



Little BoardTM /286

Technical Manual

P/N: 5000281

Revision: A

Ampro Computers, Incorporated
4757 Hellyer Avenue ■ San Jose, CA 95138
Tel (408) 360-0200 ■ FAX (408) 360-0220
<http://www.ampro.com>

NOTICE

DISCLAIMER

Ampro Computers, Incorporated makes no representations or warranties with respect to the contents of this manual or of the associated Ampro software products, and specifically disclaims any implied warranties of merchantability or fitness for any particular purpose. Ampro shall under no circumstances be liable for incidental or consequential damages or related expenses resulting from the use of this product, even if it has been notified of the possibility of such damages. Ampro reserves the right to revise this publication from time to time without obligation to notify any person of such revisions. If errors are found, please contact Ampro at the address listed on the title page of this document.

TRADEMARKS

MiniModule, MiniBackplane, Little Board, StackPlane: Ampro Computers, Inc.

AT, PC/AT, IBM: International Business Machines Corp.

DR DOS: Digital Research, Inc.

Hercules: Hercules Computer Technology

MS-DOS: Microsoft, Inc.

UNIX: AT&T Bell Laboratories

XENIX: Santa Cruz Operation, Incorporated

QNX: Quantum Software Systems, Ltd.

COPYRIGHT 1989, 1990 AMPRO COMPUTERS INCORPORATED

No part of this document may be reproduced, transmitted, transcribed, stored in a retrieval system, or translated into any language or computer language, in any form or by any means, electronic, mechanical, magnetic, optical, chemical, manual, or otherwise, without the prior written permission of Ampro Computers, Incorporated.

PREFACE

This manual is for integrators and programmers of systems based on the Ampro Little Board/286 single board system. It contains information on hardware requirements and interconnection, and details of how to program the system.

There are four chapters, organized as follows:

- **Chapter 1 -- Introduction.** General information pertaining to the Little Board/286, its features, and specifications.
- **Chapter 2 -- Hardware Configuration.** A description of how to configure and connect the Little Board/286 for use with a wide variety of onboard and external devices. Included are tables listing the pinouts of each of the board's connectors, board jumper and Configuration Memory setup, and considerations and specifications regarding peripheral devices.
- **Chapter 3 -- Software Configuration.** An overview of the system features, configuration options, and utilities that are available under the "DOS" (PC-DOS, MS-DOS, or DR-DOS) operating system, including system setup guidelines.
- **Chapter 4 -- Theory of Operation.** Detailed technical information on Little Board/286 onboard hardware and peripheral interfaces.

TABLE OF CONTENTS

CHAPTER 1 - INTRODUCTION

1.1	General Description	1-1
1.2	Little Board/286 Features.....	1-3
1.3	Little Board/286 Specifications.....	1-4

CHAPTER 2 - HARDWARE CONFIGURATION

2.1	Introduction	2-1
	2.1.1 Interface Connectors.....	2-1
	2.1.2 Jumper Configuration Options	2-3
	2.1.3 "Soft Configuration" Options	2-3
2.2	DC Power Input.....	2-4
2.3	Onboard Device Options and Configuration.....	2-6
	2.3.1 CPU and Bus Half-Speed Options.....	2-6
	2.3.2 DRAM Memory.....	2-6
	2.3.3 Math Coprocessor.....	2-7
	2.3.4 Byte-Wide Memory Sockets.....	2-8
	2.3.5 Battery-Backed Clock.....	2-10
2.4	Peripheral Connections and Configuration	2-12
	2.4.1 Utility Connector	2-12
	2.4.2 Keyboard.....	2-13
	2.4.3 Parallel Printer Port.....	2-14
	2.4.4 Serial Ports.....	2-16
	2.4.5 Floppy Disk Interface	2-17
2.5	SCSI Interface	2-20
	2.5.1 Normal Use of SCSI.....	2-21
	2.5.2 The Ampro SCSI/BIOS	2-21
	2.5.3 Simple Bidirectional I/O.....	2-22
	2.5.4 SCSI Interface Configuration	2-22
2.6	AT Expansion Bus	2-24
	2.6.1 Onboard MiniModule Header and Bus Expansion.....	2-24
	2.6.2 Using Standard PC and AT Bus Cards	2-25
	2.6.3 Expansion Bus Cable Considerations	2-26
	2.6.4 Expansion Bus Connector Pinouts.....	2-26
	2.6.5 Interrupt and DMA Channel Usage	2-30

CHAPTER 3 - SOFTWARE CONFIGURATION

3.1	Introduction	3-1
	3.1.1 Conventions	3-1
3.2	Operation with DOS.....	3-2
	3.2.1 Configuration Options	3-3
	3.2.2 Little Board/286 Utilities Overview	3-3
	3.2.3 Two Special Files	3-4
	3.2.4 The Ampro SETUP Function	2-4
3.3	CPU and BUS Speed Initialization	3-5
	3.3.1 Fixed Speed Operation (Using SETUP)	3-5
	3.3.2 Variable Speed Operation (Using SETSPEED)	3-5
3.4	Using a Parallel Printer	3-6
3.5	Using the Serial Ports.....	3-6
	3.5.1 Serial Port Initialization.....	3-6
	3.5.2 Serial Console Option.....	3-7
	3.5.3 Using a Serial Printer.....	3-8
	3.5.4 Using a Serial Modem	3-9
3.6	Using Floppy Drives	3-10
	3.6.1 Drive Parameter Setup.....	3-10
	3.6.2 Single-Floppy Configurations	3-10
3.7	Using SCSI Hard Disk Drives.....	3-11
	3.7.1 Drive Parameter Setup.....	3-11
	3.7.2 Preparation for DOS Use.....	3-12
3.8	AT Bus Controller Interfaced Hard Disk Drives	3-14

CHAPTER 4 - THEORY OF OPERATION

4.1	Introduction	4-1
4.2	Overall Architecture.....	4-3
	4.2.1 Standard AT System Functions	4-3
	4.2.2 Unique Functions.....	4-3
	4.2.3 System Memory Map	4-4
	4.2.4 System I/O Map.....	4-5
4.3	AT Motherboard Logic	4-6
	4.3.1 CPU.....	4-6
	4.3.2 CPU and Bus Speed Control.....	4-6
	4.3.3 ROM-BIOS Sockets	4-7
	4.3.4 Onboard DRAM Memory.....	4-7
	4.3.5 Interrupt Controllers	4-8
	4.3.6 DMA Controllers	4-8
	4.3.7 Programmable Timers	4-10
	4.3.8 Keyboard Interface	4-10
	4.3.9 Speaker Interface	4-11
	4.3.10 Battery-Backed Clock.....	4-11
	4.3.11 AT Expansion Bus	4-11
4.4	Floppy Disk Controller	4-12
4.5	Serial/Parallel Controller.....	4-13
	4.5.1 Serial Ports.....	4-13
	4.5.2 Parallel Printer Port.....	4-14
4.6	Small Computer System Interface (SCSI)	4-16
4.7	Byte-Wide Memory Sockets	4-19
4.8	Configuration EEPROM	4-20

APPENDIX A - SLOT BOARD/286 UTILITIES

SCSI-ID	SCSI Initiator ID reporting
SCSI-VER	SCSI version reporting
SCSICOMP	SCSI device block compare
SCSICOPY	SCSI device block copy
SCSIFMT	SCSI hard disk formatter
SCSIPARK	SCSI hard disk park utility
SCSITool	SCSI device exerciser/debugger
SETSPEED.....	Bus and CPU half speed control
SETUP	Board configuration utility

FIGURES AND TABLES

FIGURES

Figure 1-1	Mechanical Dimensions.....	1-5
Figure 2-1	Connector and Jumper Locations	2-2
Figure 2-2	Byte-Wide Socket Jumpering	2-9
Figure 4-1	Little Board/286 Block Diagram	4-2

TABLES

Table 2-1	Summary of Connector Usage.....	2-1
Table 2-2	Configuration Jumper Summary.....	2-3
Table 2-3	"Soft" Configuration Options	2-4
Table 2-4	Power Connector (J1)	2-5
Table 2-5	Onboard DRAM Memory Size Jumpering	2-7
Table 2-6	Byte-Wide Device Compatibility	2-8
Table 2-7	U32 and U33 Device Jumpering.....	2-9
Table 2-8	Video Mode Jumper (W22).....	2-10
Table 2-9	Utility Connector (J4).....	2-12
Table 2-10	Keyboard Connector (J5).....	2-14
Table 2-11	Parallel Printer Connector (J6)	2-15
Table 2-12	Parallel Printer Port Configuration	2-15
Table 2-13	Serial Port Connectors (J2,J3)	2-17
Table 2-14	Supported Floppy Formats	2-17
Table 2-15	Dual-Speed Drive Option	2-18
Table 2-16	Floppy Disk Interface Connector (J8)	2-19
Table 2-17	SCSI Interface Connector (J7).....	2-20
Table 2-18A	AT Expansion Bus Connector, A1-A32 (J9).....	2-27
Table 2-18B	AT Expansion Bus Connector, B1-B32 (J9)	2-28
Table 2-18C	AT Expansion Bus Connector, C0-C19 (J10)	2-29
Table 2-18D	AT Expansion Bus Connector, D0-D19 (J10).....	2-29
Table 2-19	Interrupt Channel Assignment.....	2-30
Table 2-20	DMA Channel Assignment.....	2-30
Table 4-1	Little Board/286 Memory Map.....	4-4
Table 4-2	Little Board/286 I/O Map	4-5
Table 4-3	Interrupt Channel Assignment.....	4-8
Table 4-4	DMA Channel Assignment.....	4-8
Table 4-5	Timer Assignment	4-10
Table 4-6	Parallel Port Register Bits.....	4-15

CHAPTER 1

INTRODUCTION

1.1 GENERAL DESCRIPTION

The Little Board/286 provides system designers with a high performance, *AT-compatible* single board microcomputer system that requires no more space than a half height 5-1/4" disk drive. This unique *single board system* is functionally equivalent to a normal AT motherboard and three or four expansion cards.

The Little Board/286 is ideally suited for embedded microcomputer applications where IBM PC/AT software, hardware, and bus compatibility are required and where low power consumption (under 9 watts), small size (5.75" x 8" x 1.1"), wide temperature range (0-70° C) and high reliability are critical. In addition, its ability to use rugged media like EPROM and nonvolatile RAM instead of disk drives makes the Little Board/286 especially well suited for use in harsh environments.

Typical applications for the Little Board/286 include:

- Data acquisition & control
- Portable instruments
- Protocol conversion
- Telecommunications
- Security systems
- Intelligent terminals
- Medical instruments
- Diskless workstations
- Point-of-sale terminals
- Network servers
- Distributed processing
- SCSI device control
- Vending machines
- Robot and machine control

The Little Board/286 includes a powerful 12 MHz or 16 MHz 80286 microprocessor, up to 4M bytes of onboard DRAM, and a full complement of AT compatible controllers. Peripheral controllers are included on the board for standard serial, printer, keyboard, speaker, and floppy interfacing. In addition, two industry standard system expansion buses -- a PC/AT Expansion Bus and a Small Computer System Interface (SCSI) -- offer system expansion flexibility.

A PC/AT bus expansion connector is provided for Ampro MinModule expansion boards which attach directly to the Little Board/286. When installed, an onboard MinModule fits entirely within the Little Board/286's outline dimensions.

Standard PC and AT bus plug-in cards can be added to the Little Board/286 using ribbon cable from the PC/AT bus expansion connector. The innovative StackPlane expansion option permits a PC bus card to be mounted parallel to the Little Board/286 without additional hardware. Also available is a compact 2-slot passive backplane which accommodates two PC or AT bus cards.

Support for SCSI direct access devices as boot or data devices is included directly in the board's ROM-BIOS. This *SCSI/BIOS* feature allows a wide variety of hard disks, floppy drives, bubble memory drives, tape drives, and other SCSI devices to be used interchangeably, greatly enhancing the ease and flexibility of system integration, maintenance, and upgrades.

Sockets on the Little Board/286 allow onboard EPROM (up to 256K bytes) or battery-backed RAM (up to 64K bytes) to function as a *bootable* DOS compatible *Solid State Disk* (SSD) drive. SSD operation offers several key advantages, including improved speed, reliability, and ruggedness, as well as reduced power consumption. Ampro's optional SSD support software converts DOS based applications into EPROM format automatically, without the need for special programming techniques. In addition, Ampro's SSD Expansion Board, which matches the 5.75" x 8" form factor of the Little Board/286, can be used to add sixteen additional byte-wide device sockets for larger SSD's.

A key advantage of the AT-compatibility of the Little Board/286 is that it runs the thousands of software applications and utility packages developed for IBM's PC and PC/AT. Language compilers, debugging aids, and software support packages for graphics, windowing, multi-tasking, user interfaces, etc., are readily available. In addition, there is a large and growing selection of DOS programs targeted at industrial and commercial applications such as communications, data acquisition and control, terminal emulation, protocol conversion, etc.

The Little Board/286 is a member of Ampro's growing family of IBM compatible *single board systems*. These products offer a broad range of price, performance, and feature alternatives in a consistent physical and functional format

1.2 LITTLE BOARD/286 FEATURES

A Complete AT Compatible System on One Board

- All the functions of a motherboard and 4 expansion cards in the space of a half-height 5-1/4" disk drive!
- Runs standard IBM PC and PC/AT software, including:
 - Disk Operating Systems
 - Languages (C, Pascal, Fortran, Basic,...)
 - Industrial and commercial applications software
- Powerful 12 MHz or 16 MHz 80286 CPU
- Standard AT DMA, timers, and interrupt controllers
- Complete onboard system memory: 512K, 1M, 2M, or 4M bytes onboard DRAM, and up to 320K bytes EPROM
- Complete set of AT-compatible peripheral ports and controllers:
 - Mini/Micro floppy controller
 - 2 RS232C serial ports, parallel printer port, keyboard port, and speaker port
 - Optional multi-mode video controller modules
 - Battery-backed real time clock
- Onboard expansion module option:
 - Three choices of onboard display controllers (EGA/CGA/MDA/Hercules CRT, CGA/MDA/Hercules CRT, CGA compatible LCD)
 - Future onboard modules available (contact factory)
- Industrialized ROM-BIOS, with support for a wide variety of SCSI devices (hard disk, tape, bubble memory,...) as well as on- and off-board bootable Solid State Disk

Ideal for Embedded Applications

- Low power (mostly CMOS):
 - Uses less than 9 watts of power!
 - Single supply operation (+5V only)
- Wide operating temperature range (0-70° C)
- Reliable:
 - Low component count (30 active components, plus RAM)
 - No backplanes or edgecard connectors required
- Onboard bootable *Solid State Disk drive* (EPROM/NOVRAM) option allows diskless DOS operation
- Same form factor and mounting dimensions as a 5-1/4" disk drive

Two Industry Standard Expansion Buses

- AT bus (header) for connection of standard IBM PC and PC/AT expansion cards
- SCSI bus for addition of Disk/Tape/Optical/Bubble drives, Scanners, and other peripheral devices

1.3 LITTLE BOARD/286 SPECIFICATIONS

CPU

- 80286 microprocessor
- CPU clock rate: 12 or 16 MHz
- Half speed option for CPU and/or expansion bus

Onboard Memory

- 512K, 1M, 2M, or 4M bytes DRAM with parity
- Award ROM-BIOS with Ampro extensions
- Two spare 28-pin byte-wide sockets, located at D0000h and D8000h, usable with:
 - 27256 32K byte EPROM's
 - Intel 27011 128K byte page-addressed EPROM's
 - 62256 32k byte static RAM's
 - Dallas DS1230 32K byte nonvolatile RAM's
 - Dallas DS1217 series NOVRAM cartridges
- 1K bit serial EEPROM, with 512 bits for OEM use

AT-Compatible Controllers

- 2 fully compatible RS232C serial ports
- Parallel printer port with bidirectional data lines
- Standard AT keyboard port
- Speaker port with 0.1 watt output drive
- Standard battery-backed real time clock with 10 year internal lithium battery
- AT-compatible mini (5-1/4") and micro (3-1/2") floppy controller:
 - 2 drive selects, 1-2 sided, 250/500K bps data rates
 - BIOS supports all standard formats (360K/720K/1.2M/1.44M)
 - Reliable all-digital phase-locked loop & write precompensation
 - Disk change sense line support
- Provision for 80287 math coprocessor option

SCSI Interface

- Full ANSI X3.131 (SCSI) compatible
- Uses high performance Adaptec AIC6250 SCSI controller
- Usable as general purpose programmed I/O lines (48 mA sink)
- Over 1.5 megabyte/sec data throughput

Physical

- 8.0 x 5.75 x 1.1 inches (5-1/4" disk drive form factor)
- Mounting slots match 5-1/4" disk drive bottom holes
- Provision for one onboard Ampro MiniModule
- Same DC connector as 5-1/4" disk drives
- Power requirement (with 512K bytes RAM):
 - 80C286 CPU: +5V +/- 5% at 1.6 A (typ.)
 - 80L286 CPU: +5V +/- 5% at 1.8 A (typ.)
- 8-layer PCB, with 2 ground layers for low noise
- Uses latest surface mount technology
- Operating environment:
 - 0-70° C (assuming adequate airflow)
 - 5-95% relative humidity (non-condensing)
- Storage temperature: -55° to +85° C

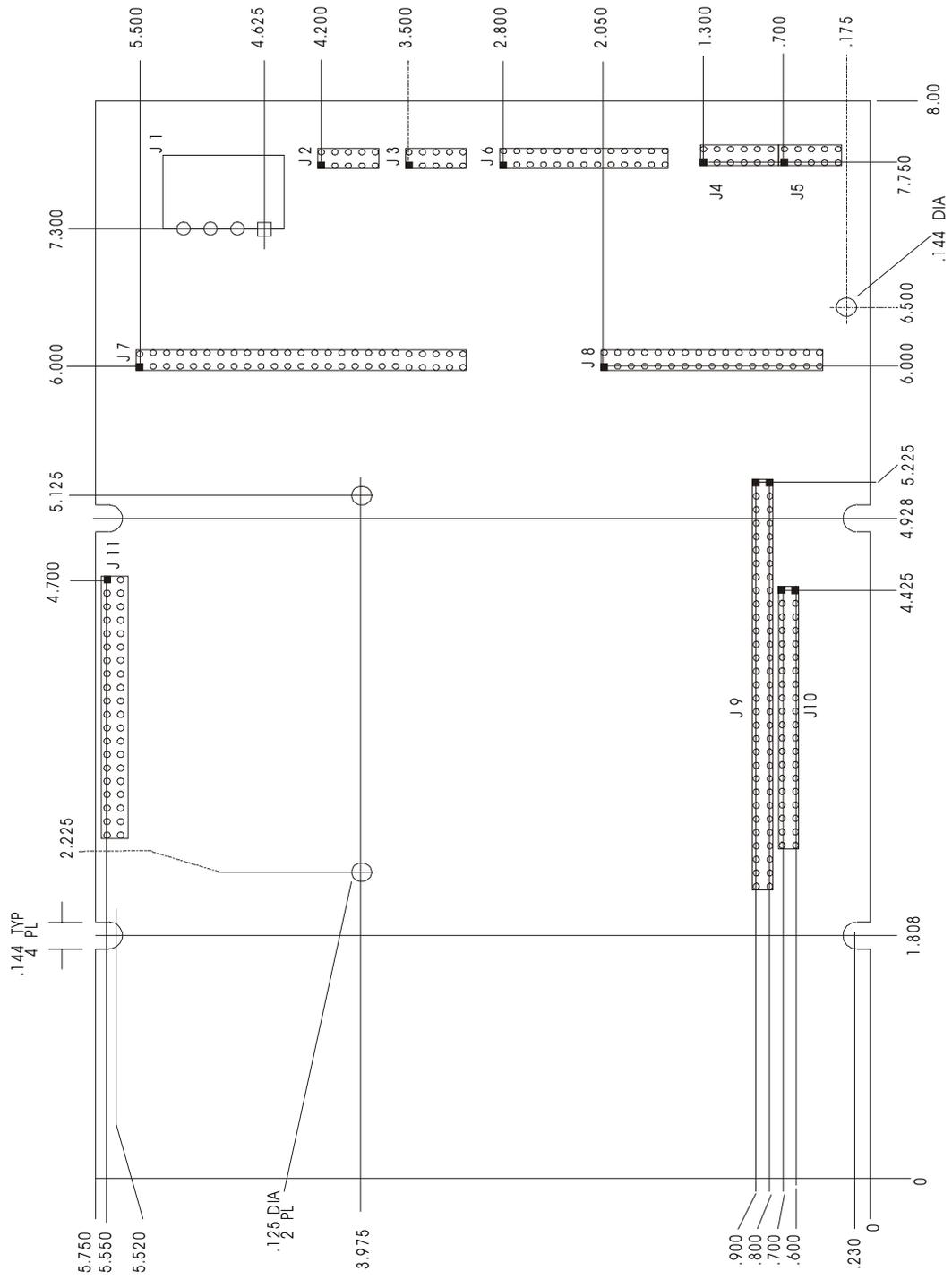


Figure 1-1. Mechanical Dimensions

CHAPTER 2

HARDWARE CONFIGURATION

2.1 INTRODUCTION

This chapter describes what is required to integrate the Little Board/286 with a wide variety of external devices, including keyboards, printers, modems, floppy and hard disk drives, and monitors. Information is provided regarding the board's connector signals and pinouts, external device requirements, interconnection cable wiring, and board configuration.

