristics that most closely match their needs. Such custom applications could include: current loop, SDLC loop, multi-drop, RS-422, fiber optic, or coaxial transmission methods.

By supporting an auxiliary serial channel, the PCP-11 allows the user to implement functions such as:

* line monitoring of protocol activity on a slave CRT;

* interfacing a TU-58 cassette tape drive for program loading or data logging;

* synchronizing a dual PCP-11 implementation; or

* even replacing the standard host system console serial interface.

With the PCP-11S, the resident PCP-Sphere kernel provides a complete operating system and programming language for diagnostics, hardware familiarization, and even applications software development and debugging. The basic facilities of the kernel can be augmented by down loading additional software from the host, or by installing PCP-Sphere extension ROMs.

Upon power-up, PCP-Sphere automatically links extension ROMs into its dictionary (a combination of a symbol table and an executable image) and will execute a user-specified sequence for appropriate initialization of the extended dictionary. The kernel even generates ROMable code suitable for EPROM programming with user-supplied equipment.

Alternatively, via DIP switch option, the PCP-11S can be configured to automatically execute a user-generated resident application program, or to auto-load and execute a host supplied program.

Chapter 2

Installation and Configuration

WARNING

The equipment described in this manual generates, uses, and can radiate radio frequency energy and if not installed and used in accordance with the instructions manual, may cause interference to radio communications. As temporarily permitted by regulation it has not been tested for compliance with the limits for Class A computing devices pursuant to Subpart J of Part 15 of FCC Rules, which are designed to provide reasonable protection against such interference. Operation of this equipment in a residential area may cause unacceptable interference to radio frequency communications, in which case the user, at his own expense, will be required to take whatever measures may be required to correct the interference.

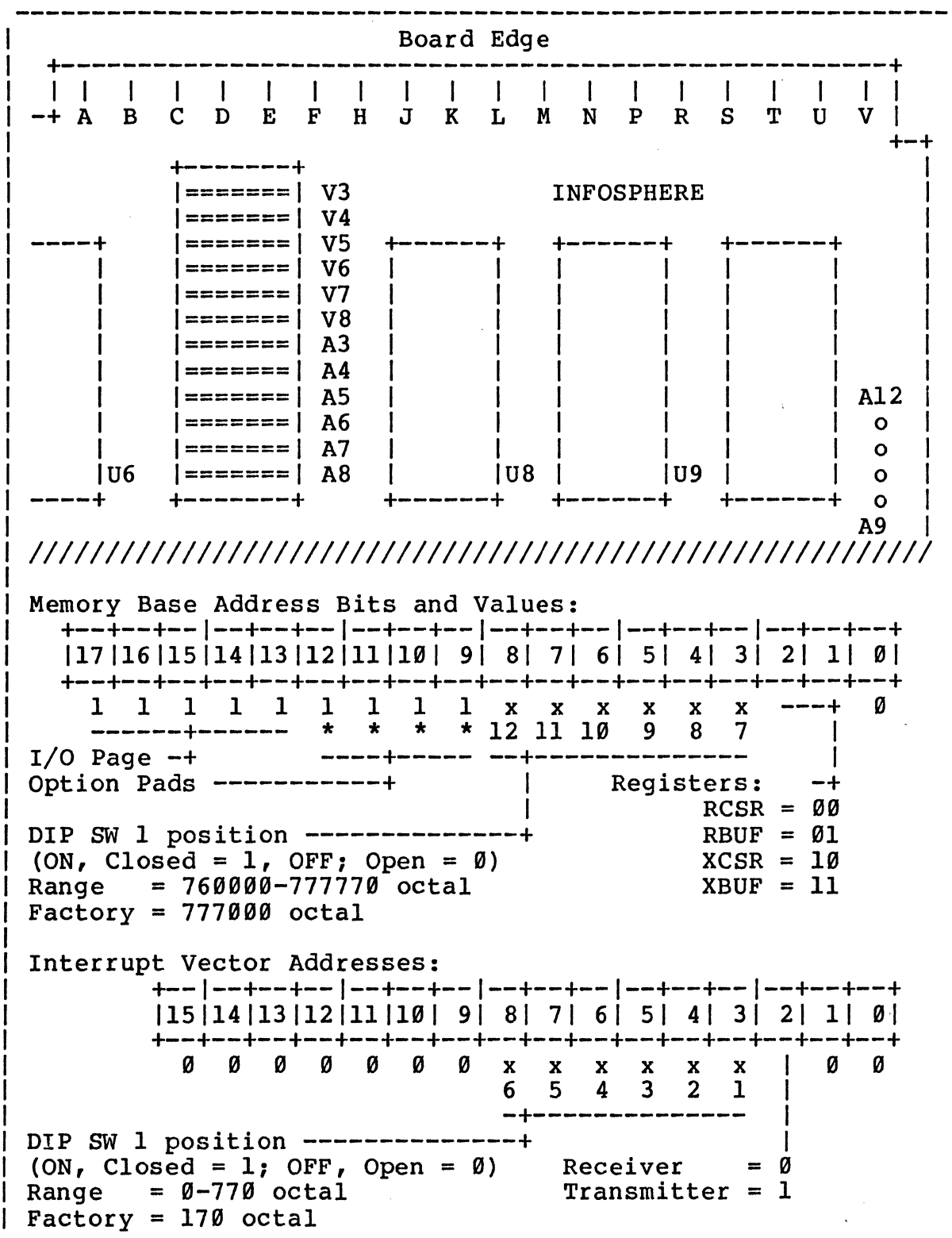
The equipment described in this manual has not been tested to show compliance with new FCC rules (47 CFR Part 15) designed to limit interference to radio and TV reception.

The PCP-11 may be installed in any dual width slot within the system backplane. Due to the 'daisy chained' interrupt grant structure of the LSI-11 bus, the interrupt priority of the module will be determined by its electrical proximity to the CPU board. The data buffering typically performed by the PCP-11 usually eliminates any sensitivity to its interrupt priority.

Damage may result from inserting or removing modules from an LSI-11 backplane with power applied.

2.1 Host Memory Addresses and Interrupt Vectors

Figure 2-1 -- Memory and Vector Addresses



The normal address range for the PCP-11 is 777000 through 777770 octal, with 18 bit addressing, or 177000 through 177770 octal, with 16 bit addressing. Bits 3 through 8 may be set via DIP switch one, as illustrated in Figure 2-1. By cutting any of the traces that lead to the four option pads next to U10, the respective address select line will be active low (logically), allowing the lower address limit to be decreased from 777000 (177000) to 760000 (160000) octal. The factory setting is 777000 (177000) octal. Bits 1 and 2 address the desired register in the module.

The interrupt vectors for the module are also selected via DIP switch 1. 64 possible locations are supported, in the range of 0 to 770 octal, in steps of 10 octal. The factory setting is 170 octal.

2.2 Memory Options

The PCP-11 family supports a wide range of industry compatible 'byte-wide' memories. The differences between these memories are accomodated via jumpers located under each of the three memory sockets. The following table lists the configuration choices for each of the memory types supported. (Note that '--' indicates the signal is the same for all types.)

Active low signals are indicated with a leading asterisk (i.e. *WR).

Figure 2-2 -- Memory Device Options

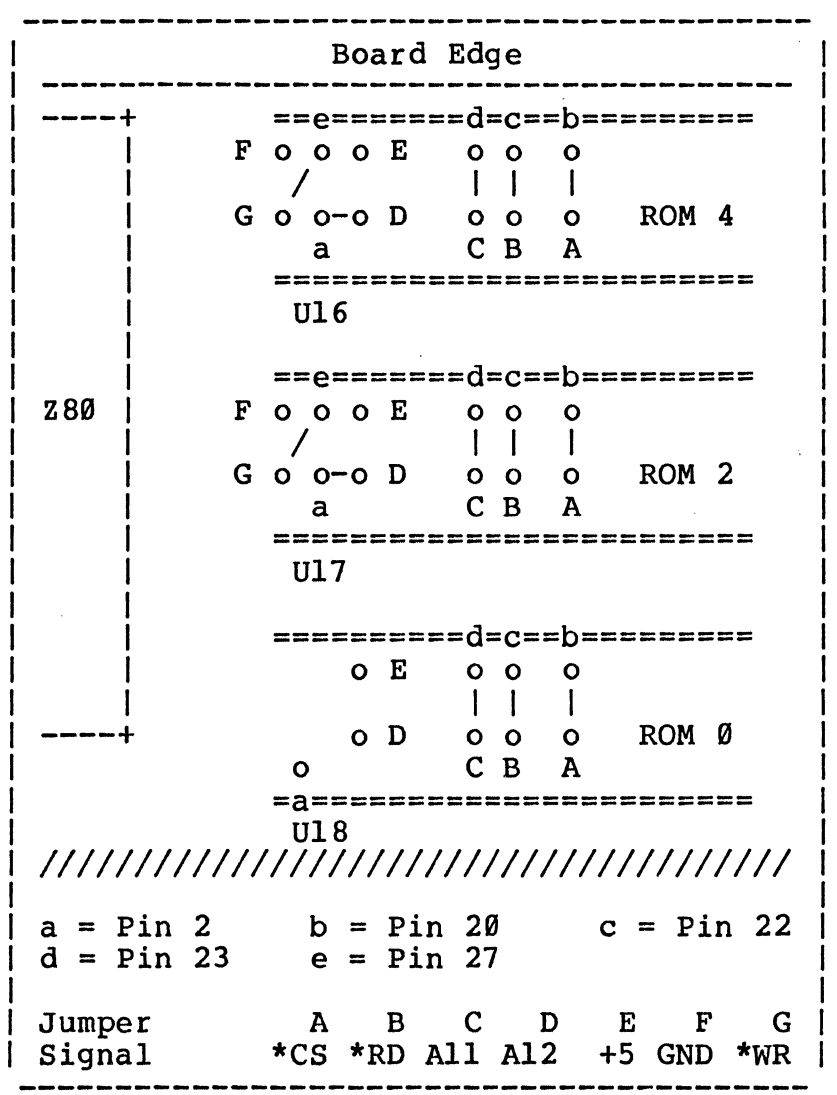
2764	2564				2016	2716	2732	2532	2764	2564
		+-------+								
+5	+5	1	28pin	28					+5	+5
A12	*CS	2		27					+5	*CS
--	A7	3	-------	26	+5	--	--	--	--	--
--	A6	4	24pin	25	A8	--	--	--	--	--
--	A5	5		24	A9	--	--	--	--	--
--	A4	6		23	*WR	+5	A11	+5	A11	A12
--	A3	7		22	*OE	*OE	*OE	*CS	*OE	GND
--	A2	8		21	A10	--	--	--	--	--
--	A1	9		20	*CS	*CS	*CS	A11	*CS	A11
--	A0	10		19	D7	--	--	--	--	--
--	D0	11		18	D6	--	--	--	--	--
--	D1	12		17	D5	--	--	--	--	--
--	D2	13		16	D4	--	--	--	--	--
--	GND	14		15	D3	--	--	--	--	--
		+-------+								

NOTE: On a PCP-11, U18 must be equiped with a ROM startup program (supplied with a PCP-11S); the other sockets may be equipped as needed. U16 and U17 may be configured with static RAM for special

applications.