Also covered in this chapter are the configuration and initialization requirements of standard and optional onboard devices. These include both the standard items such as the DRAM memory, battery-backed real time clock, and speed control functions, and also several optional additions such as the 80287 math coprocessor, EPROM and NOVRAM "Solid State Disk drives", and a variety of onboard expansion modules including the Ampro MiniModule video controllers.

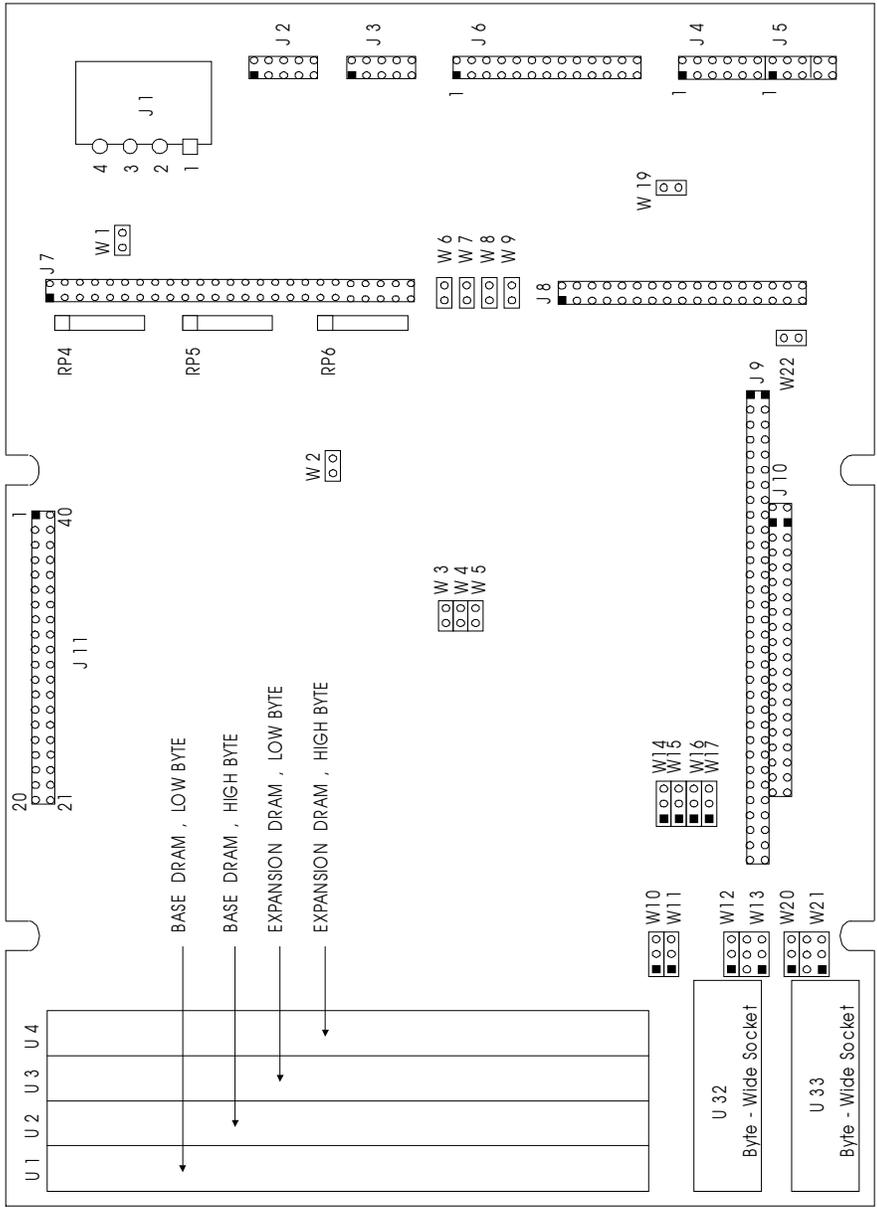
It is important to note, however, that many of the board's functions are software rather than hardware controlled. Therefore, many of the features and functions described in this manual assume the presence of the board's standard ROM-BIOS for their operation. Refer to Chapter 3 for typical software configuration information, and to Chapter 4 for technical details regarding the board's hardware and software functions.

2.1.1 Interface Connectors

Figure 2-1 shows the location of the board's interface connectors (J1-J11) and configuration jumper locations. Table 2-1 summarizes the use of each of the board's connectors. Connector pinouts and signal definitions are provided in the sections of this chapter that cover each connector interface. Typical interface cable construction data is also included in the description of each connector interface.

Table 2-1. Summary of Connector Usage

Connector	Function	Size
J1	Power Input	4-pin
J2	Secondary Serial	10-pin
J3	Primary Serial	10-pin
J4 (*)	Utility	12-pin
J5 (*)	Keyboard	6-pin
J6	Printer	26-pin
J7	SCSI	50-pin
J8	Floppy	34-pin
J9	AT Bus	64-pin
J10	AT Bus	40-pin
J11	Math Option	40-pin
(*) J4 and J5 are combined on a single 18-pin array		



2 4 6
 1 3 5
 NOTE: W13 & W21 JUMPER NUMBERING

Figure 2-1. Connectors and Jumpers

The board's I/O connectors consist of dual-row male header connectors, of varying lengths. Each of these headers can be mated with either flat ribbon (IDC) or discretely-wired connectors. J2, J3, J4, J5, and J6 have been placed together at one end of the board so that a single header plug can mate with one or more of these connectors as a group.

In some applications it is preferable to plug a specialized PC board assembly onto several or all of the I/O connectors (J2-J8). This allows you to eliminate cables, meet packaging requirements, add EMI filtering, or in other ways customize the installation of the Little Board/286 in your application. In addition, two dual-row male headers provide the PC and AT expansion bus for onboard MiniModule expansion.

2.1.2 Jumper Configuration Options

For normal AT compatible operation, no special jumpering of the Little Board/286 is required. As shipped from the factory, the board can be connected to appropriate peripherals and operated immediately. Typically, the only jumpers you will need to be concerned with are those used for setting the amount of onboard DRAM memory and for specifying whether the powerup video mode is monochrome or color. Table 2-2 summarizes the board's jumper options; they are each discussed in detail in this chapter.

Each of the board's jumper options is referred to by a "W" number (i.e. "W1"). Jumpers occur in 2-pin groups, 3-pin groups, and 6-pin groups. The pin numbering convention for all two- and three-pin jumper groups is as follows: assuming that the card edge connectors are at the bottom and the I/O connectors are at the right-hand side (Figure 2-1), "pin 1" is *topmost* on a vertically oriented line of jumper pins, or *leftmost* on horizontally oriented pins.

In the case of 2-pin jumper groups, the pair of jumper pins is either shorted or unshorted by installing or removing a shorting block. For three-pin jumper options, there are three choices: shorting pin 1 to pin 2; shorting pin 2 to pin 3; or shorting neither.

Two jumper groups (W13 and W21) each have 6 pins. The pin numbering and settings for these four 6-pin jumper groups are discussed later in this chapter.

Table 2-2. Configuration Jumper Summary

JUMPER GROUP	FUNCTION
W1	SCSI termination power
W3,4,5	DRAM memory quantity
W7	Parallel port address option
W12,13,20,21	Byte-wide memory socket setup
W19	Dual speed floppy enable
W22	Video (mono or color)
W2,6,8-11, 14-17	Factory assembly options. (Do not alter)

2.1.3 "Soft Configuration" Options

In addition to the various configuration jumper options, there are a number of system configuration parameters, which are initialized by the ROM-BIOS based on the contents of two different nonvolatile memory devices on the board -- an AT compatible nonvolatile static RAM within the real time clock device, and a unique 1024 bit configuration EEPROM).

The combination of these two devices will be referred to as the board's nonvolatile *Configuration Memory*. Through the use of this Configuration Memory function, the number of jumper configuration options has been minimized and board setup had been simplified. A configuration utility, SETUP, is provided in the ROM-BIOS and can be accessed either with a "hot-key" entry

(CTRL-ALT-ESC) or through the SETUP program on the Little Board/286 Utilities diskette.

Table 2-3 lists the main functions that are controlled by the contents of the nonvolatile Configuration Memory.

Table 2-3. "Soft" Configuration Options

■	Floppy drive quantity and type
■	Hard disk drive quantity and type
■	Serial port default initialization
■	Battery-backed clock date and time
■	Video controller type
■	Enable/disable functions for
	-- SCSI/BIOS services
	-- Parallel port
	-- Each serial port
	-- Each byte-wide socket
■	CPU and bus half-speed options

As shipped from the factory, the Little Board/386 can be booted from a bootable DOS diskette in a 5-1/4" 360K byte floppy drive. You can use the "hot-key" entry into SETUP to set system parameters prior to boot. Using SETUP, you can also load system setup parameters from a disk file, which is useful when configuring systems in a production environment. Refer to Chapter 3 and Appendix A for detailed information on the functions and use of SETUP which is used to initialize these parameters in the board's nonvolatile Configuration Memory devices.

2.2 DC POWER INPUT

The power connector, J1, has a pinout which is identical with that of power connectors on nearly all 5-1/4 inch disk drives, except that only the +5 volt and ground connections are used. *Note that pin 1 on J1 is reversed from that of the header connectors on the Little Board/286.* Refer to Table 2-4 for power connections and mating connector information, and Figure 2-2 for typical connector wiring.

Caution

Be sure the power plug is correctly wired before applying power to the board! See figure 2-2.

Table 2-4. Power Connector (J1)

PIN	SIGNAL NAME	FUNCTION
1	-----	(not used)
2	Ground	Ground return
3	Ground	Ground return
4	+5VDC	+5VDC +/- 5%
Housing: AMP 1-480424-0		
Pins: AMP 60619-1		

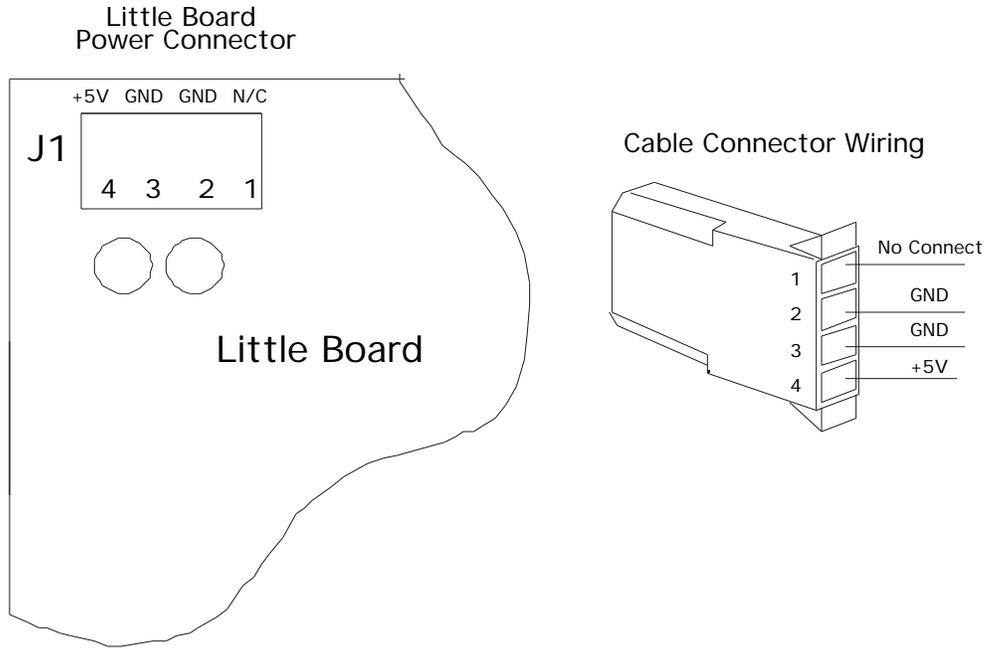


Figure 2-2. Power Connector Wiring

The precise power requirements of the Little Board/286 depend on a number of factors, including what functions are present on the board (e.g., expansion DRAM, byte-wide memory devices, 80287/80C87 math coprocessor), whether the SCSI bus is terminated, and what peripherals are connected to the board's I/O ports. For example, AT keyboards draw their power from the keyboard connector on the board, and therefore add to the total power drawn by the board from its power supply. Similarly, boards installed in the onboard expansion module position or connected to the AT Expansion Bus may draw their power through the Little Board/286.

Assuming a CMOS 80C286 processor is in use, a fully populated Little Board/286 having 1 megabyte of SIMM RAM installed (*without the math coprocessor or MiniModules installed*) requires approximately 8 watts of power (1600 mA). Note that the SCSI bus terminators consume approximately .8 watts of power by themselves, so it is important not to have them plugged into the board if SCSI is not needed. Operating the CPU with the *half-speed* option enabled will also reduce power consumption.

2.3 ONBOARD DEVICE OPTIONS AND CONFIGURATION

This section includes the configuration and installation of *onboard devices* and options. These include DRAM and byte-wide memory devices, nonvolatile RAM cartridges, math coprocessor, and a variety of onboard expansion MiniModules including the Ampro MiniModule video controllers.

2.3.1 CPU and Bus Half-Speed Options

The speed of the CPU or of the AT Expansion Bus can be switched to one-half of the normal clock speed under software control. This is set using the Ampro SETUP or SETSPEED utilities, as described in Chapter 3.

When the CPU speed is set to "half-speed," then the AT Expansion Bus speed will also run at half-speed, since it cannot exceed that of the CPU. In addition, it should be noted that the DMA controller runs at one-half of the CPU rate, so its speed is linked to the setting of the CPU half-speed option, not the bus half-speed option.

Reducing the CPU speed can be useful in lowering system power consumption for battery powered applications, or in increasing the system operating temperature range by reducing the amount of heat generated by key system components such as the DRAM memory.

Operation of the AT Expansion Bus at half speed may be required to allow the use of plug-in cards which were designed to be used in older PC compatible systems with lower system clock rate, or to permit the use of longer bus expansion cables.

The initial state of the CPU and bus half-speed options is determined by parameters stored in the nonvolatile Configuration Memory, which you can set using the Ampro SETUP utility. During system operation, the values of CPU and bus half-speed options can be altered dynamically as desired, using the Ampro SETSPEED utility. Refer to the information on the use of the SETUP and SETSPEED utilities in Chapter 3 and Appendix A for details.

2.3.2 DRAM Memory

The board has positions for up to four single in-line memory modules (SIMM's), which are organized as either 256K x 9 bits or 1M x 9 bits depending on the version of the Little Board that you are using. At least two SIMM's are required for operation, and the board must be jumpered to indicate whether two or four are being used. Note that when only two SIMM modules are used, they must be inserted in positions U1 and U2, with U3 and U4 left empty.

An additional jumper option, available for systems with 1 megabyte or more of DRAM memory installed, allows the memory above the normal 640K byte system memory area to be used as "extended memory" if desired. When the extended memory option is enabled, it begins at address 100000h.

Three jumpers (W3, W4, W5) must be set as shown in Table 2-5, to indicate the number (2 or 4) and type (256K or 1M) of *onboard* SIMM DRAM memory that is installed. Note that *all four SIMM DRAM modules must be the same type* (i.e. 256K or 1M).

Table 2-5. Onboard DRAM Memory Size Jumpering

SIMM QTY.	SIMM TYPE	LOCATION	W3	W4	W5	SYSTEM MEMORY	EXTENDED MEMORY
2	256K	U1,2	open	short	short	512K	0K
4	256K	U1,2,3,4	short	open	short	640K	0K
4	256K	U1,2,3,4	open	short	short	640K	384K
2	1M	U1,2	short	short	open	512K	0K
2	1M	U1,2	open	short	open	640K	0K
2	1M	U1,2	short	open	open	640K	1408K
4	1M	U1,2,3,4	open	open	open	640K	3456K

On a 12MHz system, use 120 nS (or faster) SIMMs. On a 16MHz system, use 100 nS (or faster) SIMMs.

An additional set of six jumpers (W10, W11, W14-W17) is factory set according to both the board's CPU clock rate and the DRAM memory's access time requirement. These are factory settings, and should not be altered.

2.3.3 Math Coprocessor

Signals for interfacing with a standard 80287/80C287 floating point coprocessor are provided at J11, which is identical in pinout with the normal 40-pin DIP socket, but on .1" row-to-row spacing. An *80287/80C287 Carrier* assembly, available from Ampro, allows you to install a 80287 math coprocessor at this location.

The required speed of the 80287/80C287 math coprocessor is 2/3 the clock rate of the CPU. Therefore the 12 MHz Little Board/286 requires an 8 MHz device (80287-8), while the 16 MHz board requires a 10 MHz device (80287-10).

Note

Be sure to install the 80287 on the 80287 Carrier with the IC's pin 1 correctly positioned in pin 1 of the IC socket; be sure to install the 80287 Carrier on the Little Board/286 so that it overlaps the board (pin 1 of J11 mates with 80287 Carrier socket receptacle 1).

When installed on J11, the 80287 math coprocessor is compatible with that of a standard AT computer in all operational details.

Note

The NMOS 80287 math coprocessor generates a large amount of heat relative to that produced by the Little Board/286 itself. Be sure to provide adequate clearance and air flow in the vicinity of the math coprocessor if the NMOS device is used.

2.3.4 Byte-Wide Memory Sockets

The Little Board/286 has two onboard byte-wide memory device sockets that can accommodate a variety of EPROM and nonvolatile RAM (NOVRAM) devices, used for simple program storage or as "Solid State Disk" (SSD) drives.

Some of the byte-wide devices which can be used in these sockets are shown in Table 2-6. The NOVRAM cartridges described in the table connect with a short ribbon cable (Dallas P/N DS9000) to a byte-wide socket: U32 or U33. Devices used in these sockets must be 300 nS (or faster), on both 12 MHz and 16 MHz boards.