In this table, the logical memory signals are correlated with the jumper pad labels:

Figure 2-3 -- Memory Signal Jumpers

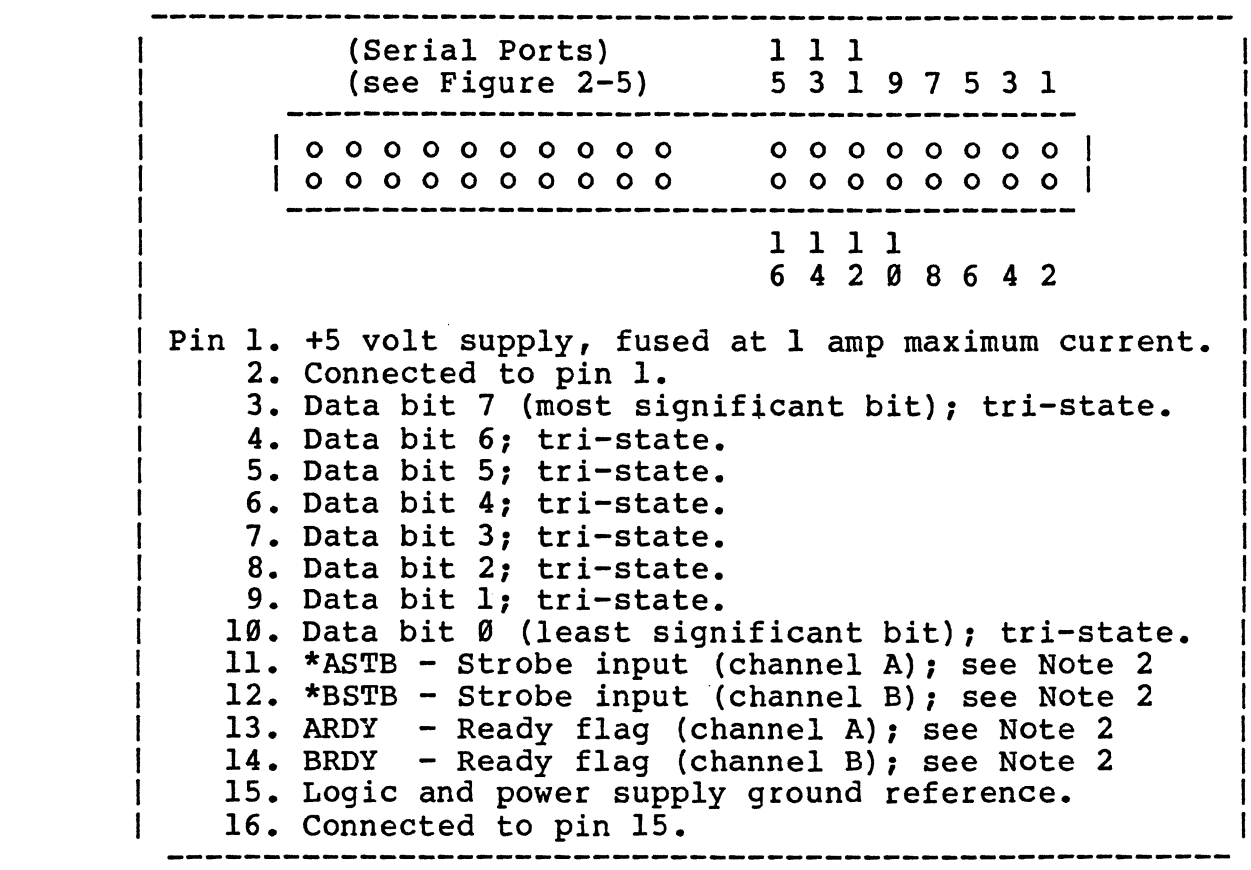


NOTE: The factory supplied jumper condition is specified in Figure 2-4. ROM Ø does not have jumpers F or G.

2.3 Parallel Port Connector

The following pinout applies to the PIO (channel A) parallel interface port connector (Ansley type 609-1607), which occupies the card edge immediately next to U36:

Figure 2-4 -- Parallel Port Connector



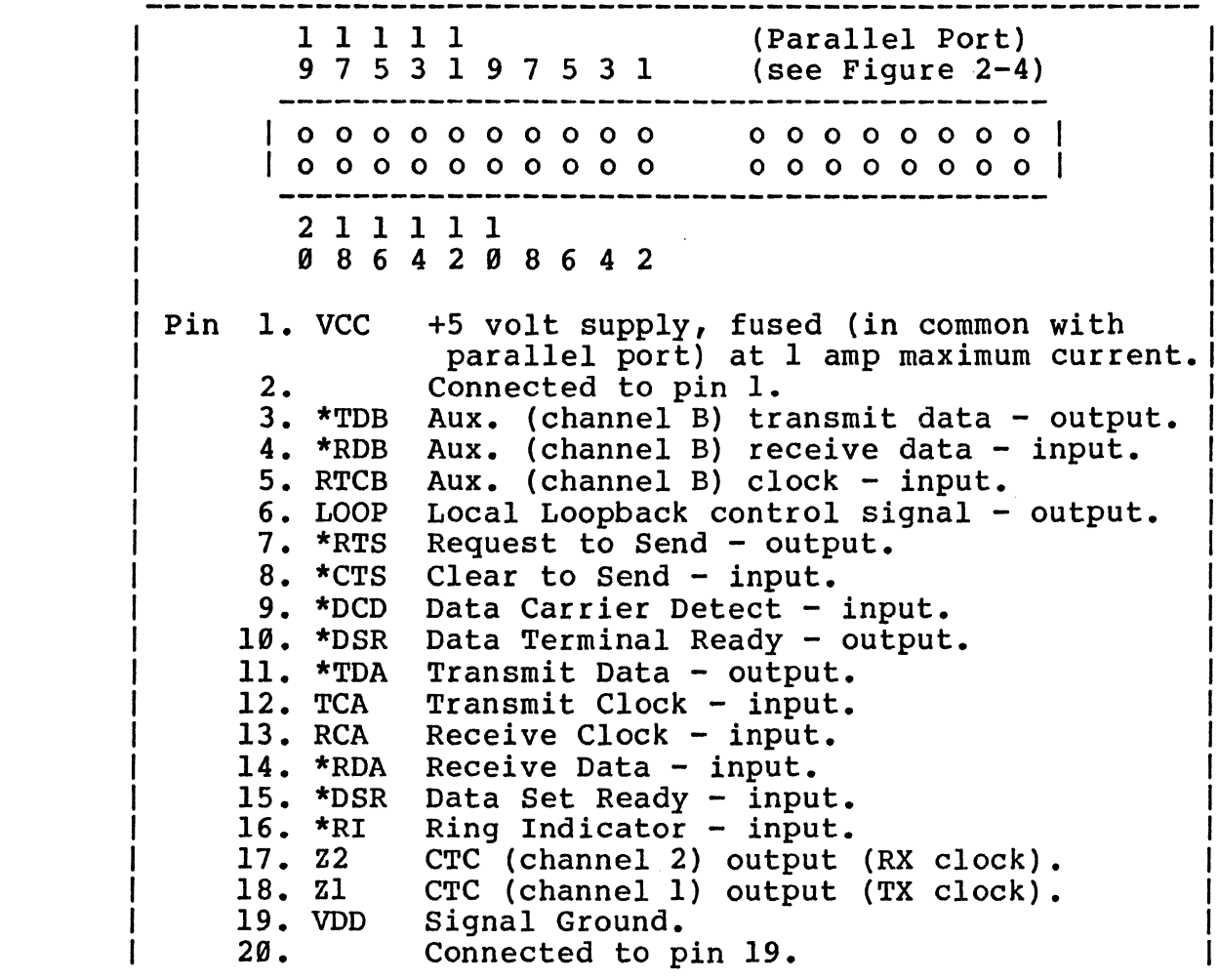
NOTE: Pin 1 is defined in accordance with industry practice (upper right hand corner when viewed from the mating connector's viewpoint) and corresponds to the marking on the connector housing (it is also labeled on the board). Pin numbering continues in ribbon cable order, alternating between upper and lower pin rows.

NOTE 2: See Zilog PIO data sheet for full functions.

2.4 Serial Ports Connector

The following pinout applies to the SIO serial interface ports connector (located as the higher order pins of the I/O connector, i.e. those pins adjacent to U35); note that all signals are TTL level and correspond to DTE designations unless otherwise indicated:

Figure 2-5 -- Serial Ports Connector



Pin			
	1.	VCC	+5 volt supply, fused (in common with parallel port) at 1 amp maximum current.
	2.		Connected to pin 1.
	3.	*TDB	Aux. (channel B) transmit data - output.
	4.	*RDB	Aux. (channel B) receive data - input.
	5.	RTCB	Aux. (channel B) clock - input.
	6.	LOOP	Local Loopback control signal - output.
	7.	*RTS	Request to Send - output.
	8.	*CTS	Clear to Send - input.
	9.	*DCD	Data Carrier Detect - input.
	1Ø.	*DSR	Data Terminal Ready - output.
	11.	*TDA	Transmit Data - output.
	12.	TCA	Transmit Clock - input.
	13.	RCA	Receive Clock - input.
	14.	*RDA	Receive Data - input.
	15.	*DSR	Data Set Ready - input.
	16.	*RI	Ring Indicator - input.
	17.	Z2	CTC (channel 2) output (RX clock).
	18.	Z1	CTC (channel 1) output (TX clock).
	19.	VDD	Signal Ground.
	2Ø.		Connected to pin 19.

NOTE: +5 volt DC power is supplied at each port for external use, and is current limited by fuse F1 (located adjacent to the connectors) to 1 amp for backplane power supply protection. Improper connections may result in inoperable external equipment. The fuse may be tested with a voltmeter at pin 1 (on either the serial or parallel ports) and ground. The fuse type is a 1 amp Picofuse (or equivalent).