Table 2-6. Byte-Wide Device Compatibility

DEVICE TYPE	SIZE	PART NUMBER
Normal EPROM (*)	8K bytes	Intel 2764
Normal EPROM (*)	16K bytes	Intel 27128
Normal EPROM	32K bytes	Intel 27256
Page-addr EPROM	128K bytes	Intel 27011
Static RAM (*)	8K bytes	Hitachi 6264
Static RAM	32K bytes	Hitachi 62256
28-pin NOVRAM	32K bytes	Dallas DS1230
NOVRAM Cartridges	32K-512K bytes	Dallas DS1217
(*) Note: 8K byte EPROM's and static RAM's will appear four times, and 16K byte EPROM's two times, within the 32K address window of the socket.		

Direct Program Access

The memory devices installed in U32 and U33 can be accessed directly by your application software, provided the program knows about them. Note that the byte-wide sockets must be enabled prior to use using the SETUP configuration program (see Chapter 3) or under control of your application program (see Chapter 4).

The byte-wide memory sockets reside at the following memory addresses:

- U32 is at D0000h
- U33 is at D8000h

Through a technique known as a "ROM-BIOS extension", it is possible to have the system run its entire application from byte-wide socket memory devices, instead of from disk drives. This is discussed in Ampro application note AAN-8702. Be sure the sockets are enabled, as mentioned above, if they are to be used with ROM-BIOS extensions.

Solid State Disk (SSD) Drives

The optional Ampro Solid State Disk (SSD) Support Software, allows you to configure a Little Board/286 system to boot, operate, and even store data using one or two EPROM and/or NOVRAM solid-state "drives" under control of the DOS (version 3.3 or later) operating system. No custom programming is required, other than use of the development tools provided in the SSD Support Software.

Using the Ampro SSD software, the byte-wide sockets can serve as individual SSD drives, or they can be combined into one larger drive. SSD drives can be used along with normal (floppy and hard disk) drives. Ampro's SSD Expansion Board provides additional SSD sockets.

Byte-Wide Socket Configuration

If the byte-wide memory sockets are used, they must be configured by means of jumpers for the device type in use. W12 and W13 are associated with U32, while W20 and W21 are associated with U33.

The supported memory devices and corresponding jumper settings are shown in Table 2-7. Note that W13 and W21 each consist of six jumper pins, numbered "1" through "6", as shown in Figure 2-3.

Table 2-7. U32 and U33 Device Jumpering

SIZE	DEVICE TYPE	P/N	U32		U33	
			W12	W13	W20	W21
8K	Normal EPROM	2764	1/2	1/2	1/2	1/2
16K	Normal EPROM	27128	1/2	1/2	1/2	1/2
32K	Normal EPROM	27256	1/2	5/6	1/2	5/6
128K	Page-addr EPROM	27011	1/2	3/4	1/2	3/4
32K	28-pin SRAM	62256	2/3	3/4	2/3	3/4
32K	28-pin NOVRAM	DS1235x	2/3	3/4	2/3	3/4
32K to 512K	NOVRAM Cartridge	DS1217x	2/3	3/4	2/3	3/4

The required access time for devices plugged into U32 and U33 is 300 nS or faster.

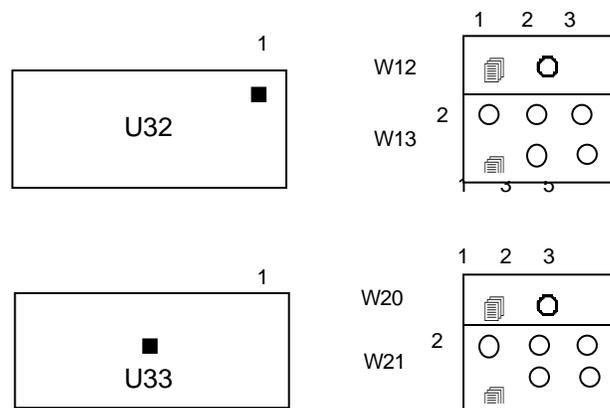


Figure 2-3. Byte-Wide Socket Jumpering

In addition to having its device configuration jumpers set, each socket must be *enabled* or *disabled* via a parameter within the board's nonvolatile Configuration Memory. The Ampro SETUP utility offers a convenient means to do this. SETUP resides in the ROM-BIOS and can be accessed either with a "hot-key" entry (CTRL-ALT-ESC) or through the SETUP program provided on the Little Board/286 Utilities diskette.

As mentioned previously, the required access time for devices plugged into U32 and U33 is 300 nS (or faster), on both 12 MHz and 16 MHz boards.

Note

When a byte-wide memory device is enabled, its memory address space is unavailable for use by other devices on the AT bus. A byte-wide socket must be marked as "disabled" by means of the Ampro SETUP utility before its memory space can be used for other purposes.

2.3.5 Battery-Backed Clock

The Little Board/286 provides, as a standard feature, an AT compatible battery-backed real time clock (with CMOS RAM). Included within the clock module is a lithium battery, so no external backup battery is required. At ten-year intervals, the entire device should be replaced. Oversized solder pads have been provided to facilitate this maintenance operation.

The real time clock date and time values and various parameters within the nonvolatile Configuration Memory are initialized at the factory for a standard configuration. The date and time are also set at the factory, but may not be set for your time zone. Use the Ampro SETUP utility to change these values as needed (see Chapter 3).

2.3.6 Video Display Mode Jumpering

Regardless of what type of video display controller is used -- whether by means of an onboard video expansion module or by means of a controller connected externally on the AT Expansion Bus -- a jumper (W22) on the Little Board/286 must be properly configured to indicate to the ROM-BIOS whether the board's powerup (or reset) video state should be *color* or *monochrome*. W22 is jumpered as indicated in Table 2-8.

Table 2-8. Video Mode Jumper (W22)

VIDEO MODE	W22
Mono (MDA or Hercules)	open
Color (CGA, EGA, VGA)	short

Note

When an EGA or VGA controller's switches or jumpers are configured for a monochrome mode, W22 should be set for "color" rather than "mono".

The function of W22 is replicated on the Utility Connector (J5) pins 9 and 10. This was done to facilitate the use of an external mono/color configuration switch, if needed. Pins 9 and 10 of J5 are not normally populated with connector pins, so you must add pins in that location if you wish to take advantage of the option.

In addition to setting W22, the desired bootup video mode must also be specified in the board's nonvolatile Configuration Memory. The Ampro SETUP utility offers a convenient means to do this. SETUP resides in the ROM-BIOS and can be accessed either with a "hot-key" entry (CTRL-ALT-ESC) or through the SETUP program provided on the setting of W22 and the parameters in the Configuration Memory do not match, an error message will be displayed during the system boot process.

Note

When an EGA or VGA controller is configured to bootup in a monochrome or CGA mode, the video controller type parameter specified using the SETUP utility should still be set for "EGA" or "VGA".

Also, don't forget to set any appropriate switches or jumpers on the video display controller for the desired video mode. Refer to your video controller's technical manual.

2.4 PERIPHERAL CONNECTIONS AND CONFIGURATION

This section covers the interface requirements of the board's external device interface connectors. Information is provided on interface and device characteristics, connector pinouts, signal definitions, jumper configuration, and nonvolatile Configuration Memory setup.

2.4.1 Utility Connector

A single 12 pin connector (J4) provides five functions: Speaker, Reset, Power Indicator LED, auxiliary power to the AT Expansion Bus connector (J9), and a Power Good status signal. The pinout and signal definitions of the Utility Connector are shown in Table 2-9. Note that the power connections indicated are not required for operation of the Little Board/286, and are only used to provide power for cards connected to the board's AT Expansion Bus connector.

Table 2-9. Utility Connector (J4)

PIN	SIGNAL NAME	FUNCTION
1 2	Speaker + Speaker -	5 volts through 33 ohms audio signal
3 4	Ground Reset	To one side of Reset button To other side of Reset button
5 6	LED Cathode LED Anode	Ground return Current source (+5V through 330 ohms)
7 8 9 10	Ground +12V power -5V power -12V power	Ground return To J9 pin B9 To J9 pin B5 To J9 pin B7
11 12	Ground POWER GOOD	Ground return Power supply status

Speaker

The board provides an AT compatible speaker output on two pins (1,2) of the Utility Connector. The speaker signal is buffered by a transistor amplifier to provide approximately .1 watt of audio power to an external speaker. Typically, a small general purpose 2 or 3 inch diameter permanent magnet speaker with an 8 ohm voice coil is used.

Speaker tones can be generated either by directly *bit-banging* the speaker control port or by setting Timer 2 to a desired frequency tone and enabling the tone with an enable port. Refer to an AT technical reference manual for custom programming information.

Pushbutton Reset

Two pins (3,4) of the Utility Connector provide connections for an external normally-open momentary switch to manually assert a system reset signal. Pin 4 is the reset (active low) input, and pin 3 is the logic ground.

Power Indicator LED

Two pins (5,6) of the Utility Connector provide connections for an external power indicator LED device. 15 mA of sourced current is available (+5V through 330 ohms). Pin 5 goes to the LED's cathode, and 6 to its anode.

AT Bus Power Options

Four pins (7,8,9,10) of the Utility Connector allow the connection of an external source of +12VDC, -5VDC, and -12VDC power to the board's AT bus (J9) for use by plug-in expansion cards. The Little Board/286 does not require these voltages.

Power Good Signal

Two pins (11,12) of the Utility Connector provide connections for a *power good* signal from power supplies which offer this feature. The purpose of this signal is hold the system in a reset state when power is not within the proper range. This signal connects to the board's reset circuitry, and is functionally equivalent to the Pushbutton Reset (pin 4). This input has a TTL level, Schmitt trigger conditioned input buffer, and requires an active low input of 1 second minimum duration for the system to reset reliably.

2.4.2 Keyboard

An AT compatible keyboard can be connected to the board's keyboard port via the first five pins (1-5) of connector J5. Normally, AT keyboards include a cable which terminates in a male 5-pin DIN plug for connection to an AT. Table 2-10 gives the board's keyboard connector pinout and signal definitions, and includes corresponding pin numbers of a normal AT DIN keyboard connector.

Pin 6 of J5 can be connected to a *keyboard inhibit switch*. When this pin is grounded, keyboard inputs will be ignored. This can be used with key operated switches for system security.

Four additional locations (7,8,9,10) of J5 are not ordinarily populated with connector pins. Of these, pins 7 and 8 are not connected, while pins 9 and 10 duplicate the function provided by W22, the video mode jumper (discussed earlier). If required, you can include the video mode jumper function in a cable assembly for external access.

Note

**“PC” keyboards are not compatible with “AT” keyboards,
and may not be used with this interface**

Table 2-10. Keyboard Connector (J5)

J5 PIN	SIGNAL NAME	DIN PIN
1	Keyboard Clock	1
2	Keyboard Data	2
3	(no connection)	3
4	Ground	4
5	Keyboard power	5
6	Keyboard inhibit	---
7 (*)	No connect	
8 (*)	No connect	
9 (*)	Same as W22-1	
10 (*)	Same as W22-2	
(*) = No connector pins.		

2.4.3 Parallel Printer Port

The board's serial/parallel controller option (U22) must be installed for this function to be available. The parallel printer connector (J6) has a pinout that allows the use of flat ribbon cable to a standard DB25S PC or AT rear panel printer connector. Table 2-11 gives the printer connector's pinout and signal definitions.

The table also indicates the pin number on the AT compatible DB25 printer connector which corresponds to each of the pins of the board's header connector. Note that the pin numbering convention for the board's header connector differs from that of DB25 connectors, even though direct connection with flat ribbon cable is possible. To clarify this, each signal's corresponding DB25 connector pin number has been included in the table.

Note

Be sure not to exceed 10 feet of total cable length between the board's connector and the printer. Beyond this distance, the signal cannot be expected to be reliable due to the limitations of the port's signal drivers.

The parallel printer port can be used either as a standard PC/AT printer port, or as a bidirectional data bus. Refer to Chapter 4 for additional information.

Table 2-11. Parallel Printer Connector (J6)

J6 PIN	SIGNAL NAME	FUNCTION	IN/OUT	DB25 PIN
1	-STROBE	Output data strobe	out	1
3	Data 0	LSB of printer data	i/o	2
5	Data 1	:	i/o	3
7	Data 2	:	i/o	4
9	Data 3	:	i/o	5
11	Data 4	:	i/o	6
13	Data 5	:	i/o	7
15	Data 6	:	i/o	8
17	Data 7	MSB of printer data	i/o	9
19	- ACK	Character accepted	in	10
21	BUSY	Cannot receive data	in	11
23	PAPER OUT	Out of paper	in	12
25	SEL OUT	Printer selected	in	13
2	-AUTOFD	Autofeed	out	14
4	-ERROR	Printer error	in	15
6	-INIT	Initialize printer	out	16
8	SEL IN	Selects printer	out	17
26	-POE	Port output enable	in	--
10,12 14,16 18,20 22,24	GND	Signal Ground	---	18-25

Printer Port Configuration

The jumper at W7 is provided to allow this port to be configured as either a *primary or secondary* AT compatible printer port. The jumper must be set as indicated in Table 2-12. When shipped, the board is configured for the *primary* option. In addition, the board's nonvolatile Configuration Memory contains a parameter which can be used to *disable* the port entirely. Use the Ampro SETUP utility to disable the parallel port, if desired.

Table 2-12. Parallel Printer Port Configuration

SELECTION	I/O ADDR.	INTERRUPT	W7
"Primary"	378-37Fh	IRQ7	open
"Secondary"	278-27Fh	IRQ5	short

As seen in Table 2-12, jumper option W7 allows a choice of two I/O port addresses for the printer port (378-37Fh or 278-27Fh).

Normally, DOS assigns the name "LPT1" to the *primary* device, and "LPT2" to the *secondary* printer device (if present). However, a scan is made for both choices (primary and secondary) and if only a *secondary* printer is detected, it will be assigned as the "LPT1". Therefore, to avoid confusion, the board's printer port should always be configured for the *primary* assignment shown in a *primary* parallel port interface is provided elsewhere in the system.

Note

There is no means to alter the printer port's interrupt, so although its I/O port address can be set to the standard LPT1 or LPT2 values, its interrupt remains at IRQ7. DOS does not use the interrupt, while some printer port control software does (such as print spoolers).

Bidirectional Parallel Port Usage

The parallel printer port can be used as a normal AT printer port, or it can be used for general purpose programmable I/O. By using the port's input and output handshake signals along with its 8-bit bidirectional data lines, many types of interfaces for specialized devices can be created, including writing data to LCD display panels, scanning keyboards, etc.

When configured as a bidirectional data bus, the parallel port cannot be used to support a standard printer.

Refer to Chapter 3 for typical system software configuration information, and to Chapter 4 for hardware details regarding nonstandard uses of this interface.

2.4.4 Serial Ports

The board's serial/parallel controller option (U22, U15, U16, U8, and associated circuitry) must be installed for the serial port function to be available. Two AT (and PC) compatible serial ports appear at header connectors J2 and J3. Each provides three RS232C output signals and five RS232C input signals, to facilitate connecting a broad range of serial devices to the Little Board/286 on either port (see Table 2-13).

Both ports support software selectable standard baud rates up to 19.2K bits/second (limited by the RS232C drivers), 5-8 data bits, and 1, 1.5, or 2 stop bits. The serial ports appear at the standard port addresses of 3F8-3FFh (primary port) and 2F8-2FFh (secondary port), using interrupts IRQ4 and IRQ3, respectively.

These two serial ports can be used for connection of printers, modems, terminals, remote hosts, or other serial devices. The main interface consideration is that the device be RS232C compatible. Many devices, such as printers and modems, require handshaking in one or both directions. Consult the documentation of the external serial device for information regarding possible handshake and other interface considerations.

Support in the ROM-BIOS allows a device connected via one of the board's serial ports to be used as the system "console" (keyboard and display), as an alternative to the use of a video controller, monitor, and keyboard. This is further discussed in Chapter 3.

Table 2-13 gives the connector pinout and signal definitions for both J2 and J3. In addition, the pins to which each signal should be wired for compatibility with the COM port external serial connectors of PC's (DB25) and AT's (DB9) are also indicated. The pinout of both serial connectors is such that when direct flat ribbon cable is used between either header and a DB9 connector, the resulting DB9 matches the standard AT serial expansion port connector pinout.

Pin 10 of each serial port connector is an *enable* signal which is normally left unconnected. When it is brought to a TTL high level, it disables the input signals. It is pulled low on the board by a 1K ohm resistor. This signal is intended for board test purposes only.

J3 is the *primary* serial port, and is supported by the board's ROM-BIOS as the DOS "COM1" device. The *secondary* serial port is J2; it is supported as the "COM2" device. However, it is possible to disable either or both serial ports using the Ampro SETUP utility. (Note: if you disable the primary port and enable the secondary port, then the secondary port gets installed as COM1 by DOS, unless there is a primary port present somewhere else in the system.)

Table 2-13. Serial Port Connectors (J2,J3)

PIN	SIGNAL NAME	FUNCTION	IN/OUT	DB25 PIN	DB9 PIN
1	DCD	Data Carrier Detect	in	8	1
2	DSR	Data Set Ready	in	6	6
3	RXD	Receive Data	in	3	2
4	RTS	Request To Send	out	4	7
5	TXD	Transmit Data	out	2	3
6	CTS	Clear To Send	in	5	8
7	DTR	Data Terminal Ready	out	20	4
8	RI	Ring Indicator	in	22	9
9	GND	Signal Ground	--	7	5
10	EN	Input Enable	in	--	--

2.4.5 Floppy Disk Interface

The board's floppy controller option (U34) must be installed for this interface to be available. The onboard floppy disk controller and ROM-BIOS support one or two floppy disk drives in any of the standard DOS formats shown in Table 2-14.

Table 2-14. Supported Floppy Formats

CAPACITY	DRIVE SIZE	TRACKS	DATA RATE	DOS VERSION REQUIRED
360K	5-1/4"	40	250 KHz	2.1+
1.2M	5-1/4"	80	500 KHz	3.0+
720K	3-1/2"	80	250 KHz	3.2+
1.44M	3-1/2"	80	500 KHz	3.3+

Floppy Drive Considerations

As indicated in Table 2-14, nearly any type of soft-sectored, single or double-sided, 40 or 80 track, mini or micro floppy disk drive is usable. Naturally, the higher the quality of the drives you use, the better your system's reliability. Here are some general considerations regarding the selection, configuration, and connection of floppy drives to the Little Board/286.

- **Drive Interface** -- The drives used must be compatible with the board's floppy disk connector signal interface, as described below. In general, any standard PC or AT compatible 5-1/4" or 3- 1/2" floppy drive will work fine.

- **Drive Quality** -- High quality, DC servo direct drive motor floppy disk drives are recommended.
- **Drive Combinations** -- Any combination of supported drives can be used.
- **Drive Select Jumpering** -- Both drives must be jumpered to the *second* drive select. Use a floppy cable with a segment of wires "twisted" as is the usual practice in PC and AT compatible systems.
- **Drive Termination** -- Resistive terminations should be installed only on the drive connected to the last interface cable connector (*farthest from the computer*).
- **Head Load Jumpering** -- When using drives with a Head Load option, jumper the drive for "head load with motor on" rather than "head load with drive select."
- **Drive Mounting** -- If the computer board is to be mounted directly on a disk drive, it is recommended that a thin metal shield be used between the disk drive and the computer board, to reduce the possibility of electromagnetic interactions. A drive and board mounting bracket is available as an option from Ampro.
- **Dual Capacity Drives** -- The Little Board/286's ROM-BIOS allows dual-capacity use of a 1.2 megabyte "high density" drive: i.e., 360 Kbyte diskettes can be read on a high density drive. However, it is not recommended that you write to 360K diskettes on the high density drive, as incomplete erasure may occur. (This recommendation applies to standard AT's as well.) In addition, there are two types of floppy drives that offer dual-capacity operation:

--**Standard AT drives:** the *data rate shifts* between 250K and 500K bits/second, but the drive's *rotational speed stays constant*.