Chapter 3

Theory of Operation

3.1 Host Interface

The PCP-11 may be conceptualized as a general purpose, programmable slave processor that communicates with its host LSI-11 via an interrupt driven, programmed I/O interface. The exact protocol used (and the meanings of register bits) will depend upon the intended application.

A high degree of design flexibility is provided through the native structure of this interface, which is patterned after an extended DLV-11 type serial interface, and includes:

* A 16 bit read-only Host Receiver Data Register, mapped as 2 consecutive write-only MPU port addresses;

* A 16 bit write-only Host Transmitter Data Register, mapped as 2 consecutive read-only MPU port addresses;

* Two Host Control/Status Registers (CSR) with standard DONE and INTERRUPT ENABLE status bits;

* Two MPU Data Register Ready status bits, maskable as MPU interrupt sources, and mapped into the MPU System Status port; and

* Four read-only status bits, split between the two CSRs, and mapped into a read/write MPU port address.

Several differing approaches may be utilized in controlling the PCP-11 and transferring data between it and its host. These include:

* An 'asynchronous' serial model where data is transferred using the low bytes of the data registers while command/status information is transferred using the high bytes. Interrupts would generally occur with each transfer;

* A 'synchronous' or block model where data and commands utilize the full 16 bit facilities of the data registers. An interrupt signals readiness, and the data is rapidly transferred while remaining within a service routine. This approach is used with traditional floppy disk controllers and has the advantage of reducing host servicing requirements.

3.2 Microprocessor and Control

The PCP-11 family uses a 4 MHz Z80A microprocessor as its basic control element. The M1 opcode fetch cycle is stretched by one wait state, enabling use of industry standard EPROMs and ROMs, as well as static RAMs for program and data memory.

A power-up reset is performed in synchronization with the LSI-11's starting sequence. In most cases, PCP-11 initialization will be accomplished before the LSI-11 completes its bootstrap sequence. A parallel port is used for system status information, with programmable interrupt capability on user-defined significant events.

The power-up sequence for the distributed resident software performs the following functions:

1) Clears the LSI-11 interface registers;

2) Validates the checksum for the operating system ROM. The TEST LED will be lit at this time. If the checksum test fails, the software will continue to execute the same test, with the TEST LED lit;

3) Conducts a basic read/write test on the dynamic RAM. Again, the TEST LED is lit, and remains lit until the test succeeds;

4) Initializes the dynamic RAM to all ones (0FF hexadecimal),so an errant program will cause a restart. This is instruction, RST 7;

5) Checks the first byte of the expansion ROM2 for a JMP instruction. If found, executes it, transferring control to a user application ROM;

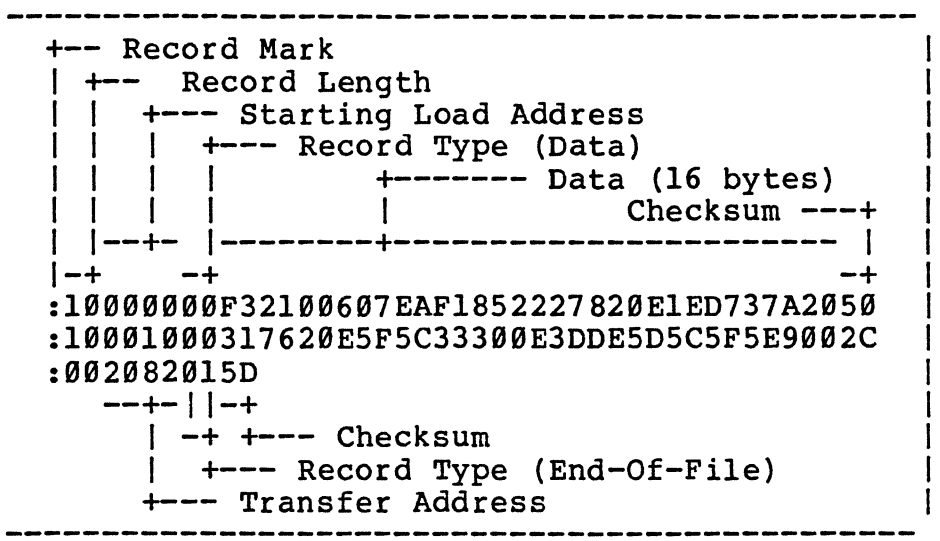
6) If #5 fails, checks bit zero of the switch register. If ON (open - high), it executes an object module down-loader. The object module should be supplied to a PCP-11S via the DL-11 compatable LSI-11 bus interface in the Intel HEX format;

7) If #6 fails, an operating system dependent initialization occurs and, on a PCP-11S, the PCP-Sphere resident operating environment starts (see the PCP-Sphere manual for details).

When an object module is loaded into the PCP-11S, the monitor scans the 'console' input stream for a valid record mark (:), and then stores the associated data into the PCP-11S memory. If the loader sees an end-of-file record (type 1) with a zero address, control will be transferred back to monitor level. If, on the other hand, the EOF record contains a non-zero address, the address will be stored in the user's saved PC register and control is transferred to the down-loaded program.

NOTE: No attempt is made to verify that the data has been properly stored. The format of an Intel HEX is summarized in Figure 3-1.

Figure 3-1 -- Intel HEX Object Format



Two hardware level methods exist for the host LSI-11 to place the PCP-11 in a known condition:

1) Strobing bit 15 of the Receiver Control Status Register causes a hardware reset of the PCP-11, just as if a power-up sequence had occured.

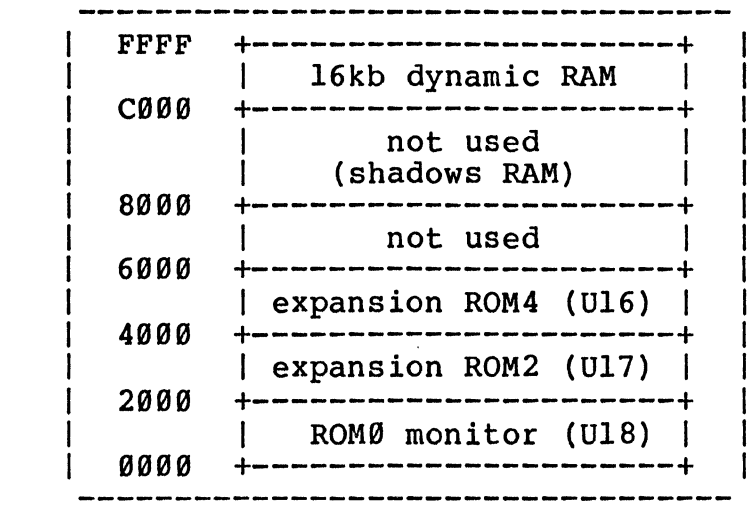
2) Strobing bit 14 of RCSR causes a non-maskable interrupt to occur. The effect of this action depends upon the software operating within the target PCP-11. Under the PCP-Sphere operating environment, this normally would cause an immediate ABORT, with execution transferring to the routine defined by the applications software.

NOTE: Under rare circumstances, improperly sequenced LSI-11 power signals (or a similar condition) may cause a PCP-11 to not properly initialize. The PCP-11 may always be restarted by strobing RCSR bit 15, as noted above.

3.3 Memory

The following memory map applies to the PCP-11:

Figure 3-2 -- PCP-11 Memory Map



NOTE: The 16kb of RAM actually appears twice in the memory map due to partial decoding; the higher order addresses are used by convention. Also, each ROM socket selects an 8kb area; with partial decoding, smaller parts will shadow within a ROM range. The floating value of unimplemented memory is undefined.

3.4 I/O Ports

The following I/O map applies to the PCP-11, with the Z-80 port addresses expressed in hexadecimal:

Figure 3-3 -- PCP-11 Port Map

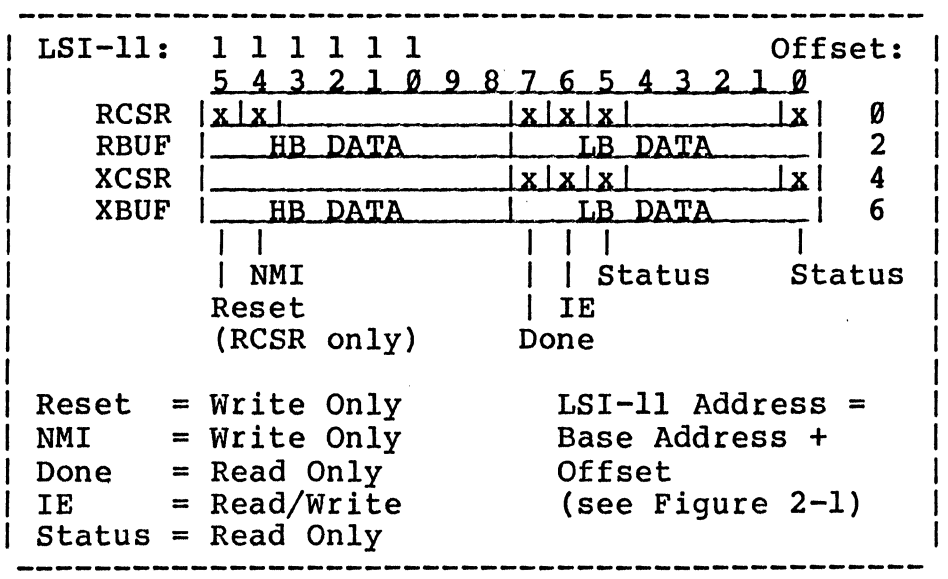
FF	not used (shadows lower ports)	
1E	LSI-11 Registers	R/W
1C	optional 2653 PGC	R/W
18	not used	
10	(shadows 0C-0D)	
0E	LSI-11 Status bits	R/W
0D	Switch Register	R
	Pulses Test LED	W
0C	CTC Channel 3	R/W
0B	CTC Channel 2	R/W
0A	CTC Channel 1	R/W
09	CTC Channel 0	R/W
08	SIO Channel B	R/W
06	SIO Channel A	R/W
04	PIO Channel B	R/W
02	PIO Channel A	R/W
00		

NOTE: Only partial address decoding is performed; the entire array of I/O ports shadow 7 times. As with memory, the floating value of unimplemented I/O addresses (and bits) is undefined.

Within the I/O port map, the LSI-11 data registers map to locations 1C and 1D (hexadecimal) on the PCP-11 and are read or write only, with both registers being mapped to the same location. The LSI-11 status registers map to bits within locations Ø2 and ØD (hexadecimal) on the PCP-11.

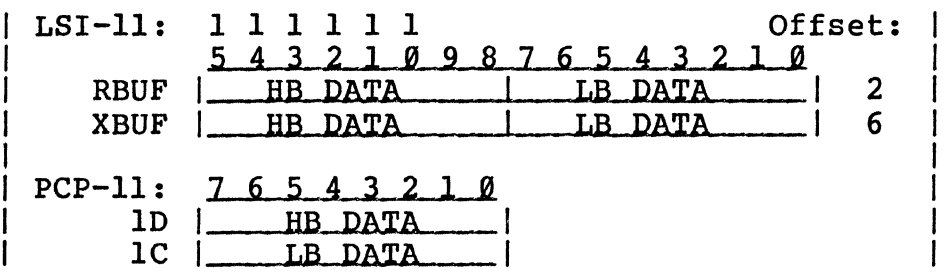
From the LSI-11 bus, the data and status registers can be conceptualized as:

Figure 3-4 -- LSI-11 Data and Status Registers



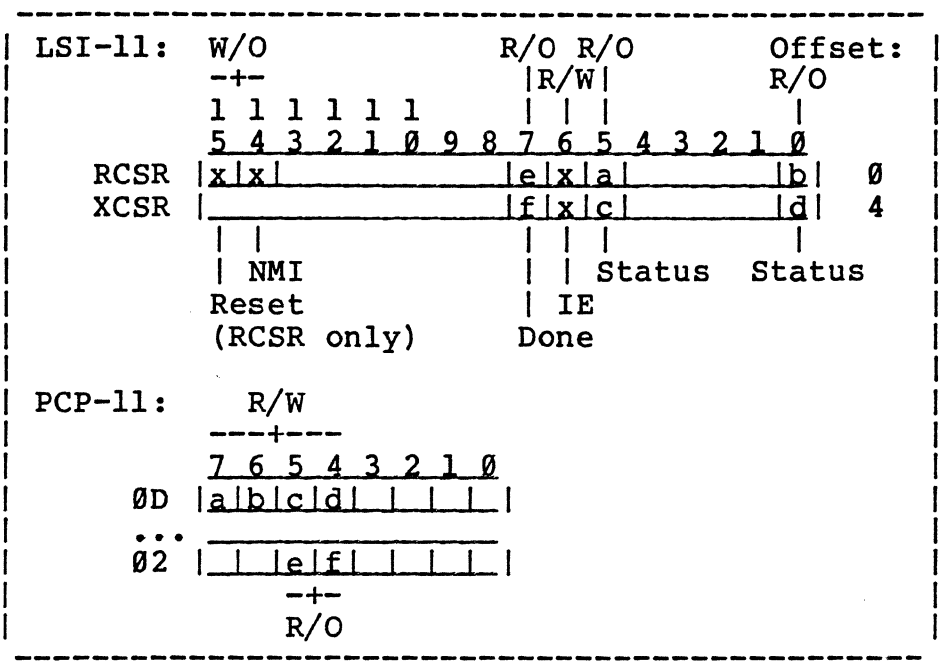
From the LSI-11, XBUF can be written to and RBUF can be read, but not vice versa. From the PCP-11, The data is read from and written to the same address, ports 1C and 1D hexadecimal.

Figure 3-5 -- LSI-11 <> PCP-11 Data Registers



The bit values in the CSR registers on the LSI-11 map to the PCP-11 ports shown below (there is no corelation for the bits marked x) with the Read/Write characteristics indicated:

Figure 3-6 -- PCP-11 Status Registers



3.4.1 PIO Parallel Ports (00-03)

The Zilog PIO chip serves a dual purpose on the PCP-11:

1) The first port (channel A) is available at the card edge for external expansion use (see Figure 2-4);

2) The second port (channel B) serves as a system status/control register, and is always operated in bit control mode.

The data bits associated with Channel B are defined as follows:

Figure 3-7 -- PIO Channel B Bit Significance

Bit Ø	PRDY	PIO B channel ready line (pin 21) - input.
1	*DSR	Active low Data Set Ready from modem - input.
2	*RING	Active low Modem Ring Indicator - input.
3	*LOCAL	Active low maintenance loop-back control for serial interface connector board - output.
4	XRDY	High when data available from LSI-11 - input (also CTC 3 trigger).
5	*RDONE	Low when LSI-11 able to accept data - input.
6	*MAINT	Low when PCP-11 installed in backplane - input.
7	*PGCI	Low when optional 2653 polynomial chip is requesting an interrupt.

NOTE: Any, or all, of these signals may be masked and serve as interrupt sources. However, the PIO interrupt logic is set for Mode 3 and is COMBINATORIAL, i.e., a change in the combination of all selected PIO interrupt sources must occur before the PIO will generate an interrupt. This is a common source of confusion when programming this device. See The Zilog Z8Ø PIO data sheet for details.

3.4.2 SIO Serial Interface (Ports Ø4 - Ø7)

The PCP-11 supports a dual channel serial interface (Zilog SIO/Ø). This circuitry implements asynchronous, isosynchronous, and synchronous protocols, and is the primary reason for the PCP-11's existance. The primary channel (A) has been designed for full user configurability. The auxiliary channel (B) is intended for asynchronous use only and deviates from normal implementation in that the auxiliary status inputs (DCD, CTS) are unused and tied to the respective control outputs (DTR, RTS); only RX, TX, and clock are supported for this channel.

In some respects, the SIO is more complex to program and control than the associated Z8Ø. Most of the SIO's behavior is defined in the Zilog/Mostek data sheet; however, there are several subtleties that require VERY careful reading to catch. The following points highlight some of the common problems encountered when programming this chip:

* The status lines (CTS, DCD, etc) are captured by transition detectors, and are only reset by explicit command. The user must carefully consider when to reset these input lines.

* Framing Error and Break condition are considered to be seperate status events. The technical definitions for these events are not provided by Zilog, nor are the bit synchronization and status recovery techniques employed by the chip.

* The bits per character setting logic does not follow a binary progression, instead following the sequence 5, 7, 6, 8.

3.4.3 CTC Counter Timer Circuit (Ports Ø8 - ØB)

The PCP-11 uses a Zilog CTC circuit to produce the baud rate clocks for the serial channels (when configured to use local clocking). Additionally, the CTC provides a programmable real-time clock, and a unique interrupt processor to indicate that the host LSI-11 has presented data to the PCP-11.

Channel Ø is used for real-time clocking, where the counter input is the 5Ø/6Ø Hz line frequency typically present as backplane signal BEVENT. In timer mode, it can derive periods from the system 4 MHz clock.

Channels 1 and 2 serve the baud rate functions, and are normally used in counter mode, where their input source is a 2 MHz clock. Divided by thirteen, this generates a 96ØØ baud (x16) maximum local clock rate. The outputs from these two channels are directed to the serial level shifting circuitry contained on the adapter card. Note that the CTC outputs are short (25Ønsec) pulses that may not meet the timing requirements of all UARTs (the external pod has circuitry designed to overcome this problem). The following table lists settings for common baud rates:

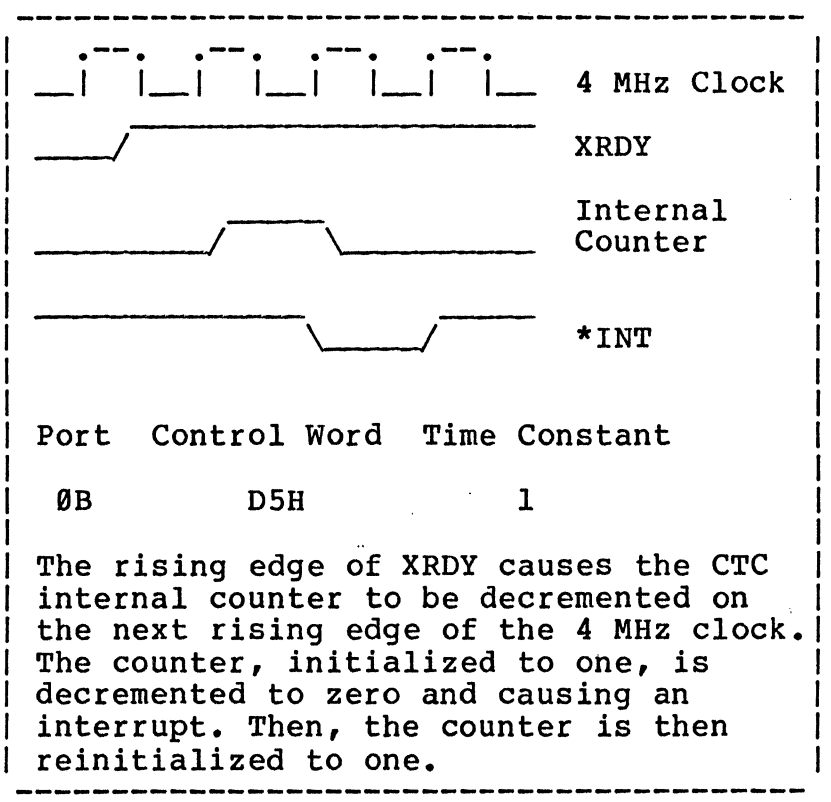
Figure 3-8 -- Baud Rate Time Constants

Rate	Control Word	Time Constant
96ØØ	45H	ØDH/13.
48ØØ	45H	1AH/26.
24ØØ	45H	34H/52.
12ØØ	45H	68H/1Ø4.
3ØØ	Ø5H	34H/52.

NOTE: For rates below 12ØØ (x16) baud, operation in timer mode is necessary to generate the longer time constants required. For asynchronous rates higher than 96ØØ (x16) baud, an external clock is required.

Channel 3 has the XRDY (LSI-11 transmitter ready) signal as a counter source. When set for a count frequency of one, it can generate interrupts each time the LSI-11 loads data into the interface registers.