--**Dual speed drives:** the data rate shifts between 250K and 300K bits/second, and the drive's *rotational speed changes* between 300 and 360 rpm as well.

A jumper (W19) is used to select between the two types of dual capacity drives. Refer to Table 2-15.

Table 2-15. Dual-Speed Drive Option

W19	FUNCTION
short	Dual-speed operation enabled
open	Dual-speed operation disabled

Floppy Interface Configuration

The desired configuration of floppy drives (number and type) must be properly initialized in the board's Configuration Memory. This is generally done using the Ampro SETUP utility. In addition, the dual-speed configuration jumper option (W19) should be set according to whether you wish *dual-speed* operation on dual capacity floppy drives (see Table 2-17).

Floppy Interface Connector

Table 2-16 shows the pinout and signal definitions of the board's floppy disk interface connector, J8. J8 is identical in pinout with the floppy connector of standard AT and PC floppy controller boards. Note that, as in a standard PC or AT, both floppy drives are jumpered to *the same* drive select -- as the "second" drive. The drives are uniquely selected as a result of a swapping of a group of seven wires (conductors 10-16) that must occur in the cable between the two drives. The seven-wire swap goes between the computer board and drive "A"; the wires to drive "B" are unswapped (or swapped a second time).

Table 2-16. Floppy Disk Interface Connector (J8)

PIN	SIGNAL NAME	FUNCTION	IN/OUT
2	-RPM/-RWC	Speed/Precomp	out
4	---	(Not used)	---
6	---	(Not used)	---
8	-IDX	Index Pulse	in
10	-MO1	Motor On 1	out
12	-DS2	Drive Select 2	out
14	-DS2	Drive Select 1	out
16	-MO2	Motor On 2	out
18	-DIRC	Direction Select	out
20	-STEP	Step	out
22	-WD	Write Delete	out
24	-WE	Write Enable	out
26	-TRK0	Track 0	in
28	-WP	Write Protect	in
30	-RDD	Read Data	in
32	-HS	Head Select	out
34	-DCHG	Disk Change	in
1-33 (all odd)		Signal Grounds	---

2.5 SCSI INTERFACE

The board's Small Computer System Interface (SCSI) controller option (U7 and associated components) must be installed for this interface to be available. The SCSI interface uses a 50-pin male header connector (J7) to interface with SCSI compatible peripherals. Table 2-17 shows the pinout and signal definitions of this interface. Refer to your SCSI device documentation, or the ANSI X3.131 SCSI specification (referenced in Chapter 4) for detailed information on the signal functions. Be sure that the maximum total SCSI bus cable length, from the board's connector to the most distant SCSI peripheral, does not exceed 18 feet.

Note

If the SCSI interface is not needed, you should either disable it by using the SETUP utility or by placing a shorting block on pins 1 and 2 of the SCSI connector (J7). When the shorting block is in place, the system bypasses any attempt to boot from the SCSI device, which can speed up the boot process.

Table 2-17. SCSI Interface Connector (J7)

PIN	SIGNAL	FUNCTION
2	-DB0	Data Bit 0 (LSB)
4	-DB1	" " 1
6	-DB2	" " 2
8	-DB3	" " 3
10	-DB4	" " 4
12	-DB5	" " 5
14	-DB6	" " 6
16	-DB7	" " 7 (MSB)
18	-DBP	Data Parity
26		Termination +5VDC
	TERMPWR	
32	-ATN	Attention
34	GROUND	Signal Ground
36	-BSY	Busy
38	-ACK	Transfer Acknowledge
40	-RST	Reset
42	-MSG	Message
44	-SEL	Select
46	-C/D	Control/Data
48	-REQ	Transfer Request
50	-I/O	Data Direction
25	---	Not used
1-49 (odd) 20,22,24 28,30	Ground	Signal Grounds

2.5.1 Normal Use of SCSI

The board's SCSI bus interface can serve a variety of purposes, including connection of hard disk controllers, tape controllers, text scanners, and printer and communications servers. Support is provided by the board's ROM-BIOS for booting of DOS from a SCSI device such as a hard disk. Virtually any device compatible with the SCSI Common Command Set for *direct access devices* can be used through the ROM-BIOS support. Some examples of direct access SCSI devices that can be accessed as DOS drives are hard disk drives, magnetic bubble drives, high density floppy drives, and some tape drives.

The Little Board/286 comes with a diskette containing software drivers and utilities for normal DOS operation using hard disks and other SCSI direct access devices. Included are a formatter, a partition editor, and a device driver capable of multiple drive and partition operation. Refer to Chapter 4 for information on software setup and drive preparation.

PC-DOS version 3.x requires that drives larger than 32 megabytes be partitioned into more than one "partition", while DR-DOS and PC-DOS 4.x allow a maximum partition size of 128M bytes. If desired under PC-DOS 3.x, each physical drive can have a maximum of up to four partitions, 32 megabytes or smaller, allowing the use of drives as large as 128 megabytes. PC-DOS 4.x and DR-DOS limit drive size to 512M bytes (four 128M byte partitions).

There are several other types of SCSI devices besides direct access devices. SCSI's additional device types include *sequential access devices* (e.g. tape), *printer devices*, *read-only devices* (e.g. CD-ROM), and *processor devices* (e.g. CPU's). In general, these other device types require special application programs, utilities, or driver software for use. For example, Ampro offers an optional tape backup program which can be used for DOS environment hard disk backup.

Hard disk support for operating systems other than DOS may or may not be automatically available through the board's ROM-BIOS resident hard disk driver. This depends on whether the operating system in question uses BIOS calls exclusively for the hard disk function, and whether the operating system has any special ROM-BIOS constraints (such as re-entrancy). Some operating systems -- multitasking ones in particular such as Unix -- interface directly with the hardware (e.g. attempt to directly program an AT compatible hard disk controller) and bypass BIOS. In those situations, the operating system must be modified to add an appropriate SCSI hard disk driver to use the to take advantage of the board's SCSI interface. An alternative is to use the Ampro MiniModule/ATDisk controller board interface, which provides compatibility with standard ATA-equipped hard disk drives. Refer to the Ampro MiniModule/ATDisk Technical Manual for additional information regarding the board's capabilities.

2.5.2 The Ampro SCSI/BIOS

Through the universal bus interface and command protocols offered by SCSI, it has become possible to connect a wide variety of mass storage devices to a computer system with virtually no changes to system software. To this, Ampro has added a further layer of universality: the *SCSI/BIOS*.

As its name implies, the SCSI/BIOS consists of a set of low level functions which have been incorporated into the ROM-BIOS to provide a hardware independent interface between system software and peripheral devices connected to the SCSI bus. The advantage of the Ampro SCSI/BIOS is that programmers can write software that uses devices connected to the SCSI bus without having to be concerned with the details of operating the board's SCSI bus interface. Also, software can be "ported" to the Little Board/286 from other hardware environments much faster due to the presence of the SCSI/BIOS, and with less difficulty and risk.

The board's SCSI interface, and the Ampro SCSI/BIOS within the board's ROM-BIOS are discussed in greater detail in Chapters 3 and 4. In addition application note AAN-8804, available from Ampro, provides details of the SCSI/BIOS functions available to programmers.

2.5.3 Simple Bidirectional I/O

When the board's SCSI bus is not needed for interfacing to standard SCSI devices, it can be used instead as a general purpose digital input/output port. The SCSI bus then becomes a set of programmable I/O signals, with TTL signal levels and open collector output drivers capable of sinking 48 mA. In this case, sixteen SCSI bus signals can be used; of these, nine are capable of bidirectional use, and seven are input-only.

The available SCSI port output signals can be "bit banged" (i.e. software activated) to generate a wide variety of signal timings and operations. For example, it is even possible to interface with optically isolated AC and DC input and output modules, and to a number of off-the-shelf data acquisition and control interfaces.

Ampro application note AAN-8801 discusses a typical non-SCSI application. Refer also to Chapter 3 for typical system configuration information, and to Chapter 4 for information about the board's SCSI bus interface.

2.5.4 SCSI Interface Configuration

Several items associated with the board's SCSI interface should be configured according to your system needs:

SCSI Bus Termination

Three 8-pin 220/330 ohm SIP resistor networks (RP4, RP5, and RP6) provide bus termination. The termination networks should be present on two, *and only two*, SCSI bus devices. An improperly terminated SCSI bus may interfere with normal system operation due to indeterminate signal levels.

Note

Be sure that the SCSI termination networks are oriented in their sockets so that pin 1 of each SIP network, generally marked with a dot, is at the top end of the board as shown in Figure 2-1.

Internally, pins 2-7 of the SIP networks have 220 ohm resistors connected to +5 volts (pin 8), and 330 ohm resistors connected to ground (pin 1).

Note that the SCSI Bus termination networks consume significant power (.8 watt). If power consumption is critical, consider having the bus termination external to the board, and externally powered.

External Termination Power Option

The SCSI specification includes the option of using *external* bus termination. To accommodate this feature, the Little Board/286 provides a jumper option (W1), which allows the board to supply termination power (+5V) to the SCSI bus TERMPWR signal (pin 26). A Shottky protection diode is included on the board, as recommended by the ANSI specification, to prevent board damage from current which might otherwise flow *from the SCSI bus*.

The default jumpering of this option is unshorted; i.e. termination power is not normally supplied by the Little Board/286.

SCSI Controller Interrupt

Jumper W2, when shorted, connects the board's SCSI interface controller interrupt output signal to interrupt IRQ15 on the AT bus. Though not currently used, the IRQ15 interrupt is reserved for future enhancements of the Ampro SCSI/BIOS. It is recommended that you not use it in your system designs, to preserve compatibility with future Ampro BIOS revisions. Therefore, leave jumper W2 in its factory setting (shorted or unshorted).

"Soft Configuration" Options

Every SCSI device must be configured for a specific SCSI bus ID, between 0 and 7. The SCSI Initiator ID of the Little Board/286 is determined by a parameter stored in the board's Configuration Memory. Use the Ampro SETUP utility to set this value. Normally, the board's SCSI "initiator" ID should be set to 7, and the ID's of the disk drives and other SCSI "target" devices should be set to 0, 1, etc.

Another important SCSI-related option, available via a parameter stored in the Configuration Memory, is whether or not the SCSI/BIOS services in the board's ROM-BIOS are desired. Using the Ampro SETUP utility, you can *disable* or *enable* SCSI/BIOS services. If you set the parameter to *disable* the SCSI/BIOS services, there will be no support BIOS support for it at all. This might be desirable for several reasons:

- To speed system booting when normal SCSI interfacing is not required, such as in a floppy-only system. Otherwise, there is a delay during system booting, while the system waits to see if a SCSI device at ID0 LUN0 becomes ready.
- To disallow SCSI/BIOS control of the SCSI hardware, when the interface is to be used in a non-standard manner.
- When using an AT hard disk controller (on the bus, or connected by means of the MiniModule/ATDisk controller interface) instead of using a SCSI device.

2.6 AT EXPANSION BUS

An AT compatible expansion bus appears at connectors J9 and J10, except that it is brought out to a pair of *header connectors*, rather than edgcard connectors.

The PC-bus subset of the board's AT Expansion Bus connects to the first 62 pins of J9; the two additional pins of J9 (A32, B32) offer additional ground signals, for enhanced system reliability. The 36-pin edgcard connector, normally included in an AT expansion bus, is replaced by a 40-pin header connector, at J10. As with J9, additional ground connections have been provided. In this case, there are two additional ground pins *at each end* of the connector. AT Expansion Bus connector pinouts are given later in this chapter.

Table 2-18 shows the pinout and signal functions of the signals on the board's AT Expansion Bus and MiniModule headers. Further information about these signals is available in numerous publications, including the IBM technical reference manuals for the PC and AT computers, and from the reference documents listed in the beginning of Chapter 4.

Note that the board's AT Expansion Bus and MiniModule connector pins are numbered differently from the other header connectors on the board, so that they correspond to the scheme normally used on the PC and AT Expansion Bus. Specifically, rather than being designated *numerically* (1, 2, 3, etc.) the signals of the AT Expansion Bus are designated *alpha-numerically* (A1, A2, ..., B1, B2, ..., etc).

With the exception of the onboard serial, parallel, and floppy controllers, all signal outputs driving the AT Expansion Bus are rated at 24 mA sink current at a logic "low" level of 0.4 Volts. (The serial and parallel controllers provide 16 mA bus drive, and the floppy controller provides 12 mA.)

Note

Most PC and AT bus plug-in cards are intended for use on an 8 MHz bus. It is therefore advisable to use one of the Ampro utilities (SETUP or SETSPEED) to enable the board's AT expansion bus "half-speed" option when PC or AT bus plug-in cards are to be used, unless the cards are known to operate properly at the faster bus speed of the Little Board/286.

2.6.1 Onboard MiniModule Header and Bus Expansion

An important feature of the Little Board/286 is its onboard MiniModule headers. Two header connectors, J9 and J10, provide an AT Expansion Bus interface for use with one of Ampro's 3.6" x 3.8" expansion MiniModules. When installed on J9 (PC bus subset of the AT expansion bus), the onboard expansion module fits within the Little Board/286's space envelope. The onboard expansion module provides a *bus pass-through* connector, allowing additional external bus expansion when the module is installed.

Several options are available for mounting multiple MiniModules on the Little Board/286:

- **Stacker/MM** -- Allows two MiniModules to be stacked together and mounted as a single unit.
- **StackPlane/MM** -- Provides two expansion bus header connectors for mounting MiniModules on a Little Board form-factor board. When used with the Stacker/MM, up to four MiniModules can be attached. The SSD Expansion board also offers two expansion bus header connectors for mounting MiniModules.
- **Pass-through connectors** -- The MiniModule/ATDisk controller offers a pass-through connector, which can be used to stack an additional MiniModule without needing a Stacker/MM.

Ampro currently offers several display controller MiniModules which offer a wide variety of standard PC and AT video modes, including VGA, EGA, CGA, MDA, and Hercules display compatibility, on CRT monitors, LCD panels, and electroluminescent (EL) panels. Ampro also offers additional MiniModules, including serial/parallel interfaces, modems, etc. Contact Ampro for information on additional MiniModules.

2.6.2 Using Standard PC and AT Bus Cards

There are several options available from Ampro which allow you to add expansion cards to the Little Board/286 system:

- **StackPlane/PC Expansion Adapter** -- a board having the identical dimensions and mounting points as the Little Board/286. This adapter converts a *PC bus* add-on card to the *Little Board* form factor. The StackPlane/PC adapter piggy-backs above or below the Little Board/286 and connects to J9 (PC bus subset of the AT bus) by means of a short ribbon cable. Several StackPlane/PC adapter assemblies can be stacked.
- **MiniBackplane/ATBus** -- two-slot *AT bus* passive backplane with 64-pin and 40-pin male bus headers for cable connection to J9 and J10 on the Little Board/286. Includes options of onboard +12V, -12V, and -5V DC-to-DC converters, which makes possible complete +5V-only system operation.
- **Cable-Mount Edgecard Sockets** -- 62-pin and 36-pin edgecard connectors can be used to construct a ribbon cable which can connect the Little Board/286's AT expansion bus (or PC bus subset) *directly* to a PC or AT plug-in expansion card using ribbon cable only. The edgecard sockets are available from

PCD Connectors
Winchester, MA
(617) 721-1280

RF18-2582-5 (36-conductor edgecard socket)
RF31-2582-5 (62-conductor edgecard socket)

Contact an Ampro sales representative for further information on these optional bus expansion products.

2.6.3 Expansion Bus Cable Considerations

Here are several points regarding the use of the board's AT Expansion Bus:

- **Cable Length** -- *Keep the ribbon cable used as short as possible.* Expansion bus cables in excess of six inches can result in reduced system reliability. We recommend:
 - For cable lengths up to 6 inches, use a high quality ribbon cable, such as 3M part number 3365/64 (64 conductor) and 3365/40 (40 conductor).
 - For cable lengths between 6 and 12 inches, use a high quality ribbon cable *with ground plane*, such as 3M part number 3353/64 (64 conductor) and 3353/40 (40 conductor).
 - Cable lengths in excess of 12 inches are not recommended.
- **Backplane Quality** -- Be sure to use a high quality backplane having minimal signal crosstalk. Use of power and ground planes, and ground guard traces between bus signals is recommended.
- **Reset Deglitching** -- Some PC and AT bus expansion cards have asynchronous TTL inputs that are especially vulnerable to cable noise and crosstalk. In particular, the active high RESET line is one to watch out for. If this signal is found to be susceptible, a 200pf to 500pf capacitor connected between the RESET signal and ground can be used to prevent false expansion card resets. This RESET deglitching capacitor is included on the Ampro MiniBackplanes.
- **Bus Half-Speed Options** -- Typically PC and AT expansion bus cards are designed for 8 MHz maximum expansion bus speed, though new boards and boards of your own design may allow faster system operation. You can use the Ampro SETUP or SETSPEED utilities to set bus operation to *full-* or *half-speed*. When set to half-speed, the "CLK" signal on expansion bus pin J9-B20 becomes half the system clock (i.e. 8 MHz on a 16 MHz board, and 6 MHz on a 12 MHz board). Use full-speed bus operation with caution!

2.6.4 Expansion Bus Connector Pinouts

Tables 2-18A through 2-18D indicate the pinout and signal functions of the signals on the board's AT Expansion Bus connectors. Further information about these signals is available in numerous publications, including the IBM technical reference manuals for the PC and AT computers, and from the reference documents listed in the beginning of Chapter 4.

The board's AT Expansion Bus header connector (J9 and J10) pins have been numbered differently from the other header connectors on the board, so that they correspond to the scheme normally used on PC and AT bus edgcard connectors. The even numbered pins correspond to the "B" and "D" rows of the AT bus; and the odd numbered pins correspond to the "A" and "C" rows (e.g. J9 pin 1 is labeled "B1"). The first 62 pins of J9 (beginning at A1 and B1) are the ones which correspond to the 62 conductors of the PC bus subset of the AT backplane. The last two pins are spare grounds. On J10, however, the *middle* 36 pins (out of 40, total) correspond to the AT bus 36 conductor edgcard connector, with 2 spare grounds *at each end* of the board's 40 pin connector. The first two pins of J10 on the Little Board/ATBus have been designated "C0" and "D0", to emphasize the fact that they are added grounds.

Note that the Little Board/286 does not generate +/- 12VDC or -5VDC for the AT Expansion Bus. If devices on the AT Expansion Bus will require these voltages, they can be supplied to the AT Expansion Bus connector from the board's utility connector (J4). One handy feature of the Ampro MiniBackplane/ATBus is that it has a provision for optional onboard DC-to-DC converters for all bus voltages other than +5VDC.