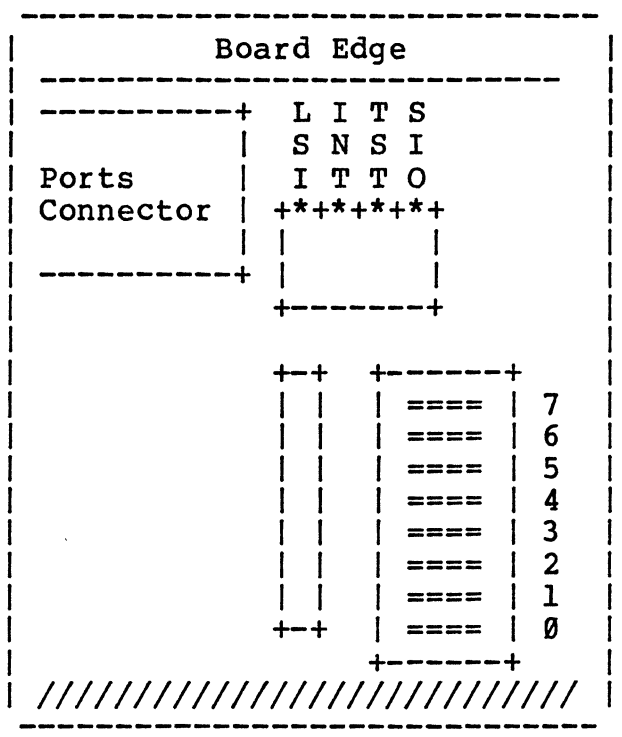
Figure 3-9 -- CTC Channel 3 With XRDY Trigger



3.4.4 Test LED and Switch Register (Port ØC)

As an input source, Port ØC provides the contents of the 8 bit DIP switch register indicated by Figure 3-9. Bit Ø is used by the system monitor(s) to indicate that auto-load of an object program is to occur on power-up. Unless pre-empted by special operating system functions, the remaining switches may be used by applications software to auto-configure. Note that the bit correlation is printed on the board surface next to each switch and may vary from the numbering contained upon the switch package.

Figure 3-10 -- Test LEDs and Port 0C Switch Register



As an output destination, Port 0C pulses the TXT LED (marked on the edge of the card). The monitor(s) use this LED to indicate that self-test routines are executing.

3.4.5 Polynomial Generator/Checker (Ports 18 - 1B)

The PCP-11 is designed to accept a Signetics 2653 Polynomial Generator and Checker (PGC) circuit in U24. This chip is specifically designed to assist in character class detection and CRC calculations for Bisynch protocol applications (its use for other applications is limited). The PGC is available as an option from the factory, as well as through normal distribution channels.

Since the PGC does not support the Zilog interrupt protocol, its request line is accessed via the PIO, which in turn can generate an appropriate interrupt request.

3.4.6 Host LSI-11 Data Registers (Ports 1C - 1D)

Primary communications with the host LSI-11 occur through I/O ports 1C (LSI-11 low register bytes) and 1D (high bytes). These registers are read/write, but data does not echo back (i.e. the Z80 reads from and writes to the LSI-11 interface registers).

Full handshaking is supported; a set of flip/flops set or reset (as appropriate) each time access is made to these registers and ports. If the associated status lines are polled, or used as a source of interrupts, overrun conditions will not occur.

Note that handshaking only occurs on the low order byte (port 1C), so that the Z80 can process a 16 bit transfer with only one interrupt. Typical Z80 protocol is to manipulate the high byte first, and then the low byte (which causes a change in handshaking).

The RBUF and XBUF registers are illustrated in Figure 3-5.

3.4.7 LSI-11 Status Pits (Ports 02 and 0D)

In addition to the basic 16 bit bidirectional data registers used to interface the PCP-11 to its host LSI-11, 4 status bits can be set or read at port 0D and 2 status bits may be read at port 02 by the Z80 to provide additional information to the host.

The RCSR and XCSR registers are illustrated in Figure 3-6.

3.5 Z80 Interrupts

The PCP-11 I/O ports can be manipulated either on a polled basis (i.e. by checking status bits), or by using interrupts to indicate changes in device status.

While the Z80 supports three methods of interrupt control, only the daisy-chained type 2 method is meaningful on the PCP-11. The hardware has been structured so that only those devices that support type 2 interrupts may generate them (as noted above, those that can't, pass their request on to a device that can relay it for them).

The following table lists the type 2 interrupt sources, in their priority order (highest listed first):

Figure 3-11 -- Z80 Type 2 Interrupt Sources

CTC Channel 0 - Real Time Clock
CTC Channel 1 - Baud Rate (normally not used)
CTC Channel 2 - Baud Rate (normally not used)
CTC Channel 3 - LSI-11 Transmitter Ready (XRDY)
SIO Channel A - Receiver
SIO Channel A - Transmitter
SIO Channel A - External/Status
SIO Channel B - Receiver
SIO Channel B - Transmitter
SIO Channel B - External/Status
PIO Channel A - External Parallel Port
PIO Channel B - System Status Port

When implementing interrupt control software, the user should keep in mind the following characteristics of Mode 2 interrupts:

1) Once an interrupt request has been issued by a device, all other devices farther down the chain are blocked from acknowledging requests. This condition does not change until a return from interrupt has occurred for the highest requesting device.

2) The Z80 implicitly disables interrupts on reset, as well as each time an interrupt is acknowledged. User-supplied service routines must explicitly re-enable them. (However, the PIO does not reset on CPU reset.)

3) If a service routine re-enables interrupts prior to its completion, it may be interrupted by another device higher in the chain.

Chapter 4

PTE

PCP-11S Terminal Emulator

PTE allows the user to interact with a PCP-11S, using the host console terminal and RT-11 system as the 'console' terminal to the PCP-11S. In other words, PTE makes your LSI-11 system look like a terminal to the PCP-11S resident operating system.

PTE allows you to:

* Input data to the PCP-11S from your keyboard (any 7 bit data, except for three reserved characters; ^X, ^Z, and ESC).

* Output data from the PCP-11S to your display.

* Input or output data to/from the PCP-11S and an RT-11 device. (This is especially useful for loading source and object modules or creating a log file of a development session.)

4.1 Invoking PTE

To call PTE from the system device, respond to the dot printed by the keyboard monitor by typing:

```
.R PTE
```

PTE will start up in interactive mode, where any data output from the PCP-11S will be displayed on your terminal, and anything you type will be input to the PCP-11S. This is the normal mode that PTE is used in.

NOTE: The data flow in each direction is double buffered by PTE, helping to insure that no data can be lost. However, in the unlikely event that the keyboard type-ahead buffer overflows, PTE will alert the user by ringing the console bell.

Under some circumstances, VT-100 terminals (and similar devices that use XON and XOFF with their host), may cause output to be suspended while they perform an internal operation. If this occurs, typing ^Q (control Q) from the keyboard will restore the data flow.

Control character transparency is accomplished by gaining control of the keyboard interrupt vector. This means that communications with RT-11, and control character processing, do not normally occur.

4.2 Mode Control

The user can change the mode in which PTE operates by obtaining PTE's attention. This is accomplished by typing the ESCAPE key; PTE responds to this reserved character (which can not be sent to the PCP-11S from the keyboard) with an asterisk, indicating its readiness to accept a command.

Once the program has entered command mode, it restores control of the console keyboard to RT-11. Commands are processed by the Command String Interpreter (CSI). Consequently, all the normal RT-11 line editing commands are available when in command mode.

The valid commands, and their arguments, are listed in Table 4-1.

Figure 4-1 -- PTE Mode Options

/? Displays a summary of the switches (options) available.

/A ASCII mode is used for most interactive communications, as well as source file transfers. In this mode, data is stripped to the least significant 7 bits, line feeds are removed if they follow a carriage return, and nulls are ignored. This is a default setting.

/C Console mode is also a default setting, and disables sending or storing of file transfer data.

/F Full Duplex mode is the normal style of interactive communications. In this mode, characters sent to the PCP-11S are expected to be echoed by the PCP-11S.

/H Half Duplex mode causes PTE to echo back each character sent to the PCP-11S, as it is sent.

/I Image mode disables the ASCII mode, and allows transparent transfers of 8 bit data.

/N New file specification causes PTE to close any currently open RT-11 files, parse the file specification, open the file (if one is given), and start the file transfer.

/X Exit causes PTE to perform an orderly EXIT to RT-11, closing any currently open RT-11 files.

4.3 File Transfers

Data files may be transferred to or from the PCP-11S target system, in ASCII (coded) or image (transparent) mode.

To initiate a transfer, enter command mode and respond to the CSI prompt with a file specification and the /N switch. Specifying a file causes any currently open files to be closed, and immediately initiates the transfer. Note that transfers may only occur in one direction at any time. The default file type is .HEX.

For example, the following interaction might be used to load a source module when using PCP-Sphere:

```
.R PTE
> *FOO.NTH/N
```

In this example, the user typed an ESCAPE, the CSI responded with the command prompt, and the user indicated the file DK:FOO.NTH was to be sent to the PCP-11S. The file would be down-loaded in source form to PCP-Sphere as if the user had directly entered it. When end of file is reached, PTE transfers control back to the keyboard.

Alternately, by terminating the file name with an equals (=) or less than sign (<), an output file may be specified. In this case, PTE will store any data received from the PCP-11S in the named file. This option might be used to save an object module created with PCP-Sphere or to create a transaction log.

NOTE: While a file transfer is underway, you may still enter keyboard data, which will be merged with the file data. This allows suspension of a transfer by forcing entry to command mode, with subsequent specification of a null file transfer.

4.4 Terminating PTE

Since PTE assumes control of the keyboard vector, the only way to interact with RT-11 is by entering command mode. Once in command mode, the /X switch will perform an orderly EXIT to RT-11, closing any open files.

WARNING

Since data transfers to the PCP-11S are not performed under interrupt control, it is possible to lock up the system if the PCP-11S ceases to accept keyboard input.

In this event, it is not possible to enter command mode, nor to subsequently issue the /X command. However, since PTE saves the normal console vector at location 1012 (octal) console ODT may be used to restore communications with RT-11. Alternately, the system can be rebooted.

4.5 Regaining Control of the PCP-11S

PTE uses two reserved characters to support user level control over the PCP-11S. Typing a ^Z (Control Z) causes the target PCP-11S to perform a cold restart, just as if a power up sequence had occurred. Typing a ^X (Control X) generates a non-maskable interrupt to the PCP-11S (which under the PCP-Sphere operating environment, normally causes an immediate ABORT).

Note that these special operators are processed by PTE immediately, if it has control of the console vector (i.e. they can be used to break out of a keyboard lock-up condition).

4.6 Changing the PCP-11 Addresses and Vectors

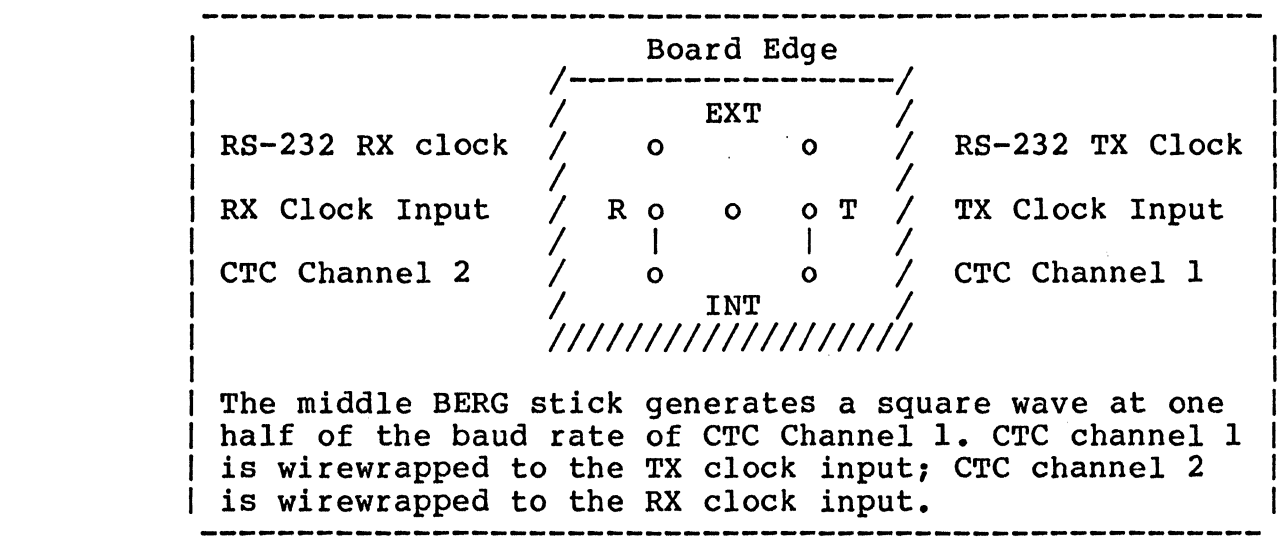
Locations 1000 to 1010 (octal, inclusive) contain the addresses and receiver vector of the target PCP-11. While normally configured to match the factory default hardware settings, they may be patched by the user if the need occurs to change these settings on the PCP-11.

Appendix A

RS-423 Serial Interface Pod

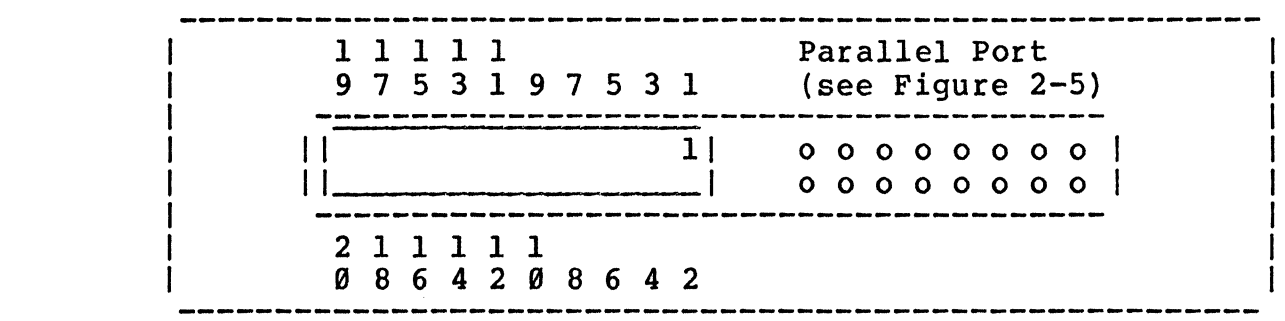
PCP-11 serial interface loopback circuitry and level shifting occurs in an external pod. This pod generates RS-423 signals in RS-232 compatible mode (±5 volts inverted logic). This assembly contains a pair of BERG stick connectors for selecting the transmit and receive clock sources for the primary serial channel. The pinout follows:

Figure A-1 -- Primary Serial Channel Clock Jumpers



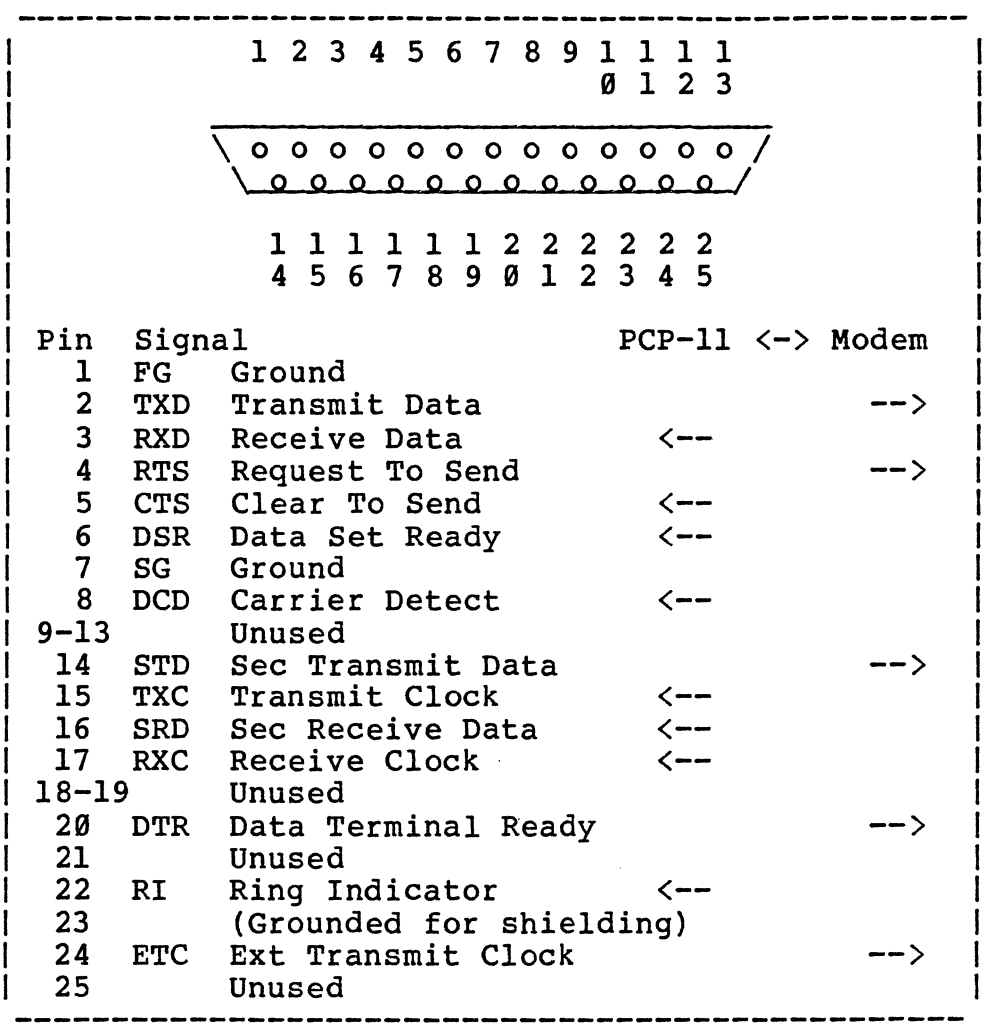
The serial interface pod includes a 20 conductor ribbon cable, with connector, that plugs into the PCP-11 serial ports connector. The 20 pin connector is plugged into the 40 pin connector so that the smaller connector is flush with the edge closest to the card handle.

Figure A-2 -- PCP-11 and Pod Interconnection



The assembly also includes a 25 conductor cable, which terminates in a male DB-25 connector. This cable carries the following signals:

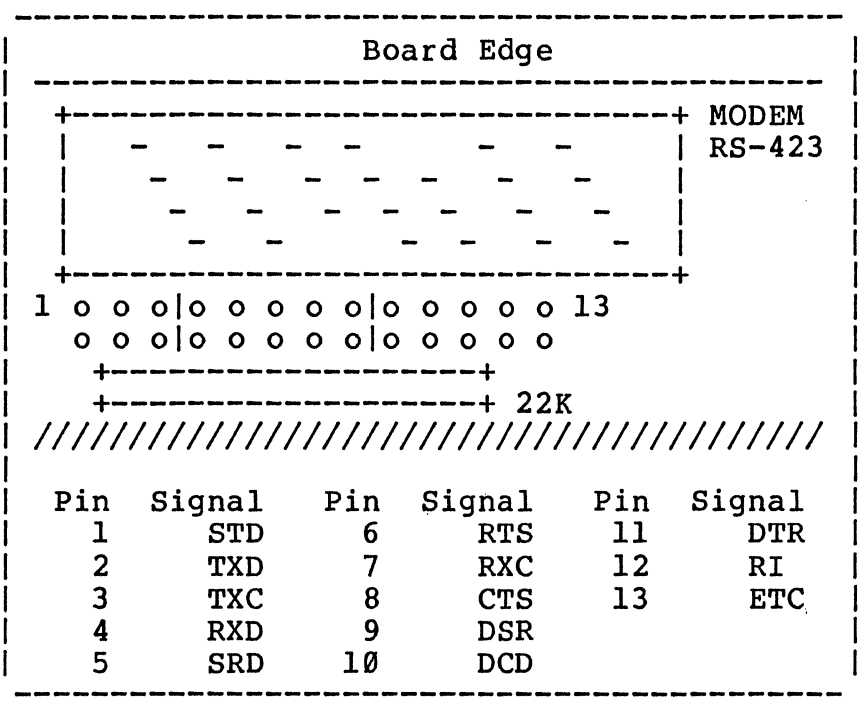
Figure A-3 -- DB-25 Pinouts



Pin	Signal		PCP-11 <-> Modem
1	FG	Ground	
2	TXD	Transmit Data	-->
3	RXD	Receive Data	<--
4	RTS	Request To Send	-->
5	CTS	Clear To Send	<--
6	DSR	Data Set Ready	<--
7	SG	Ground	
8	DCD	Carrier Detect	<--
9-13		Unused	
14	STD	Sec Transmit Data	-->
15	TXC	Transmit Clock	<--
16	SRD	Sec Receive Data	<--
17	RXC	Receive Clock	<--
18-19		Unused	
20	DTR	Data Terminal Ready	-->
21		Unused	
22	RI	Ring Indicator	<--
23		(Grounded for shielding)	
24	ETC	Ext Transmit Clock	-->
25		Unused	