Table 2-18A. AT Expansion Bus Connector, A1-A32 (J9)

PIN	SIGNAL NAME	FUNCTION	IN/OUT
A1	-I/O CH CK	Memory Parity Error	in
A2	D7	Data bit 7	i/o
A3	D6	Data bit 6	i/o
A4	D5	Data bit 5	i/o
A5	D4	Data bit 4	i/o
A6	D3	Data bit 3	i/o
A7	D2	Data bit 2	i/o
A8	D1	Data bit 1	i/o
A9	D0	Data bit 0	i/o
A10	I/O CH RDY	Processor Ready Ctrl	in
A11	AEN	Address Enable	i/o
A12	A19	Address bit 19	i/o
A13	A18	Address bit 18	i/o
A14	A17	Address bit 17	i/o
A15	A16	Address bit 16	i/o
A16	A15	Address bit 15	i/o
A17	A14	Address bit 14	i/o
A18	A13	Address bit 13	i/o
A19	A12	Address bit 12	i/o
A20	A11	Address bit 11	i/o
A21	A10	Address bit 10	i/o
A22	A9	Address bit 9	i/o
A23	A8	Address bit 8	i/o
A24	A7	Address bit 7	i/o
A25	A6	Address bit 6	i/o
A26	A5	Address bit 5	i/o
A27	A4	Address bit 4	i/o
A28	A3	Address bit 3	i/o
A29	A2	Address bit 2	i/o
A30	A1	Address bit 1	i/o
A31	A0	Address bit 0	i/o
A32	GND (*)	Ground	---

(*) Note: A32 is an added ground. It can be left unconnected when using standard backplanes or cards.

Table 2-18B. AT Expansion Bus Connector, B1-B32 (J9)

PIN	SIGNAL NAME	FUNCTION	IN/OUT
B1	GND	Ground	---
B2	RESET	System reset signal	out
B3	+5V	+5 volts power	---
B4	IRQ9	Interrupt request 9	in
B5	-5V	To J4 pin 9	---
B6	DRQ2	DMA request 2	in
B7	-12V	To J4 pin 10	---
B8	-0WS	Zero wait state	in
B9	+12V	To J4 pin 8	---
B10	GND	Ground	---
B11	-SMEMW	Memory Write (lower 1MB)	i/o
B12	-SMEMR	Memory Read (lower 1MB)	i/o
B13	-IOW	I/O Write	i/o
B14	-IOR	I/O Read	i/o
B15	-DACK3	DMA Acknowledge 3	in
B16	DRQ3	DMA Request 3	out
B17	-DACK1	DMA Acknowledge 1	out
B18	DRQ1	DMA Request 1	in
B19	-REFRESH	Memory Refresh	i/o
B20	CLK	CPU clock (e.g. 12MHz)	out
B21	IRQ7	Interrupt Request 7	in
B22	IRQ6	Interrupt Request 6	in
B23	IRQ5	Interrupt Request 5	in
B24	IRQ4	Interrupt Request 4	in
B25	IRQ3	Interrupt Request 3	in
B26	-DACK2	DMA Acknowledge 2	out
B27	T/C	DMA Terminal Count	out
B28	BALE	Address latch enable	out
B29	+5V	+5 volts power	---
B30	OSC	14.3 MHz clock	out
B31	GND	Ground	out
B32	GND (*)	Ground	out

(*) Note: B32 is an added ground. It can be left unconnected when using standard backplanes or cards.

Table 2-18C. AT Expansion Bus Connector, C0-C19 (J10)

PIN	SIGNAL NAME	FUNCTION	IN/OUT
C0	GND (*)	Ground	---
C1	SBHE	Bus High Enable	i/o
C2	LA23	Address bit 23	i/o
C3	LA22	Address bit 22	i/o
C4	LA21	Address bit 21	i/o
C5	LA20	Address bit 20	i/o
C6	LA19	Address bit 19	i/o
C7	LA18	Address bit 18	i/o
C8	LA17	Address bit 17	i/o
C9	-MEMR	Memory Read	i/o
C10	-MEMW	Memory Write	i/o
C11	SD8	Data bit 8	i/o
C12	SD9	Data bit 9	i/o
C13	SD10	Data bit 10	i/o
C14	SD 11	Data bit 11	i/o
C15	SD12	Data bit 12	i/o
C16	SD13	Data bit 13	i/o
C17	SD14	Data bit 14	i/o
C18	SD15	Data bit 15	i/o
C19	GND (*)	Ground	i/o

(*) Note: C1 – C18 are the standard AT bus edgcard connector signals; C0 & C19 are extra grounds, and can be left unconnected when using standard backplanes or cards.

Table 2-18D. AT Expansion Bus Connector, D0-D19 (J10)

PIN	SIGNAL NAME	FUNCTION	IN/OUT
D0	GND (*)	Ground	---
D1	-MEMCS16	16-bit mem access	in
D2	-IOCS16	16-bit I/O access	in
D3	IRQ10	Interrupt Request 10	in
D4	IRQ11	Interrupt Request 11	in
D5	IRQ12	Interrupt Request 12	in
D6	IRQ13	Interrupt Request 13	in
D7	IRQ14	Interrupt Request 14	in
D8	-DACK0	DMA Acknowledge 0	out
D9	DRQ0	DMA Request 0	in
D10	-DACK5	DMA Acknowledge 5	out
D11	DRQ5	DMA Request 5	in
D12	-DACK6	DMA Acknowledge 6	out
D13	DRQ6	DMA Request 6	in
D14	-DACK7	DMA Acknowledge 7	out
D15	DRQ7	DMA Request 7	in
D16	+5V	+5 volts power	---
D17	-MASTER	Bus master assert	in
D18	GND	Ground	---
D19	GND (*)	Ground	---

(*) Note: D1 – D18 are the standard AT bus edgcard connector signals; D0 & D19 are extra grounds, and can be left unconnected when using standard backplanes or cards.

2.6.5 Interrupt and DMA Channel Usage

The PC and AT buses provide a number of interrupt and DMA control signals. When you add PC or AT bus plug-in cards which require either interrupt or DMA support, you must select which channels of these functions will be used by the added card. This is typically done using either switches or configuration jumpers on the plug-in card. It is important that you not configure the added card to use an interrupt or DMA channel that is already being used! For your convenience, Tables 2-19 and 2-20 provide a summary of the normal assignment of interrupt and DMA channels on the Little Board/286.

Table 2-19. Interrupt Channel Assignment

INTERRUPT	FUNCTION
IRQ0	ROM-BIOS clock tick function, from Timer 0
IRQ1	Keyboard controller output buffer full
IRQ2	Cascade input for IRQ8-15
IRQ3	Secondary serial port (if present)
IRQ4	Primary serial port (if present)
IRQ5	Reserved for secondary parallel printer
IRQ6	Floppy controller (if present)
IRQ7	Parallel printer (if present)
IRQ8	Battery-backed clock alarm
IRQ9	Available (corresponds to a PC's IRQ2)
IRQ10	Available
IRQ11	Available
IRQ12	Available
IRQ13	80287 Math Coprocessor (if present)
IRQ14	Reserved for hard disk controller use
IRQ15	Available

Table 2-20. DMA Channel Assignment

CHANNEL	FUNCTION
0	Available for 8-bit transfers
1	Available for 8 bit transfers
2	Floppy controller (if present)
3	Available for 8-bit transfers
4	Cascade for channels 0-3
5	Available for 16-bit transfers
6	Available for 16-bit transfers
7	SCSI controller (if present)

CHAPTER 3

SOFTWARE CONFIGURATION

3.1 INTRODUCTION

This chapter provides an overview of the system features, configuration options, and utilities that are available under the DOS operating system. A combination of standard DOS and Ampro-supplied utilities and drivers allows you to create a highly customized system based on the Little Board/286.

Assuming you have connected the Little Board/286 to a monitor, keyboard, and floppy drive as described in Chapter 2, you should have no difficulty booting the system under DOS. You will probably want to take advantage of the flexibility designed into the Little Board/286 ROM-BIOS and utilities software to create a customized DOS-based installation. You can tailor your software configuration to a wide variety of hardware configurations, as indicated in Chapter 2. This chapter provides a discussion of how to configure the operating system and utility software for many typical installations.

Some familiarity with DOS (PC-DOS, MS-DOS, or DR-DOS) is assumed. Please refer to the appropriate IBM, Microsoft, or Digital Research DOS reference manuals for further information on the use of the DOS operating system and its drivers and utilities. In addition, Appendix A of this manual contains detailed descriptions of each of the Ampro Little Board/286 driver and utility programs.

3.1.1 Conventions

In the descriptions of the use of software utilities, keyboard inputs which you will make to the system are shown underlined. This has been done to make it easy for you to distinguish between the computer's prompts and the operator's keystrokes. For example,

A><DIR<Enter>

means that you type "DIR" and then press the <Enter> key on your keyboard. (On some keyboards this key may also be labeled <Return>.)

In most cases, upper and lower case letters can be used interchangeably when you supply parameters to a program. For example:

A><SETSPEED B=HI<Enter>

has the same effect as:

A><setspeed b=hi<Enter>

OPERATION WITH DOS

The Little Board/286 and its ROM-BIOS have been designed to allow the use of IBM PC-DOS, Microsoft MS-DOS, or Digital Research DR-DOS, versions 2.x or 3.x, and PC-DOS or MS-DOS version 4.01, as the board's operating system. Throughout this chapter, the term "DOS" will be used to refer to any of these operating systems, except in cases where a difference in operation exists.

Caution

Many variations of MS-DOS exist which have been customized for operation on specific computer systems. These cannot be expected to function properly on the Little Board/286, and should not be used. Use "IBM PC-DOS", or use the "generic IBM compatible" version of MS-DOS *supplied directly by Microsoft on an OEM basis*, if you do not choose to use the DR-DOS operating system supplied by Ampro.

When used with DOS, the standard AT devices are available, including keyboard port, speaker port, parallel printer port, serial ports, programmable timer, DMA controllers, interrupt controllers, and battery-backed clock (with internal CMOS RAM). In addition, the optional Ampro MiniModule/VGA, MiniModule/EGA, MiniModule/CGA, and MiniModule/LCD onboard display controller modules offer the most popular PC- and AT-compatible video modes: VGA, EGA, CGA, MDA, and Hercules monochrome graphics on CRT, LCD, or electroluminescent (EL) displays.

The board's two 8250-based RS232C serial ports are normally supported by DOS as the "COM1" and "COM2" ports. The Parallel Printer port is normally the DOS "LPT1" device. These devices, as well as the keyboard and speaker ports, are supported by the ROM-BIOS in an AT-compatible manner.

The DOS and ROM-BIOS time-of-day clocks are supported in the usual manner. They are initialized automatically by the ROM-BIOS to the value stored in the onboard battery-backed clock on system reset or powerup.

DOS version 3.2 (or later) is required for operation with 80-track (720K) mini- or micro-floppy drives. DOS version 3.3 (or later) is needed for use with 1.44 megabyte microfloppies, and with the optional Ampro Solid State Disk support utilities. DOS version 3.x is required for SCSI hard disk usage under the board's ROM-BIOS and SCSI support software. In addition, support for hard disks larger than 32 megabytes is available under DR-DOS version 3.4 (or later), or under MS-DOS or PC-DOS versions 4.0 (or later).

3.2.1 Configuration Options

A wide degree of system configurability is available through options provided by: (1) the DOS operating system and its drivers and utilities; (2) the Ampro Little Board/286 Utilities.

Typical configuration options include:

- Console selection and setup
- Parallel and Serial printer port selection and setup
- Modem interface
- A variety of floppy disk drives
- One or more hard disk drives, either SCSI or AT compatible

Examples of how a typical system can be configured for many of these options are given later in this chapter.

The following section briefly summarizes the features of the software included in the Ampro Little Board/286 utilities and drivers. Appendix A contains detailed program descriptions.

3.2.2 Little Board/286 Utilities Overview

The following programs are contained on the Little Board/286 Utilities diskette at this time. Actual utility names and descriptions may vary, in which case appropriate "DOC" files are included on the utilities diskette. Each of these programs are described in Appendix A of this manual (alphabetically arranged).

- **SCSICOMP** -- SCSI compare utility. Compares data between two SCSI "direct access devices"
- **SCSICOPY** -- SCSI block copy utility. Copies data between two SCSI "direct access devices".
- **SCSIFMT** -- SCSI hard disk formatter. Performs low level drive format.
- **SCSIPARK** -- Hard disk park utility.
- **SCSITool** -- SCSI debugger utility. Allows low level commands to be given by a user to any SCSI bus device.
- **SCSI-ID** -- SCSI Initiator ID test. Allows testing of the SCSI Initiator ID of the board on which the utility runs. Results testable from within batch files.
- **SETSPEED** -- Provides a means to control the Little Board/286 CPU and bus half-speed options *during system operation*.
- **SETUP** -- System "soft configuration" function, accessible from a "hot-key" at boot time or command line entry during system operation.

This chapter discusses the use of many of these utilities and drivers in typical system configurations. Refer to Appendix A for detailed descriptions of each Ampro utility and driver included with the Little Board/286.

3.2.3 Two Special Files

Before reading about the various options available for customizing your system, you should understand the use of two special DOS files: CONFIG.SYS and AUTOEXEC.BAT. These are discussed in detail in the DOS reference manuals, but will now be discussed briefly as an introduction to this chapter.

- **CONFIG.SYS** -- "CONFIG.SYS" is a text file, which you must create. It is used to set certain system parameters and to instruct the operating system to install one or more *drivers* or system extensions. Each driver is a program which will be loaded by DOS from your boot device. For each driver program that you want included in your system's operating system configuration, you simply include its name in a text file called "CONFIG.SYS" which must be *in the root directory on your boot drive* (floppy, hard disk, or SSD). Refer to your DOS technical reference for details on the structure, creation, and use of the CONFIG.SYS file. Note that is not enough to simply create the CONFIG.SYS file. You must reboot your system from a drive containing the new CONFIG.SYS file in its root directory to have DOS actually install the desired drivers and functions as part of your system's operating environment.
- **AUTOEXEC.BAT** -- The second special file is called "AUTOEXEC.BAT". This is a file of commands that you would like to have your system run automatically on powerup or reset. As with the CONFIG.SYS file, the AUTOEXEC.BAT file must be located *in the root directory on your boot drive*, and you must reboot your system from a drive containing the new AUTOEXEC.BAT file in its root directory, to have DOS actually execute the programs and functions specified.

In this chapter, you will see that there are a number of Ampro utility commands which you might wish to include in an AUTOEXEC.BAT file.

3.2.4 The Ampro SETUP Function

The Ampro SETUP function is used to initialize or modify the contents of the board's nonvolatile *Configuration Memory*, i.e. the NOVRAM in the battery-backed clock chip, and the configuration EEPROM device. The effect of SETUP occurs *at system boot time*.

The SETUP function is contained directly within the ROM-BIOS. This allows access to all configuration parameters *without the use of the SETUP utility*, by using the "Ctrl-Alt-Esc" keyboard sequence at powerup or reset time. As a convenience, the SETUP utility provides entry into the ROM-BIOS SETUP function at any time during system operation. The SETUP utility also allows you to save or restore Configuration Memory contents to or from disk files. Refer to Appendix A for a list of the system parameters controlled by SETUP and a description of its use.

3.3 CPU AND BUS SPEED INITIALIZATION

One of the most important system parameters to configure is the half-speed option for the CPU and/or expansion bus. In most installations, you will want to have the CPU operate at full speed. Assuming 12 or 16 MHz operation of the board's CPU, it is most common to operate the bus at "half-speed", due to the fact that many PC and AT bus expansion boards require a maximum bus speed of 8 MHz.

There are two ways to control the half-speed options:

3.3.1 Fixed Speed Operation (Using SETUP)

The onboard nonvolatile configuration memory contains two parameters which indicate to the ROM-BIOS whether to automatically set the CPU and/or expansion bus half-speed options, *on powerup or reset*. The Ampro SETUP function allows you to store the desired values for these two parameters in the board's Configuration Memory. Note that the effect of using SETUP to change either half-speed option does not take effect until you reboot the system.

3.3.2 Variable Speed Operation (Using SETSPEED)

During system operation, you can *dynamically* alter the half-speed option for either the CPU or expansion bus using the Ampro SETSPEED utility. The SETSPEED program can be invoked *manually*, from the DOS command line. Alternatively, a SETSPEED command can be issued *automatically*, from within a batch file. The effect of using SETSPEED is immediate, and does not require rebooting the system.

Some examples of when you might use the SETSPEED command are:

- To reduce CPU speed for the purpose of conserving power during battery operation (up to 15%).
- To allow the use of a program which was intended for operation on a slower computer; you might run the speed-critical from a batch file such as this:

```
SETSPEED /P=LO  
SLOWPROG  
SETSPEED /P=HI
```

- To allow the use of an expansion bus card that cannot operate at full bus speed; you might shift the bus to half-speed prior to running the slow card's operating software, and then return the bus to full speed afterwards, as in the previous example.
- For a brief SETSPEED help screen, use the command:

```
A><u>SETSPEED ?</u><Enter>
```

3.4 USING A PARALLEL PRINTER

No special configuration is required when using the system with a Centronics type parallel printer. Simply refer to the parallel printer as the DOS "LPT1" or "LPT2" device (depending on how the port is jumpered -- see note below), and be sure the parallel printer port is *enabled*, using SETUP. Most application software uses LPT1 as the default printer port, so printing to the parallel printer interface is automatic if the port is jumpered that way.

For example, the following DOS commands will result in printing to the parallel printer:

A><u>COPY CONFIG.SYS LPT1</u><Enter> . . . prints contents of CONFIG.SYS

A><u>DIR >LPT1</u><Enter> . . . prints the directory

In addition, the <PrtSc> ("Print Screen") key will print the contents of the video screen to the LPT1 device. Also, the "Printer Echo" function can be used to cause all characters written to the console to also be written to the LPT1 device. The Printer Echo function is enabled with <Ctrl-P>. Printer Echo remains enabled until another <Ctrl-P> is entered.

Note

The "LPTn" (n=1,2) designation is a "logical" value, not a "physical" value. During the system boot sequence, the ROM-BIOS scans both parallel printer addresses, and installs the first parallel printer it locates as the LPT1 device, and the second, if found, as LPT2. Therefore, simply changing the port's address and interrupt jumpers on the Little Board/286 will not change it from LPT1 to LPT2; there must be an LPT1 device elsewhere in the system to be able to change the onboard printer port to LPT2

3.5 USING THE SERIAL PORTS

This section discusses several uses of the board's two RS232C serial ports, and provides some examples of typical installations.

3.5.1 Serial Port Initialization

In general, before the board's serial ports can be used, they must be properly enabled and initialized. This is sometimes done directly by the application program which will access the port's hardware, but is usually done as part of the system configuration and boot process.

Each serial port must be *enabled*, for their hardware functions to be available. The board's ROM-BIOS enables or disables each of the two ports based on a parameter within the board's Configuration Memory. You can enable or disable either serial port using the SETUP utility. Note that the effect of *changing a port's enable/disable function using SETUP does not take effect until you reboot the system.*

Note

The "COMn" (n=1,2) designation is a logical value, not a physical value. During the system boot sequence, the ROM-BIOS scans both sets of serial port addresses, and installs the first serial port it locates as the COM1 device, and the second, if found, as COM2. Therefore, if you disable the board's primary serial port, the secondary port will become "COM1", not "COM2"! There must be COM1 (primary serial port) device elsewhere in the system for the onboard secondary serial port to be accessed as COM2.

In addition, each port's baud rate and data characteristics must be initialized to match the requirements of the external devices that will be connected. One way to set (and alter) either serial port's baud rate and data characteristics is by means of the DOS MODE command. Here is how you might use the MODE command to initialize the COM1 port to 9600 baud, no parity, 8 data bits, and 1 stop bit:

MODE COM1:9600,n,8,1

In many cases it is not necessary to use a mode command, because the software which will operate the serial port will initialize the port when it loads.

3.5.2 Serial Console Option

In many *embedded* applications it is desirable to substitute an RS232C serial device (terminal, remote computer, or other serial device) for the video controller, monitor, and keyboard normally used as the PC console device. The serial console device can be connected to the board's primary or secondary serial port; the Ampro ROM-BIOS contains support for serial input and serial output. Serial console support parameters are stored in the board's Configuration Memory and are controlled by the SETUP function, which is used to install the required console support.