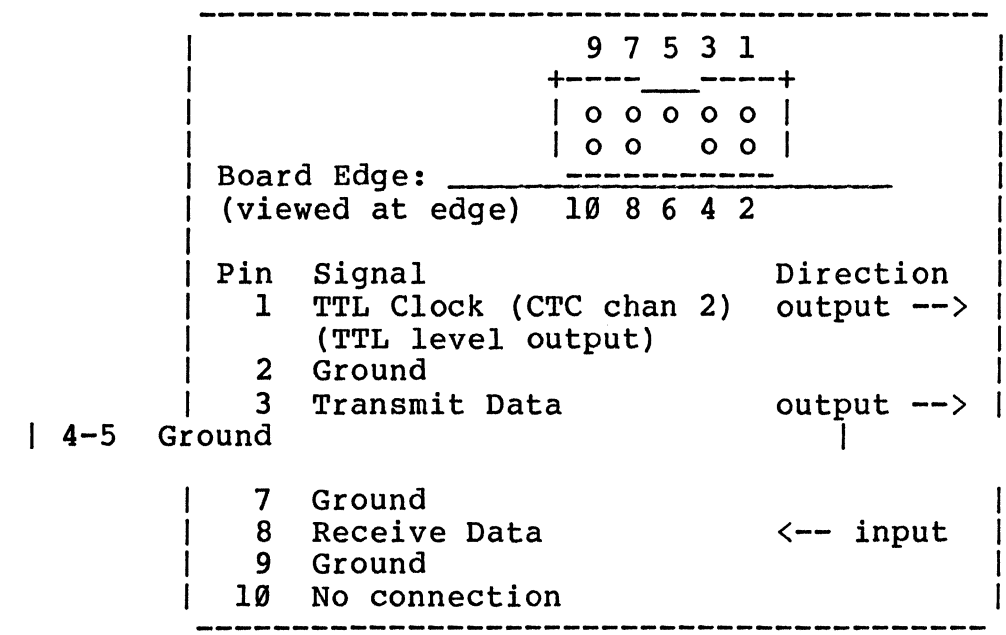
The thirteen active signals listed above pass through a jumper area, allowing the user to reconfigure the modem cable pinnouts (for example, to implement a null modem). The order of the signals within this jumper area is:

Figure A-4 -- Serial Interface Jumper Area



The assembly also contains a 10 pin connector for the secondary (channel B, or auxiliary) serial channel that conforms to the pinout of a DEC DLV-11J:

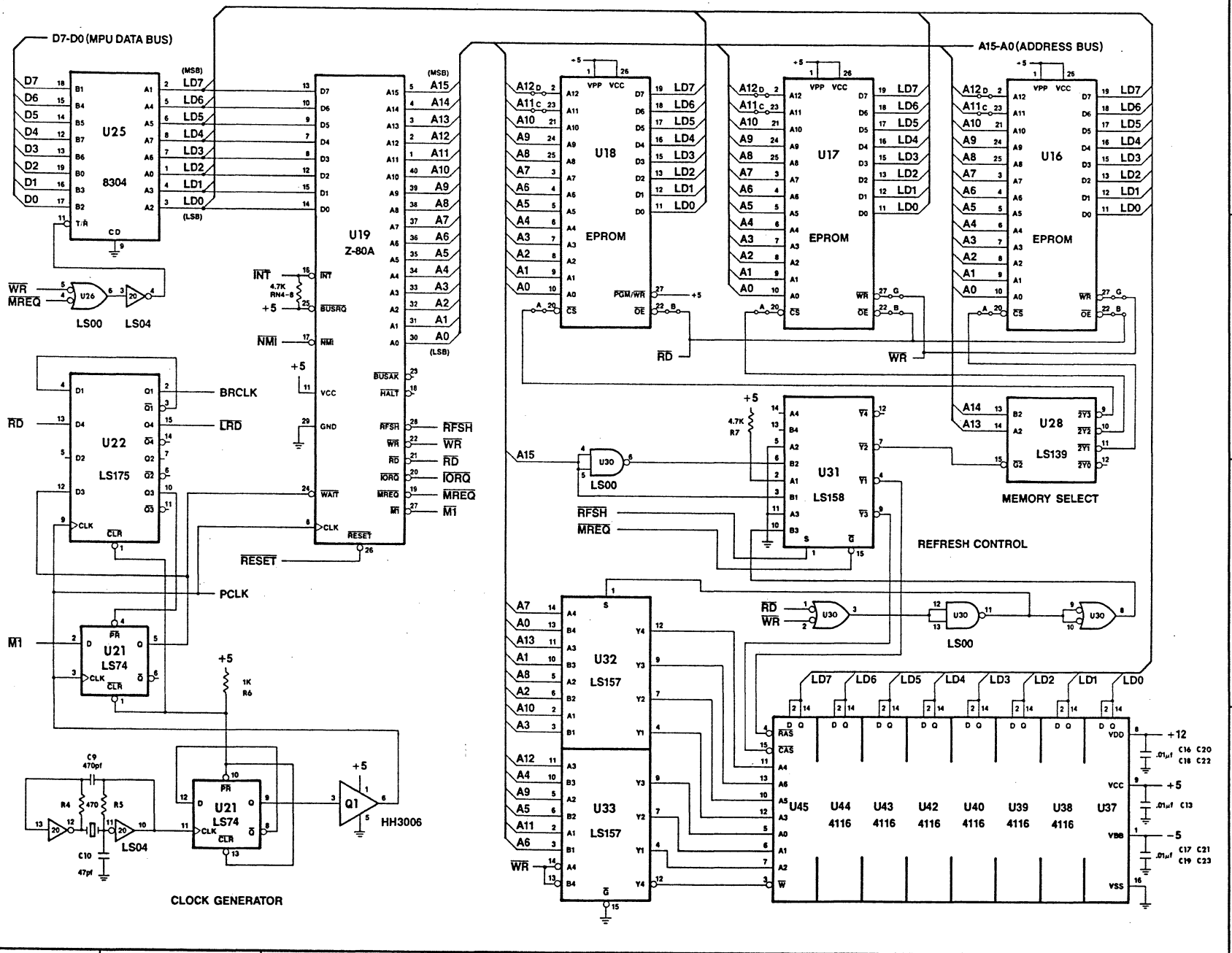
Figure A-5 -- Auxiliary (Channel B) Serial Port



Appendix B

PCP-11 Schematics

The information contained within the PCP-11 schematics is proprietary information of Infosphere, and is only released upon receipt of a non-disclosure agreement. The attached form should be completed and returned to Infosphere, Inc along with the request for schematics.

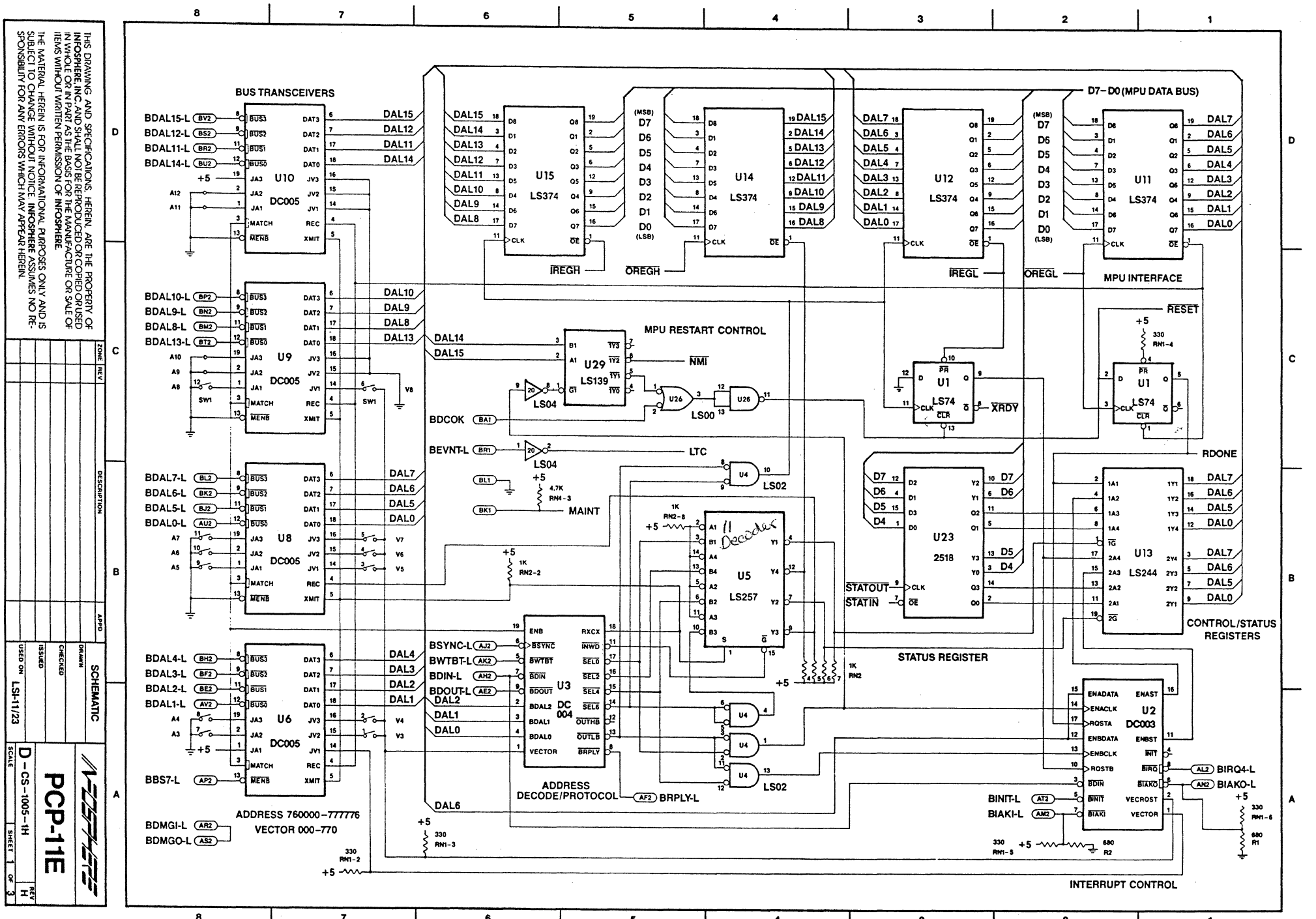


ZONE	REV	DESCRIPTION	APPD

SCHEMATIC	
DRAWN	
CHECKED	
ISSUED	
USED ON	LSI-11/23

PCP-11E

SCALE	
D-CS-1005-2H	
SHEET 2 OF 3	REV H



THIS DRAWING AND SPECIFICATIONS, HEREIN, ARE THE PROPERTY OF INFOSPHERE, INC. AND SHALL NOT BE REPRODUCED OR COPIED OR USED IN WHOLE OR IN PART AS THE BASIS FOR THE MANUFACTURE OR SALE OF ITEMS WITHOUT WRITTEN PERMISSION OF INFOSPHERE.