SETUP parameters provide support for serial console input, such as a serial keyboard or modem, and serial console output, such as a serial display terminal or remote computer. Additionally, a standard AT keyboard can be combined with a serial display, or serial input can be used with a standard video display.

Caution

Be careful when changing the console configuration using SETUP. It is possible to specify "none" for console input and output, in which case there will be no console access to the system.

Note

Many DOS programs use ROM-BIOS display control functions, or write directly to video RAM. Such software cannot be used on a serial console device (e.g. ASCII terminal), with the exception of programs intended to be used with the DOS ANSLSYS driver if the terminal is ANSI compatible. On the other hand, programs which output appropriate terminal control strings to the console, or which are specifically installed for the particular serial display device, can be used.

To allow an RS232C serial device connected to either of the two serial ports to be used as a system console device, use the SETUP utility to select the serial port and configure the appropriate parameters. As an example, the console output can be set to the primary serial port, 19.2K baud, 8 data bits and 1 stop bit, while the console input can be set to the secondary serial port, 9600 baud, 8 data bits and 1 stop bit.

After booting the system with this configuration, data normally destined for a standard video display controller and monitor will be routed to the external serial device instead. Keyboard data will be obtained from *both* the external serial device *and* the standard AT keyboard (if present).

Refer to Appendix A for additional information on using SETUP to install the serial driver options.

Serial Console Cabling and Setup

The serial console device must naturally be set to the same data format as the board's serial port. Normally, the Little Board/286 serial port's Data Set Ready (DSR) and Clear To Send (CTS) *input handshake signals must be true* ("active"), for the ROM-BIOS to send data out. (This hardware handshake can be disabled with the SETUP utility.) When hardware handshaking is enabled, be sure to connect the DSR and CTS signal inputs to appropriate handshake signals from the external serial device's serial interface connector. Alternatively, loop the board's serial output handshake signals to its input signals as follows: DTR out, to DSR in; RTS out, to CTS in.

3.5.3 Using a Serial Printer

Serial printers can be connected to either of the board's RS232C serial ports as well.

You can instruct DOS to print to the appropriate serial port (COM1 or COM2) instead of to the Parallel Printer port (LPT1 or LPT2) with the DOS MODE utility. For example, the following command makes the COM1 serial device become the system's default printer device:

A><u>MODE LPT1=COM1</u><Enter>

Now you can print to the serial printer as though it is connected to the system's parallel printer port.

You can return the printer device to the Parallel Printer port with the following command:

A>MODE LPT1<Enter>

If desired, MODE commands can be included in your system's AUTOEXEC.BAT startup file to assign the printer function to a serial port. Also, additional MODE commands can be used in other batch files to dynamically reassign the DOS printer device function as often as required during system operation.

Printer Cabling and Setup

The printer must be set for a suitable baud rate, and appropriate data and handshake signals must be connected between the computer's and the Printer's serial connectors. In particular, be sure to connect the serial port's "Ready to Receive Data" handshake output signal to the serial port's Clear to Send (CTS) input handshake input signal. In addition, the board's Data Set Ready (DSR) *input handshake signal must be true*, for the ROM-BIOS to send output data, so connect that signal to an appropriate handshake output signal from the printer, or loop it to the board's DTR output signal. Chapter 2 describes the jumper options available for configuring the serial port.

3.5.4 Using a Serial Modem

In addition to their use as console or serial printer ports, either of the board's two RS232C serial ports can be used as a modem interface. Most PC communications programs control the serial port hardware directly, rather than using DOS or ROM-BIOS functions. When configured as described in Chapter 2, the board's serial ports are fully compatible with those of standard AT's and PC's, so all such software should work properly.

When connecting a serial modem, be sure to connect appropriate input and output handshake signals, depending on what is required by the communications software you use. Most communications programs take care of initializing the serial port baud rate and data characteristics; if not, be sure to use the DOS MODE command to initialize the port.

A number of powerful communications programs are available which can be used to control communications applications via a modem. Several of these programs offer powerful "script" languages, which allow you to generate complex automatically functioning applications with very little programming effort. Examples are: PCTALK, PROCOMM, XTALK, and MIRROR.

3.6 USING FLOPPY DRIVES

DOS compatible floppy disk functions are provided within the ROM-BIOS. Diskettes may be copied, formatted, verified, etc. using the standard DOS commands. This includes all of the **DOS** mini (5-1/4") and micro (3-1/2") floppy formats -- 160K, 180K, 320K, 360K, 720K, 1.2M, and 1.44M -- provided the appropriate version of DOS is used (see Chapter 2). In addition, dual capacity use of high density floppy drives is supported, which means that 360K floppies can be read in a 1.2M 5-1/4" drive, and 720K floppies can be read in a 1.44M 3-1/2" drive.

The Little Board/286 contains a non-volatile RAM system Configuration Memory, which contains information on the number and type of floppy drives connected to the system. As shipped, the board's Configuration Memory is initialized to expect the first physical floppy drive (called "A") to be a 360K 5-1/4" drive. However, the board's ROM-BIOS automatically adapts to whatever type of drive is connected as drive A, whether 360K, 720K, 1.2M, or 1.44M. Therefore, you can boot your system from any type of drive. In addition, the board's ROM-BIOS supports dual capacity use of high density drives (1.2M and 1.44M), so you can boot from a 360K 5-1/4" diskette in a 1.2M 5-1/4" drive, and you can boot from a 720K 3-1/2" diskette in a 1.44M 3-1/2" drive.

3.6.1 Drive Parameter Setup

Within the board's Configuration Memory are parameters for the quantity (one or two) and type (360K, 720K, 1.2M, or 1.44M) of drives. If the drives connected to your system do not match those in the board's Configuration Memory, you will see an error message indicating the mismatch, and instructing you to "Run SETUP". Once you have booted DOS, you can reconfigure the system for the correct floppy drive configuration using the Ampro SETUP utility. For instructions on using SETUP, refer to the detailed description of the SETUP utility in Appendix A of this manual.

3.6.2 Single-Floppy Configurations

One handy feature of DOS is its built-in support for single-drive systems. If you specify only one floppy drive in the board's Configuration Memory (using SETUP), the operating system will automatically assign drive letters "A" and "B" to the single floppy drive. With this configuration, you can copy files between two diskettes on one drive, as though your system has two drives; DOS will prompt you to change diskettes when needed.

3.7 USING SCSI HARD DISK DRIVES

One unique feature of the Little Board/286 is that its ROM-BIOS contains hard disk support functions, which map to the board's Small Computer System Interface, rather than to a standard AT bus hard disk controller. This has the advantage of offering the added flexibility of SCSI device support, with its benefits of interchangeability of peripherals, flexibility of configuration, and ease of system upgrading and support.

Hard disk devices accessed via the board's SCSI interface are available to DOS through standard ROM-BIOS functions (INT 13), supplied by the Ampro SCSI/BIOS contained within the board's ROM-BIOS. The ROM-BIOS hard disk support allows direct system booting from SCSI Common Command *Set direct access devices*. In addition to hard disks, other types of SCSI *direct access devices* can be used interchangeably to provide a fully compatible hard disk function. These include bubble memory drives, RAM disks, optical disks, and tape drives.

Virtually all DOS applications run normally in this SCSI-based hard disk environment, because programs nearly always use either DOS or ROM-BIOS functions for disk drive access. It is extremely rare for DOS environment software to attempt to access hard disk controller *hardware* directly. If a program does require disk controller *hardware* access, it will need to be modified to use the board's SCSI hardware or SCSI/BIOS functions. In this case, low level SCSI functions available within the board's ROM-BIOS can be used to simplify the task.

In most cases, SCSI hard disk drives can be used without the need for custom programming. The board's ROM-BIOS and SCSI support software allow the use of one or more hard disk drives under PC-DOS, MS-DOS, DR-DOS.

A combination of Ampro and DOS utilities are used in the formatting and preparation of SCSI hard disk drives. Included on the Ampro utilities diskette are utilities for SCSI drive formatting, parking, and other SCSI functions.

3.7.1 Drive Parameter Setup

Several parameters need to be set in the board's Configuration Memory, using the Ampro SETUP utility. These are outlined below. For detailed instructions on SETUP, refer to the program's description in the Appendix of this manual.

- **SCSI Initiator ID** -- the Ampro Little Board/286 is the SCSI *Initiator* in its transactions with SCSI *Target* devices such as hard disk drives. Every SCSI device (Target or Initiator) must have a unique ID, between 0 and 7. Usually, the Ampro Little Board/286 is set for "7", because this is the highest priority ID, and because the Ampro SCSI/BIOS performs a SCSI bus Reset function on system powerup or reset when the CPU board's ID is 7.
- **SCSI Target Device ID's and LUN's** -- the Configuration Memory has space for the specification of one or more SCSI target device ID's and Logical Unit Numbers (LUN's). Every SCSI Target device must have a SCSI ID, between 0 and 7, normally set by jumpers or switches on the device. Unless multiple devices are operated from a single SCSI Target controller, the LUN will be 0. For example, a typical system with a SCSI drive connected to it will have the drive configured as ID0, LUN0. Multiple SCSI drives, if present and properly formatted, can result in multiple DOS hard disk drives being installed by the ROM-BIOS at system boot time.

Note

PC-DOS and MS-DOS allow installing one or two hard disk drives, while DR-DOS allows up to eight.

- **BOOT Device Specification** -- the board's Configuration Memory also contains a parameter indicating whether a hard disk or floppy drive is to be used as a boot device. SETUP's "Extended Options/Peripheral Configuration" menu contains this parameter. Using this option, the boot device can be set as either the "Floppy Drive" or "Hard Drive". When "Hard Drive" is selected, the drive indicated by this menu under "1st Hard Disk" becomes the boot drive.
- **SCSI Disk I/O Retries** -- you can specify the number of read/write retries to be performed by the board's ROM-BIOS routines *during DOS use* of SCSI drives. This option is located in SETUP's "Extended SCSI and Hard Disk Configuration" menu.
- **SCSI/BIOS Services Enable/Disable** -- you can enable or disable the SCSI functions within the board's ROM-BIOS. Naturally, if you want to use SCSI for hard disk drives under DOS, this option must be enabled. Disabling the SCSI/BIOS services will speed up system booting when the SCSI functions in the BIOS are not required, and will prevent the BIOS from accessing the SCSI hardware in applications that use the SCSI interface for non-SCSI purposes.

Note

Do not define disk drive parameters for the "AT HDC Disk" options in SETUP's "Standard (CMOS) Setup" menu for SCSI-connected drives. These parameters are for defining drives connected to an AT bus hard disk controller, or for drives connected to an "ATA" (or "IDE") drive interface.

4.0.0...._@_@ Preparation for DOS Use

Before you can use a hard disk drive connected to the board's SCSI port, it must be properly connected and jumpered, and the appropriate parameters must be set in the board's Configuration Memory as indicated above. In most cases, formatting and other preparation for use with DOS is a simple process, and can be accomplished as indicated in this section. For additional details on the installation and preparation of SCSI disk drives, refer to the detailed program descriptions in the Appendix of this manual.

The following is a brief procedure you can use to prepare a SCSI hard disk drive for use with DOS:

4. **Setting the SCSI Device ID's**— using the options in SETUP's Extended SCSI and "Hard Disk Configuration" menu, specify the appropriate SCSI device ID's for both the drive (must match jumpers on the drive or drive controller) and the Ampro Little Board/286. Normally, you will want to set the Ampro Little Board/286 to SCSI ID 7 so that a SCSI bus Reset will be issued on system powerup or reset. Typically, SCSI drives come preset to SCSI ID 0, LUN 0. The "SCSI Initiator ID" option sets the CPU board's SCSI Initiator ID; "SCSI Disk Map" options are used to specify the ID and LUN of up to seven SCSI drives; "DOS Disk Map" options designate one or more SCSI devices for access by DOS. For example, in a system with one SCSI drive, you will probably set "SCSI Initiator" to "7", "SCSI Disk 1" to "Id 0, Lun 0", and "1st Hard Disk" to "SCSI Disk 1".
4. **"Low Level" Formatting** – The low level format erases all data from the drive and prepares it for use. In many cases, the low level format function eliminates bad blocks from the usable area of the drive, by using spare blocks to replace bad ones. Most embedded SCSI drives are factory formatted. However, it is best to perform a low level format prior to using a drive for the first time. Use the Ampro SCISIFMT utility to perform the low level format of the drive. The drive must be SCSI *Common Command Set* ("CCS") compatible, to be formatted by the SCISIFMT utility.
4. **Drive Partitioning** – Reboot the system from a floppy diskette in drive A containing the operating system, and run the DOS FDISK utility in the manner described in your DOS documentation. You may be creating one or multiple partitions, depending on the size of the drive and the partition limitations of the particular DOS you are using.
4. **Final Preparation for System Access**— Again, reboot the system from a floppy diskette in drive A. What you do next varies, depending on what operating system you are using.

DR-DOS: Run the DOS "SYS" command, to copy the operating system to the hard disk drive(s) that you have created in the above steps. Finally, copy anything else you need to the drive(s), and then reboot the system *without the floppy diskette in drive A*, to verify that everything has been installed properly.

PC-DOS or MS-DOS: Use the "FORMAT /S" command, to copy the operating system to the DOS *boot* drive ("drive C"); or the "FORMAT" command, in case of drives or drive partitions other than the DOS boot partition. Finally, copy anything else you need to the drive(s), and then reboot the system *without the floppy diskette in drive A*, to verify that everything has been installed properly.

3.8 USING AT BUS CONTROLLER INTERFACED HARD DISK DRIVES

The Little Board/286 can be used with one or two hard disk drives connected by means of a standard AT bus plug-in hard disk controller. These can be used instead of SCSI hard disk drives, or *along with* SCSI drives. Included in the AT bus controller hard disk drive category are drives which have *embedded* AT bus hard disk controllers -- referred to as AT Bus Attachment (ATA) or Integrated Drive Electronics (IDE) drives -- interfaced by means of an Ampro MiniModule/ATDisk adapter that provides the required buffering and other interface signals.

These can be used instead of SCSI hard disk drives, or *along with* SCSI drives. In DOS applications, the total number of drives used (SCSI and AT bus controller interfaced) can be up to eight under DR-DOS, or two, under PC-DOS or MS-DOS.

One advantage of using an ATA interface rather than SCSI for hard disk interfacing is that operating systems or programs that talk directly to a standard AT compatible hard disk controllers can successfully be used with a Little Board/286. This includes multitasking operating systems such as Unix, Xenix, QNX, etc.

To configure the system for use with one or two ATA or AT bus hard disk controller interfaced hard disk drives, several parameters must be set in the board's Configuration Memory, using the Ampro SETUP utility. These parameters are outlined below. For detailed instructions on the use of SETUP, refer to the program's description in Appendix A of this manual. When using the setup program or other Ampro software utilities, ATA interfaced hard disk drives are referred to as drives connected to an *AT bus hard disk controller*.

- **Drive Characteristics** -- the board's Configuration Memory contains the parameters which specify the physical format of the each drive to be connected by means of an AT bus hard disk controller (or ATA drives). Included are the total number of cylinders, number of heads, cylinder to begin precompensation, landing zone cylinder number, and the number of sectors per cylinder. These parameters are set using SETUP's drive definition options in the first menu page of the Standard (CMOS) Setup.

Note

Do not set the data for these parameters if the drive is connected via SCSI

- **Drive Selection** -- in addition to specifying the physical characteristics of each AT hard disk controller (or ATA) interfaced hard disk drive, you must also specify how they are to be used by the ROM-BIOS. This is done using the SETUP Extended SCSI and Hard Disk Configuration menu (third menu page). Specify one or two AT hard disk controller (or ATA) interfaced drives (defined in the above step) in the DOS disk map as DOS drives to be installed by the ROM-BIOS at system boot time. Note that you can have up to four drives total (depending on the type of DOS being used), and they can be a combination of SCSI and AT hard disk controller (or ATA) interfaced drives.

- **BOOT Device Specification** -- the board's Configuration Memory also contains a parameter indicating whether a hard disk or floppy drive is to be used as a boot device. SETUP's Extended Options/Peripheral Configuration menu (second menu page) contains this setting. Using this option, the boot device can be set as either the "Floppy Drive" or "DOS Hard Drive". When "DOS Hard Drive" is selected, the drive indicated by the "1st Hard Disk" in the "DOS Disk Map," defined above, becomes the boot drive.

Once the system's Configuration Memory is properly configured in this manner, the AT hard disk controller (or ATA) interfaced hard disk drive(s) can be formatted and otherwise prepared in the standard manner. Refer to your operating system, disk controller, and disk drive documentation for specific procedures and requirements.

CHAPTER 4

THEORY OF OPERATION

4.1 INTRODUCTION

With the exception of several unique functions which are discussed in detail in this chapter, the Little Board/286 is functionally identical, from both a hardware and software perspective, with the IBM PC/AT computer. This chapter will only briefly touch on *standard* PC/AT features and functions, but rather will focus on the *non-standard* functions, which are *unique to the Ampro Little Board/286*.

For detailed technical information on the architecture and functions of a PC/AT and its normal software environment, it is recommended that you consult publications such as the following:

- **IBM Technical Reference, Personal Computer AT**
IBM Document number 1502243
- **DR DOS User's and Reference Guide** -- available from Ampro
- **IBM Disk Operating System Version 3.30**
IBM Document number 80X0667
- **IBM Disk Operating System Technical Reference**
IBM Document number 80X0945
- **The Peter Norton Programmer's Guide to the IBM PC**
Microsoft Press
A Division of Microsoft Corporation
10700 Northrup Way
Box 97200
Bellevue, Washington 98009
- **Interfacing to the IBM Personal Computer**
Lewis C. Eggebrecht
Howard W. Sams & Co., Inc.
A Subsidiary of Macmillan, Inc.
4300 West 62nd Street
Indianapolis, IN 46268 USA

In reading the material regarding the board's hardware functions, bear in mind that many of the board's devices are highly programmable, so their functional behavior is dependent on a variety of programming options. Therefore, to explain the normal functions performed by these devices as used on the Little Board/286, it is assumed that those devices are initialized and/or operated by the board's ROM-BIOS.