THE MATERIAL HEREIN IS FOR INFORMATIONAL PURPOSES ONLY AND IS SUBJECT TO CHANGE WITHOUT NOTICE. INFOSPHERE ASSUMES NO RESPONSIBILITY FOR ANY ERRORS WHICH MAY APPEAR HEREIN.

ZONE	REV	DESCRIPTION	APPD

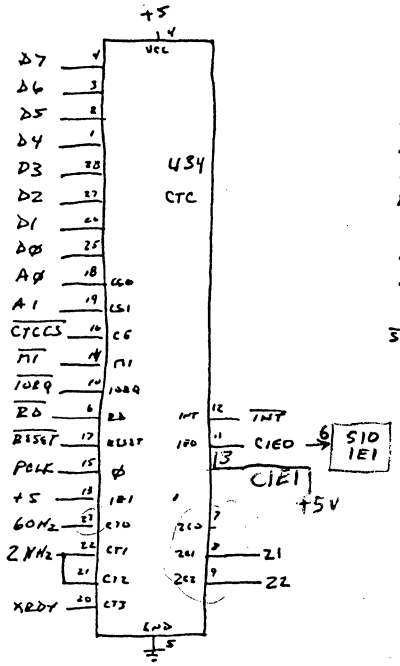
USED ON: LSI-11/23
ISSUED
CHECKED
DRAWN
SCHEMATIC

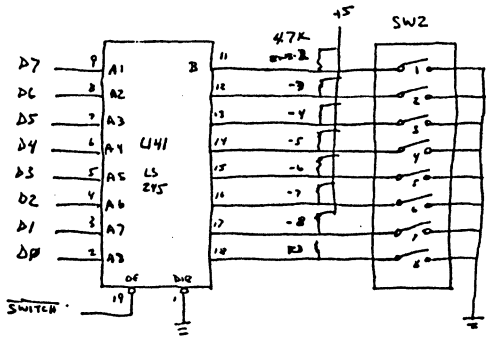
INFOSPHERE

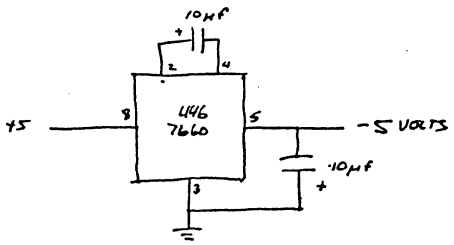
PCP-11E

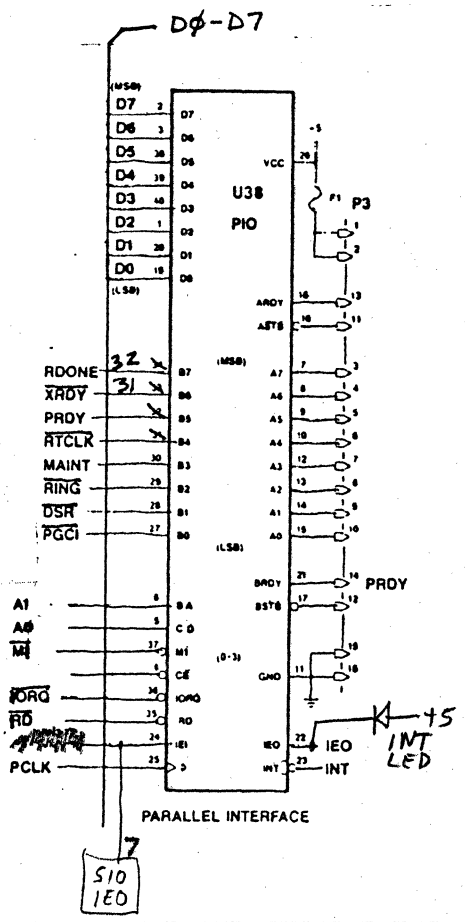
SCALE
D-CS-1005-1H
SHEET 1 OF 3
REV H

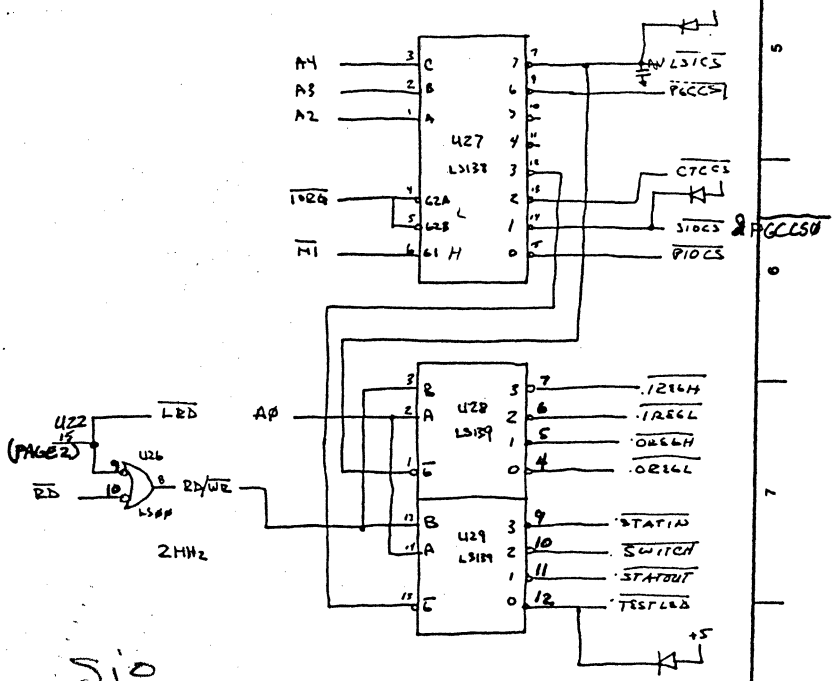
Inward & reading from PCP
Sel = 666
out_ = PCP











SIO

over

ZONE	REV	DESCRIPTION	APPD

SCHEMATIC

DRAWN

CHECKED

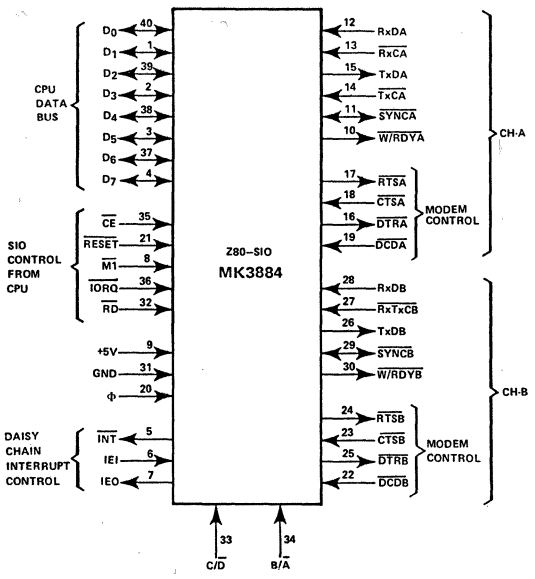
ISSUED

USED ON

INFOSPHERE

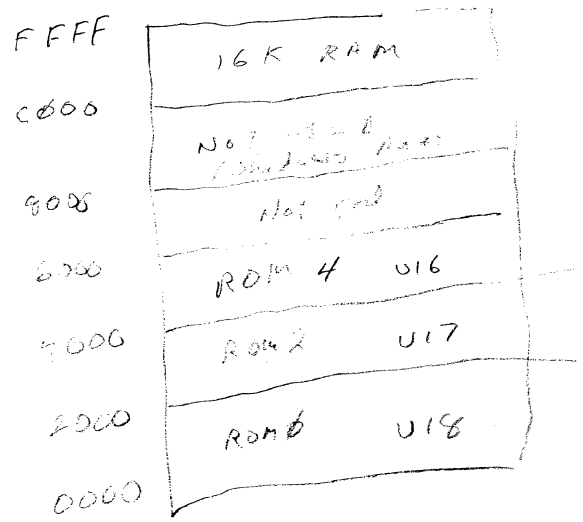
PCP-11E

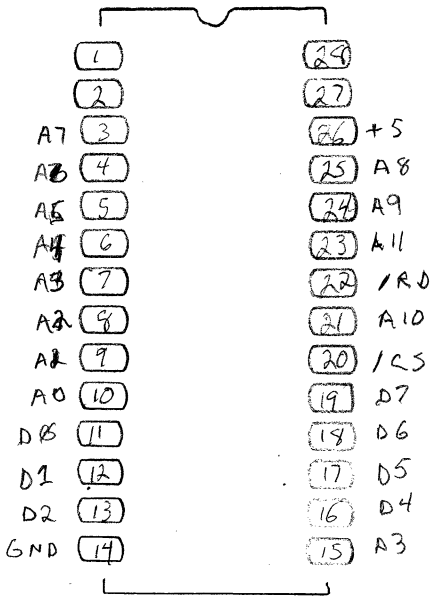
D-CS-1005-2H REV H


Z80–SIO
MK3684
CPU DATA BUS
D0 40
D1 1
D2 39
D3 2
D4 38
D5 3
D6 37
D7 4
SIO CONTROL FROM CPU
CE 35
RESET 21
M1 8
IORQ 36
RD 32
+5V 9
GND 31
Φ 20
DAISY CHAIN INTERRUPT CONTROL
INT 5
IEI 6
IEO 7
C/D 33
B/A 34
12 RxDA
13 RxCA
15 TxDA
14 TxCA
11 SYNCA
10 W/RDYA
17 RTSA
18 CTSA
16 DTRA
19 DCDA
MODEM CONTROL
CH-A
28 RxDB
27 RxTxCB
26 TxDB
29 SYNCB
30 W/RDYB
24 RTSB
23 CTSB
25 DTRB
22 DCDB
MODEM CONTROL
CH-B

Section 2.2

PCP-11 Default

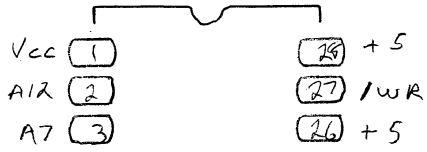




Boot socket

Aha is 2732

4K by 8



Other sockets

2764

8K x 8

or

In theory 6264

8K x 8 SRAM

where CS2 is tied to +5 (?)

$3.75