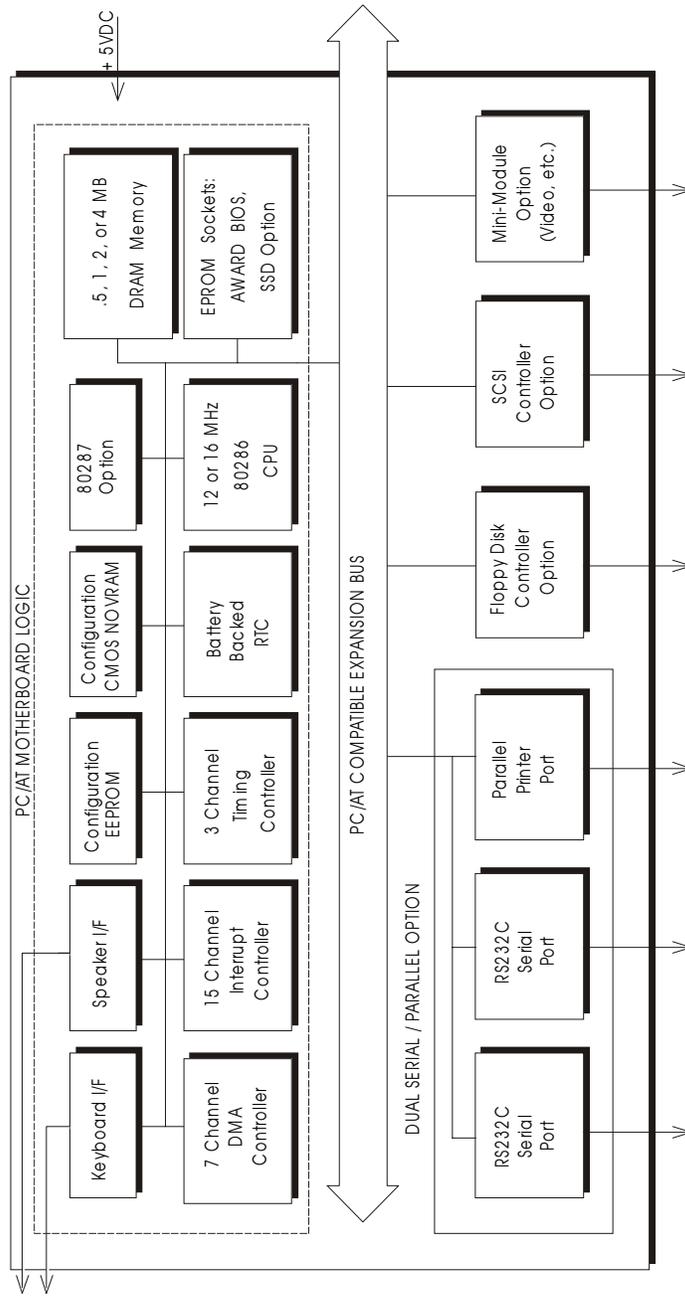


Figure 4-1. Little Board/286 B lock Diagram

4.2 OVERALL ARCHITECTURE

As mentioned previously, the Little Board/286 is essentially a complete PC/AT compatible *system*, including the equivalent of a motherboard and three or four expansion cards, constructed on a single electronic assembly that has the same dimensions as a AT expansion card.

Figure 4-1 shows a block diagram of the Little Board/286's onboard logic.

The following sections discuss the Little Board/286 onboard subsystems, grouped as *standard AT system functions* and *unique functions* (not found in standard AT systems):

4.2.1 Standard AT System Functions

- **AT Motherboard Logic** -- this includes the 80286 CPU, system and extended memory (up to 4 megabytes, total), ROM-BIOS, DMA controllers, interrupt controllers, system clocks, programmable timers, keyboard interface, speaker port, battery-backed real time clock and CMOS RAM (with watch-dog timer option), and the 80287/80C287 math coprocessor socket.
- **Floppy Disk Controller** -- a complete AT compatible floppy disk controller.
- **Serial/Parallel Controller** -- two AT compatible serial ports (with an option of RS232C/RS422/RS485 driver/receivers on the primary port); and an AT-compatible parallel printer port (with bidirectional data option).
- **AT Expansion Bus** -- a full AT compatible (PC bus superset) I/O channel interface allowing the addition of PC and AT compatible expansion cards.

4.2.2 Unique Functions

- **Byte-wide Memory Sockets** -- two 28-pin byte-wide sockets which are usable as directly accessible data or program memory, or as DOS *solid state disk (SSD) drives* (using the Ampro SSD Support Software option).
- **Configuration EEPROM** -- a 1024 bit serial electrically reprogrammable EPROM device which is used to store system configuration (and other) data. This provides additional setup configuration functions beyond those normally contained in the nonvolatile CMOS memory within an AT's battery backed clock device.
- **Small Computer System Interface (SCSI) Host Adapter** -- a full-function ANSI X3.131 compatible asynchronous/synchronous SCSI bus interface. The board's ROM-BIOS supports a one or more SCSI *direct access devices* (asynchronous usage only) as AT compatible hard disks.

4.2.3 System Memory Map

The 80286 used as the central processing unit (CPU) on the Little Board/286 has a memory address space which is defined by 24 address bits. Therefore, it can address 16 megabytes of memory. The board's memory space is utilized as indicated in Table 4-1.

Basically, the board's onboard resources (including up to 4 MB DRAM, optional video module, byte-wide sockets, and ROM-BIOS) occupy the first megabyte (starting at 00000h), and also provide an option of onboard *extended memory* (384K or more) starting at 100000h. The remainder of the board's 16 megabyte memory space is available for expansion cards on the board's 16-bit AT Expansion Bus.

Table 4-1. Little Board/286 Memory Map

Memory	Function
FE0000-FFFFFFh	Duplicates onboard memory at 0e0000-0FFFFFFh
100000-FDFFFFh	"Extended memory" space. 384K or more can be supplied by onboard DRAM, if 1MB or more of DRAM is installed onboard, and jumpers are set to enable onboard <u>extended memory</u> .
0F0000-0FFFFFFh	ROM-BIOS sockets at U10 and U11. U10 contains "odd" address byte, and U11 contains "even" address byte. Contains a startup vector at address 0FFFF0h.
0E0000-0EFFFFh	Duplicates onboard memory at 0F0000-0FFFFFFh. Reserved for future expansion.
0D8000-0DFFFFh	Byte-wide memory socket at U33.
0C0000-0C7FFFh	Byte-wide memory socket at U32.
0C0000-0CFFFFh	Reserved for expansion bus ROM's.
0A0000-0BFFFFh	Normally contains video RAM, as follows: CGA Video: B8000 – BFFFFh Monochrome: B0000 – B7FFFh EGA Video: A0000 - AFFFFh
080000-09FFFFh	128k bytes onboard DRAM (optional).
000000-07FFFFh	512k bytes onboard DRAM.

4.2.4 System I/O Map

Table 4-2 provides a detailed listing of the I/O port assignments used on the Little Board/286. With the exception of the SCSI controller and configuration EEPROM access register, the I/O port functions and addresses shown in Table 4-2 are all functionally identical with their counterparts in a "standard AT" system from both a hardware and software perspective.

In many cases, the board's ROM-BIOS functions provide all of the services that you will need to control and access devices on the board and connected to its I/O interfaces. If direct programming of the board's peripheral interfaces is necessary, refer to one of the many available references on programming the IBM PC and PC/AT for details on programming the standard functions.

Table 4-2. Little Board/286 I/O Map

I/O Address	Function
03F8 - 03FFh	Primary serial port
03F0 - 03F7h	Floppy disk controller ports (37C65B) 3F2 – FDC Digital output register (LDOR) 3F4 – FDC Main status register 3F5 – FDC Data register 3F7 – FDC Control register (LDCR)
03D0 – 03DFh	CGA display adapter (option)
03C0 – 03CFh	EGA display adapter (option)
03B0 – 03BFh	Monochrome display adapter (option)
03A0 – 03Afh	Reserved for primary bisync port
0380 – 038Fh	Reserved for secondary bisync port
0378 – 037Fh	Primary parallel printer port
0372 – 0373h	Configuration EEPROM access register
0370 – 0371h	SCSI controller ports (AIC-6250)
0360 – 036Fh	Reserved
0300 – 031Fh	Reserved for prototype card
02F8 – 02FFh	Secondary serial port
0278 – 027Fh	Secondary parallel printer port
0200 – 0207h	Reserved for game port
01F0 – 01F7h	Reserved for AT bus hard disk controller
00F0 – 00FFh	Math coprocessor (80287) 00F1 – Reset 00F0 – Clear busy
00C0 – 00DFh	DMA controller 2 (8237 equivalent)
00A0 – 00BFh	Interrupt controller 2 (8359 equivalent)
0080 – 009Fh	DMA page registers (74LS612 equivalent)
0070 – 007Fh	Real time clock and NMI mask
0060 – 006Fh	Keyboard controller (8042 equivalent)
0040 – 005Fh	Programmable timer (8254 equivalent)
0020 – 003Fh	Interrupt controller 1 (8359 equivalent)

The remainder of this chapter will examine the above functions, with brief emphasis on the *standard functions*, and more detailed emphasis on the *unique functions*.

4.3 AT MOTHERBOARD LOGIC

Among its many functions, the Little Board/286 contains the equivalent of an entire PC/AT motherboard. This section briefly describes the *motherboard subsystem* contained within the Little Board/286.

4.3.1 CPU

A powerful 80286-type microprocessor (U5) represents the central element in the Little Board/286. A variety of equivalent devices can be used, including the Intel 80286 standard device, the AMD 80L286 low power device, and Harris (or other) 80C286 CMOS device.

The Little Board/286 is designed to operate at a variety of clock speeds, determined by the value of the master clock oscillator, U20. U20 must be selected for twice the desired CPU clock rate. Standard product is currently available from Ampro with either 16 or 12 MHz CPU clock rates. Appropriate speed 80286, RAM, delay line, and math coprocessor devices must be present, depending on the CPU clock rate. In addition, the board must be properly jumpered according to the clock rate and memory wait state requirements (factory settings, do not alter!).

4.3.2 CPU and Bus Speed Control

To allow for maximum software and hardware flexibility, logic has been included which allows you to shift the CPU or the bus to half-speed mode, under software control. The CPU and bus speed together, or the bus speed alone, can be placed at half-speed. For example, a 12 MHz Little Board/286 can be made to operate at a CPU and bus rate of 6 MHz, or its expansion bus alone can be slowed down.

When the CPU is running at full speed and the bus is set for half-speed, the CPU operating frequency automatically shifts to half its normal rate during memory and I/O transactions on the AT expansion bus. When this occurs, the CLK signal on the expansion bus (J9-B20) goes to 1/2 its normal speed.

Note

The bus cannot run faster than the CPU, so if the CPU speed option is set for half-speed, the bus will run at half-speed regardless of the setting of the bus speed option.

Most software is independent of CPU speed. However, occasionally programs may be found which were intended for older 6 MHz or 8 MHz AT's. In this case, employing the *CPU half-speed* option under software control may be useful. In addition, many PC or AT Expansion Bus plug-in cards have data access times which are too slow for 12 or 16 MHz expansion bus operation, so the *bus half-speed* option -- especially switched during system operation via software -- can be useful. It is also possible to reduce the board's power consumption and heat dissipation by making use of the CPU half-speed option.

The CPU and bus half-speed options are controlled by the ROM-BIOS SETUP utility and SETSPEED program, which is included on the Little Board/286 Utilities diskettes. Refer to Chapter 3 for further details. The expansion bus speed parameter in the board's Configuration Memory is set to *half-speed* operation at the factory.

4.3.3 ROM-BIOS Sockets

A pair of EPROM sockets (U10,U11) with 16-bit wide data access normally contains the board's AT compatible ROM-BIOS. Each socket is intended for use with a 27C256 EPROM (or equivalent) device. The "high" (odd) data byte EPROM goes in U10, and the "low" (even) byte goes in U11.

The board's wait state control logic automatically inserts four memory wait states in all CPU accesses to these sockets. The ROM-BIOS sockets occupy the memory area from F0000h through FFFFFh.

4.3.4 Onboard DRAM Memory

The board has four positions (U1-4) for "SIMM" DRAM modules each of which provides either 256K X 9 bits or 1M X 9 bits of system RAM. The onboard DRAM memory occupies the *bottom* of the 80286's 16 megabyte memory space, beginning at 00000h.

The ninth bit of each DRAM module is used to generate a RAM parity bit, and the board's ASIC devices generate or check RAM *parity* on each memory write or read. If a parity error is detected, an NMI interrupt is generated, and the system software then determines what corrective action is to be taken.

Either two or four onboard DRAM modules must be installed for the board to function. Configuration jumpers indicate the number of DRAM memory modules present (two or four). Additional system RAM, up to a maximum of 16 megabytes, can be installed on the AT Expansion Bus.

A jumper option offers the feature of converting RAM memory beyond 640K bytes into what is termed *extended memory*. With the extended memory option enabled, 640K bytes of the board's DRAM is present in the normal 80286 system address space, and the balance of the memory installed on the board is accessed in the extended address range beginning at 100000h. This extended memory can be accessed by means of a number of available application and utility programs, including both cache and RAM-disk drivers. Special programming techniques must be used when writing software which accesses extended memory.

DRAM refresh is accomplished in the standard manner, with a 15 uS time base generated by Timer 1. Refresh addresses are generated by one of the board's ASIC devices, and a REFRESH signal on the AT Expansion Bus accommodates expansion cards plugged into the bus.

4.3.5 Interrupt Controllers

A pair of 8259A compatible interrupt controllers, within the board's ASIC devices, provides fifteen prioritized interrupt levels. Of these, several are normally associated with the board's onboard device interfaces and controllers, and several are available on the AT Expansion Bus. Table 4-3 lists the normal interrupt usage.

Table 4-3. Interrupt Channel Assignment

INTERRUPT	FUNCTION
IRQ0	ROM-BIOS clock tick function, from Timer 0
IRQ1	Keyboard controller output buffer full
IRQ2	Cascade input for IRQ8-15
IRQ3	Secondary serial port (if present)
IRQ4	Primary serial port (if present)
IRQ5	Secondary parallel printer (if present)
IRQ6	Floppy controller (if present)
IRQ7	Primary parallel printer (if present)
IRQ8	Reserved for battery-backed clock alarm
IRQ9 *	EGA or VGA controller (if present)
IRQ10	Available
IRQ11	Available
IRQ12	Available
IRQ13	80287 Math Coprocessor (if present)
IRQ14	ATA or bus hard disk controller (if present)
IRQ15	Reserved for onboard SCSI controller

* Corresponds to IRQ2 on a PC's expansion bus.

4.3.6 DMA Controllers

Two 8237A compatible DMA controllers are implemented within the board's ASIC devices, resulting in seven DMA channels. The hardware and software implementation of these controllers, as well as the additional address generation logic included, are functionally identical to what is found in a standard AT. The seven DMA channels are utilized on the Little Board/286 as indicated in Table 4-4.

Table 4-4. DMA Channel Assignment

CHANNEL	FUNCTION
0	Available for 8-bit transfers
1	Available for 8-bit transfers
2	Floppy controller (if present)
3	Available for 8-bit transfers
4	Cascade for channels 0-3
5	Available for 16-bit transfers
6	Available for 16-bit transfers
7	SCSI controller (if present)

As in all AT compatible computers, several distinctions exist between the features of DMA Channels 0-3 and Channels 5-7:

- Channels 0-3 -- these four channels are similar to those of a PC, and are used for *8-bit* transfers, between 8-bit I/O adapters and both 8-bit and 16-bit system memory. For these channels, addresses A0-A15 are generated directly by the 8237A compatible DMA controller logic, while addresses A16-A23 are generated by AT compatible DMA *page registers*. These channels can transfer data throughout the AT's 16 megabyte address space, with a maximum block transfer size of 64K bytes.
- Channels 5-7 -- these three channels support 16-bit transfers, between 16-bit I/O adapters and 16-bit system memory. For these channels, A1-A16 are generated by the 8237A compatible DMA controller logic, and A17-A23 are generated by *page registers*. These channels can transfer data throughout the AT's 16 megabyte address space, with a maximum block transfer size of 128K bytes. It is important to note, that *channels 5-7 cannot transfer data on odd byte boundaries, because the address bit A0 is not supported.*

The following two restrictions apply to the use of the board's AT-compatible DMA channels:

Note 1

One restriction of the AT architecture is that addresses during a DMA block transfer *cannot* automatically cross page boundaries (i.e. 64K boundaries for Channels 0-3, or 128K boundaries for Channels 5-7). However, the board's ROM-BIOS disk services handle boundary crossing automatically

Note 2

The DMA controller cannot perform 16-bit to *8-bit* transfers, such as between the 16-bit SCSI interface and 8-bit memory on a bus connected device (e.g. video or I/O controllers, etc.). In such cases, the software must transfer the data in two steps: (1) It must perform a 16-bit DMA transfer between the 16-bit I/O port and 16-bit *system memory*; and (2) it must perform a CPU-controlled block move between 16-bit system memory and the 8-bit bus connected memory. In particular, it should be noted that the board's ROM-BIOS has no knowledge of what type of memory (8-bit or 16-bit) is involved when BIOS functions transfer data between the board's 16-bit SCSI port and memory. Consequently, programs which require BIOS functions to transfer data between a SCSI-connected hard disk and 8-bit RAM on the AT bus (including video controller display RAM) may require modification.

DMA Transfer Speed

The board's DMA controller operates at one-half the CPU speed. Therefore, the DMA controller clock rate is 8 MHz on 16 MHz boards, and 6 MHz on 12 MHz boards. This results in a clock cycle time of either 167 (12 MHz boards) or 125 nS (16 MHz boards). In the AT architecture, DRAM refresh is controlled over the CPU/memory bus. For this reason, all DMA data transfers are generally done using the controller's *byte mode*, so that the required refresh cycles can occur. Consequently, the DMA throughput rate in the AT architecture is limited by both the current activity of the CPU and by the bus transfer latency.

All DMA bus transactions require 5 clock cycles. Adding 2.5 clock cycles for the DMA controller to acquire the bus, and 1.5 cycles for the CPU to regain control of the bus, results in a requirement of 9 clock cycles per DMA data cycle (8 or 16 bits). At 16 MHz, the DMA transfer rate is therefore .89 mega-transfers per second (MT/S), and at 12 MHz it is .67 MT/S. However, there is also a constant DRAM refresh rate that reduces these numbers by approximately 5%, to .84 MT/S (16 MHz) and .63 MT/S (12 MHz). If the transfers are 16 bits, these numbers become: **1.6 megabytes/sec (16 MHz) and 1.2 megabytes/sec (12 MHz)**. 8-bit transfers result in half the throughput.

4.3.7 Programmable Timers

An 8254 compatible timer/counter device is also included in the board's ASIC devices. This device is utilized in the same manner as in a standard AT implementation. Each channel of the 8254 is driven by a 1.190 MHz clock, derived from a 14.318 MHz oscillator, which can be internally divided down to provide a variety of frequencies.

The standard use of this device's timers is summarized in Table 4-5. Timer 3 can also be used as a general purpose timer if the speaker function is not required.

Table 4-5. Timer Assignment

TIMER	FUNCTION
0	ROM-BIOS clock tick (18.2 Hz)
1	DRAM refresh request timing (15 uS)
2	Speaker tone generation time base

4.3.8 Keyboard Interface

A mask programmed microcontroller (U31) functions as an AT compatible keyboard controller. The port supports standard AT keyboards. The keyboard is generally connected by means of a four-wire cable consisting of +5 volts, ground, and two bi-directional signal lines, one for data and one for clock.

A keyboard need not be connected to this port for the system to operate. A flag in the system configuration CMOS NOVRAM (in the battery-backed clock device) can be set so that the ROM-BIOS power on self test will not halt if a keyboard is not present.

Besides generating the keyboard interface, the keyboard controller contains a digital I/O port which is used for a variety of system functions:

- It is used to operate the “A20 Gate” and 80286 CPU reset which are associated with system *protected* mode control, as on other AT compatible computers;
- It provides several control signals for Little Board/286 onboard logic control, including: CPU and bus speed control; EEPROM device control; serial port enables; printer port enable; byte-wide socket enables; mono/color jumper sensing; keyboard disable switch sensing.

4.3.9 Speaker Interface

One of the board's ASIC devices (U28) provides the logic for a PC compatible speaker port. The speaker logic signal is buffered by a transistor amplifier, and provides approximately .1 watt of audio power to an external 8 ohm speaker.

As in the standard PC, the speaker circuit's output frequency is based on two control signals: the output of Timer 2; and the programming of two bits, 0 and 1, in I/O port 61h. Bit 1 of I/O port 61h is one term of a 2-input AND gate in ASIC U28, the other term being the output from Timer 2. Thus, setting bit 1 to a logic 1 enables the output of Timer 2 to the speaker, and a logic 0 disables it. If Timer 2 is disabled (by setting bit 0 of port 61h to a 0), then bit 1 of port 61h can be used to control the speaker *directly*.

4.3.10 Battery-Backed Clock

An AT compatible date/time clock is located within the device at location U29. The device also contains a CMOS static RAM, compatible with that in standard AT's, with the exception of the added Watch Dog Timer function. System configuration data is normally stored in the clock chip's CMOS RAM in a manner consistent with the convention used in other AT compatible computers.

The battery-backed clock can be set using the SETUP. The CMOS RAM within the clock chip is also initialized, along with the board's configuration EEPROM, using SETUP.

One unique feature of the board's battery-backed clock device is that it contains the backup battery *directly within the device!* The battery is rated for a minimum of 10 years of clock and internal CMOS RAM backup under conditions of no power to the board. Extra large soldering "pads" have been used for the clock chip's pins, to make it easy to replace the device when the battery is exhausted.

4.3.11 AT Expansion Bus

An I/O channel compatible with the standard AT Expansion Bus is provided on a pair of dual-row header connectors labeled "J9" and "J10" on the board. J9 is a 64-pin header which contains a PC Expansion Bus plus two added grounds, while J10 is a 40-pin header which carries the added signals that distinguish the AT bus from the PC bus, plus four extra grounds. The signals to the AT Expansion Bus are buffered, and offer approximately 24 mA of output drive current. Signal levels are TTL compatible (inputs and outputs).

The PC bus signals on J9 include an 8-bit bi-directional data bus, 20 address lines, 6 levels of interrupt, three DMA channel handshake lines, a number of other control lines, and power and ground for expansion cards. The first 62 pins of J9 (A1-31, B1-31) correspond to the standard signals on the 62-pin PC bus edgcard backplane; the last two pins of J9 carry extra

grounds for optional use to enhance system reliability. An independent crystal (Y1) provides J9's 14.318 MHz bus OSC signal. The timing of the OSC signal is asynchronous to other expansion bus signals.

For compatibility with standard implementations of the AT bus, four wait states are inserted in all expansion bus transactions for both I/O and memory cycles. *Memory* wait states on the bus can be reduced, in two ways:

- If the *0 Wait State* control signal on J9 is activated by the bus memory card, then no wait states are inserted;
- If a memory card activates the *MEMCS16* line on the AT Expansion Bus, only 1 wait state is used in its access, unless the memory card adds additional wait states.

The AT bus signals on J10 include six high order address lines, the high order data byte, five additional interrupt requests (IRQ's), four additional DMA channel handshake signal pairs (DRQ/DACK's), and several additional control signals. It should be noted that the upper six address lines (LA17-LA23) are unlatched; they can be latched using the expansion bus BALE signal.

4.4 FLOPPY DISK CONTROLLER

A 37C65B floppy disk subsystem controller (U34) provides the equivalent of an entire AT compatible floppy disk controller plug-in card within a single LSI device. This "superchip" integrates:

- Formatter/controller
- Digital data separation
- Digital write precompensation
- Data rate selection
- Clock generation
- Drive interface drivers and receivers
- Drive motor control
- AT compatible control registers

The 37C65B is completely compatible with the industry standard NEC765 floppy controller, and associated external devices, as implemented in a standard AT. Data rates of 125, 250, 300, and 500 Kbits/second are usable.

Two types of dual capacity drives, which allow a single drive to read both 360 Kbyte and 1.2 Megabyte diskettes, are supported:

- **Standard AT drives** -- the data rate shifts between 250K and 500K bps, but the drive rotational speed stays constant.
- **Dual speed drives** -- the data rate shifts between 250K and 300K bps, and the drive rotational speed also shifts between 300 and 360 rpm.

A jumper (W19) is used to select between the two types of dual capacity drives. It must be on for dual speed drive use. It does not affect the use of normal single speed drives.

The board's 37C65B floppy controller contains a highly reliable "2nd order" digital data separator. Write precompensation (187 nS) is also generated digitally within the 37C65B.

Because both of these critical circuits are fully digital, the floppy drive interface is highly reliable and temperature independent.

The board's ROM-BIOS initializes the floppy subsystem configuration based on data stored within the board's Configuration Memory (in the battery-backed CMOS RAM within the clock device). All combinations of supported drives can be used (see Chapter 2).

Interrupt channel 6 and DMA channel 2 are used for normal operation of the floppy controller subsystem. If the floppy controller option is not present, these resources -- and also the floppy controller's I/O port addresses -- are available on the AT Expansion Bus. In fact, a floppy controller card on the AT Expansion Bus could be substituted, provided the onboard floppy controller option is not present.

Refer to Chapter 2 for jumpering information and for the port's connector pinout.

4.5 SERIAL/PARALLEL CONTROLLER

A single high integration component (U22) provides two AT (and PC) compatible serial ports, and an AT (and PC) compatible parallel printer port as well. These devices are identical in functionality with the normal serial and parallel controllers of standard AT's, with one exception: the Little Board/286 parallel port has *bidirectional data lines*.

4.5.1 Serial Ports

The 16C452 device contains two 8250 (16C450) compatible serial controllers. As is the case with standard PC and AT compatible serial controllers, these devices provide two full duplex asynchronous communications channels with software selectable baud rates (up to 57.6K baud), word size (5-8 bits), and stop bits (1, 1.5, or 2).

Each of the two serial ports is equipped with three output and 5 input signals (as in the standard AT). Signals are buffered with EIA compatible RS232C buffers (U15 and U16). U16 is used for the *primary serial channel*, while U15 is for the *secondary serial channel*. Although the Little Board/286 operates from a single +5 volt power supply, +12 and -12 VDC voltages for operation of the RS232C drivers are generated by means of onboard DC-to-DC converters. The -12V power is generated within the RS232C driver devices themselves, while the +12V power is generated by a separate power converter circuit (U8).

Either or both serial ports can be hardware disabled via configuration data within the board's configuration EEPROM (under control of the board's ROM-BIOS). This is normally done using the Ampro SETUP utility. When a port is disabled in this manner, its I/O port addresses and system interrupt become available on the AT Expansion Bus for other devices in the system. For example, an application might require that a modem or alternative serial communications card be substituted for the *primary serial device*.

Each serial port's I/O connector includes a driver output enable signal (J3 pin 10) which is used for factory test purposes only. This signal is pulled low on the board by a 1K resistor and should be left unconnected.

Refer to Chapter 2 for information on the connector pinouts of these two ports.

4.5.2 Parallel Printer Port

The parallel printer port on the Little Board/286 is functionally identical to that found in a standard AT with one exception: its eight data lines can be put in a mode in which they are used to *read* the state of the printer port data lines (rather than being write only). Because you can use this port as a two-way 8-bit bus, it can be very valuable in custom applications.

All data and interface control signals are TTL compatible, and the port can be utilized as a set of *general purpose digital I/O signals* if it is not required for use with a parallel printer. There are many non-printer applications for this port, including using it to control an LCD display panel, scanning keyboards, sensing switches, and even interfacing with optically-isolated I/O modules.

Jumper options allow the port to be configured as either the *primary* or *secondary* printer device, which differ in the particular I/O port base address (378h or 278h) used. Refer to Chapter 2 for jumpering information.

Note

There is no means to alter the printer port's interrupt. Although its I/O port address can be set to the standard LPT1 or LPT2 values, its interrupt remains at IRQ7. DOS does not use the interrupt, while some printer port control software does (such as print spoolers).

The parallel port hardware can be *disabled* by the board's ROM-BIOS at powerup or reset, based on a parameter in the board's configuration EEPROM. The port enable/disable parameter is normally set using the Ampro SETUP configuration utility (see Chapter 3). When the port is disabled in this manner, its I/O port addresses and system interrupt becomes available on the AT Expansion Bus for other devices in the system.

Bidirectional Printer Port Use

A unique "Printer Output Enable" (POE) signal, which can be used to control the data direction of the printer port's eight data lines, is provided at the board's printer interface connector (J3 pin 26). This signal is pulled low by a 1K resistor, R32, which normally keeps the printer port data lines in the *output* direction. To use the printer port data lines bidirectionally, you must remove R32. Note that with R32 removed, the POE signal must be grounded for normal parallel printer port use.

A convenient way to generate the port output enable signal is simple to tie the POE signal (J3 pin 26) to one of the parallel port's output control lines (-STROBE, -AUTOFD, -INIT, or -SEL IN). Ampro recommends using -INIT. Whichever line you use, be sure that your software controls that line in such a way that you get the data direction you want.

In addition to the bidirectional use of the eight data lines, the four control lines (-STROBE, -AUTOFD, -INIT, and -SEL IN) can be used as general purpose output lines. Similarly, the five status lines (-ERROR, SEL OUT, PAPER EMPTY, -ACK, and BUSY) can be used as general purpose input lines.

It is also possible to use the four control lines as input lines. To use a control line as an input line, you must first write to its corresponding bit in the control register. Write a 1 if the line is shown as active high in Table 4-6, Write a zero if it is shown as active low. This will cause the corresponding signal line to "float" (be in a TTL high state). The drivers for these lines are open collector drivers with 10K ohm pullups. When these lines are floating, they can be used as inputs.

In summary, the parallel port can provide up to 12 outputs plus 5 inputs, or up to 17 inputs, as follows:

Data (8 lines) (Read/Write)
 Control (4 lines) (Read/Write)
 Status (5 lines) (Read Only)

Bit 4 in the control register (Table 4-6) can be used to enable interrupts. If this bit is high (equal to 1), then a rising edge on the -ACK (IRQ) line will produce an interrupt.

Table 4-6. Parallel Port Register Bits

REGISTER	BIT	SIGNAL NAME OR FUNCTION	IN/OUT	ACTIVE HI/LO	J3 PIN	DB25F PIN
PORT	0	Data 0	i/o	high	3	2
	1	Data 1	i/o	high	5	3
	2	Data 2	i/o	high	7	4
	3	Data 3	i/o	high	9	5
	4	Data 4	i/o	high	11	6
	5	Data 5	i/o	high	13	7
	6	Data 6	i/o	high	15	8
	7	Data 7	i/o	high	17	9
CONTROL	0	-STROBE	out*	low	1	1
	1	-AUTOFD	out*	low	2	14
	2	-INIT	out*	high	6	16
	3	-SEL IN	out*	low	8	17
	4	IRQ Enable	---	high	---	---
	5	1	---	---	---	---
	6	1	---	---	---	---
	7	1	---	---	---	---
STATUS	0	1	---	---	---	---
	1	1	---	---	---	---
	2	1	---	---	---	---
	3	-ERROR	in	low	4	15
	4	SEL OUT	in	high	25	13
	5	PAPER EMPTY	in	high	23	12
	6	-ACK (IRQ)	in	high	19	10
	7	BUSY	in	low	21	11
* Can also be used as input (see text).						

4.6 SMALL COMPUTER SYSTEM INTERFACE (SCSI)

The board's SCSI bus interface is completely controlled by an AIC-6250 SCSI protocol controller device. The AIC-6250 contains a complete ANSI X3.131 compatible SCSI bus interface, including 48 mA bus drivers, 8-bit data FIFO buffer, and hardware support for bus arbitration, REQ/ACK handshaking, asynchronous and synchronous SCSI data transfer modes, and all required SCSI Initiator and Target functions. (Although the board's *SCSI hardware* has the ability to support for SCSI Target functions, disconnect/reselect, and synchronous data transfers, these functions are not implemented in the current SCSI/BIOS.)

Please refer to the AIC-6250 technical manual for additional technical details on the SCSI controller device:

AIC-6250 Technical Manual

Adaptec, Inc.
691 S. Milpitas Blvd.
Milpitas, CA 95035

For more information on the SCSI interface in general, refer to the American National Standards Institute document, ANSI X3.131-1986 available from:

American National Standards Institute
1430 Broadway
New York, NY 10018

When the SCSI option is present, the AIC-6250's DMA request output signal connects to DMA channel 7. This channel provides a 16-bit DMA data path to system RAM memory on the Little Board/286. Note that, as in the standard AT architecture, this DMA channel cannot transfer data beginning on odd-byte boundaries. When a transfer to an odd-byte aligned buffer is requested by software utilizing ROM-BIOS functions, the block of data is first transferred by the ROM-BIOS to a temporary buffer having an even-byte buffer, and then shifted by a memory block move to the desired odd-byte aligned destination. Therefore, maximum SCSI transfer speed can be realized if buffers are aligned to even-byte boundaries. Otherwise there is approximately 5% degradation in throughput.

A jumper option (W2) allows the interrupt output of the AIC-6250 to be connected to the IRQ15 system interrupt. Though not currently used, IRQ15 is reserved for use by the board's ROM-BIOS and SCSI support software.

Socketed 220/330 ohm bus termination is provided on the board. In addition, a jumper pair can be shorted to provide power to the SCSI "TERMPWR" signal, which is required if external termination networks are used. As recommended by the SCSI specification, a Shottky diode on the board limits the flow of current between the board's +5V power and the SCSI TERMPWR line to be *away from the board*. This prevents system damage from occurring in the event other devices on the SCSI bus are also supplying power to the TERMPWR line.

Note that you can eliminate the board's SCSI *hardware* function by removing the AIC-6250 from its socket on the board. When the port is disabled in this manner, its I/O port addresses and system interrupt and DMA support become available on the AT Expansion Bus for other devices in the system.

Standard Hard Disk Support

Drives connected to the board's Small Computer System Interface (SCSI) can be used to replace the hard disk controller function normally found in a standard AT. Depending on the hard disk configuration parameters stored in the board's nonvolatile configuration EEPROM memory (using SETUP), the board's ROM-BIOS automatically maps hard disk related functions to the SCSI interface, using the SCSI direct access device *Common Command Set* for compatibility with a wide variety of SCSI disk drives. Due to the use of the SCSI Common Command Set, a variety of devices can be used to supply the hard disk function. Other *direct access device* options include: magnetic bubble memory drives, optical storage drives, certain types of tape drives, and solid state disk drives (EPROM, RAM, NOVRAM).

The board's SCSI Initiator ID (0-7), used in all SCSI bus accesses, is normally read by the ROM-BIOS from data stored in the board's configuration EEPROM. The Initiator ID value is normally set using the Ampro SETUP utility, as described in Chapter 3. In normal system configurations, the default Initiator ID of 7 is recommended because the board's ROM-BIOS performs a SCSI bus reset on powerup or system reset when this value is used.

The following restrictions apply to the use of the board's SCSI interface and ROM-BIOS support:

Note 1

One restriction of the AT architecture is that addresses during a DMA block transfer cannot automatically cross page boundaries (i.e. 64K boundaries for Channels 0-3, or 128K boundaries for Channels 5-7). However, the board's ROM-BIOS disk services handle boundary crossing automatically.

Note 2

The board's DMA controller cannot perform 16-bit to 8-bit transfers, such as between the 16-bit SCSI interface and 8-bit memory on a bus connected device (e.g. video or I/O controllers, etc.). In such cases, the software must transfer the data in two steps: (1) It must perform a 16-bit DMA transfer between the 16-bit I/O port and 16-bit *system memory*; and (2) it must perform a CPU-controlled block move between 16-bit system memory and the 8-bit bus connected memory. In particular, it should be noted that the board's ROM-BIOS has no knowledge of what type of memory (8-bit or 16-bit) is involved when BIOS functions transfer data between the board's 16-bit SCSI port and memory. Consequently, programs which require BIOS functions to transfer data between a SCSI-connected hard disk and 8-bit RAM on the AT bus (including video controller display RAM) may require modification.

Ampro SCSI/BIOS

To make it easy to use SCSI devices and functions not supported by the standard Ampro drivers and utilities, functions have been included in the board's ROM-BIOS which allow the use of the SCSI interface without the need for direct hardware programming. This support is accessed through an extension of the INT13 ROM-BIOS support, and is known as the Ampro *SCSI/BIOS*.

The Ampro SCSI/BIOS adds an additional layer of standardization to SCSI, by providing a high level interface for software that eliminates the need for direct programming of the SCSI bus interface hardware. This standard software interface is accomplished through an extension to the INT13 support normally provided in a PC or AT ROM-BIOS for hard disk access. SCSI drivers and utilities written around the Ampro SCSI/BIOS INT13 extensions will run on any system in which the INT13 extensions are available. (This includes all of Ampro's DOS compatible Little Board *single board systems*.)

Detailed information on the SCSI/BIOS functions provided within the Little Board/286 ROM-BIOS is available in application note AAN-8804, available from Ampro.

A SCSI/BIOS disable option is provided within the board's configuration EEPROM parameters. It serves to eliminate the SCSI services *within the ROM-BIOS*. With the SCSI/BIOS disabled, you can still use the AIC-6250 device from your software.

If you plan to support SCSI devices using the SCSI/BIOS, you will probably need to become acquainted with the SCSI bus and its protocols. To obtain a copy of the ANSI SCSI specification, contact the American National Standards Institute referred to previously in this chapter.

Non-SCSI Programmed I/O

When not needed for standard SCSI functions, this interface can be employed as a programmable I/O port. In these non-SCSI applications, the AIC-6250's inputs (from the SCSI connector) provide Schmitt trigger input conditioning, and the outputs (to the SCSI connector) provide open collector drivers capable of sinking 48 mA, and which may be enabled and disabled under software control.

Not all of the SCSI bus interface signals can be used in non-SCSI applications, as the AIC-6250 was intended for SCSI specific use. Sixteen of the SCSI bus signals can be used simultaneously as *inputs*; of these, a maximum of nine can simultaneously be used for *bidirectional I/O*. The available SCSI port output signals can be "bit banged" to generate a wide variety of signal uses and timings; it is even possible to connect the SCSI port to optically isolated AC and DC input and output modules, or to a number of off-the-shelf data acquisition and control interfaces. Information on how to use the Little Board/286 AIC-6250 SCSI interface for non-SCSI programmed I/O is provided by application notes AAN-8901 and AAN-8801, available from Ampro.

Note

You cannot use the board's SCSI interface in this manner if it is also being used for normal SCSI functions.

If you plan to use SCSI interface in non-SCSI applications, you will probably require a copy of the AIC-6250 SCSI Chip technical manual referred to previously in this chapter.

4.7 BYTE-WIDE MEMORY SOCKETS

The two 28-pin byte-wide memory sockets at U32 and U33 support a variety of 28-pin JEDEC pinout memory devices, including EPROM, static RAM, EEPROM, page-addressed EPROM, and nonvolatile RAM (NOVRAM) cartridges. Chapter 2 includes a table indicating the types and capacities of the memory devices each socket will hold. Ampro's solid state disk (SSD) drive support within the ROM-BIOS and optional SSD Support Software treat these sockets as up to two DOS disk devices, containing as much as 1 megabyte of EPROM SSD storage (both sockets combined).

The byte-wide sockets are accessed as *8-bit* devices, each within a separate memory area. Note that the address space of each socket is fixed at 32K bytes, regardless of the type of device being used. This means that when smaller devices or page-addressed EPROM's are used, these devices are "mirrored" (i.e. appear at multiple addresses) within the 32K byte socket window.

The byte-wide sockets can be disabled by the ROM-BIOS at powerup and boot time via options stored in the board's configuration EEPROM (initialized using the SETUP utility). Additionally, the sockets can be enabled or disabled under software control, using a special ROM-BIOS call provided for this purpose. When a byte-wide socket is *disabled*, its address space is available on the expansion bus; but when a socket is *enabled*, its address space is not available on the bus.

The sockets are accessed with four wait states. The required speed of devices used in these locations is 300 nS (or faster), on both 12 MHz and 16 MHz boards.

The following example shows a simple assembly language routine that controls the byte-wide memory sockets:

```
MOV AH,0CDH ; AMPRO function call
MOV AL,nn   ; Use "03" for U32 or "04" for U33
MOV BX,nn   ; Use "00" to turn OFF or "01" to turn ON
INT 13H
```

4.8 CONFIGURATION EEPROM

A great deal of flexibility of system configuration has been made possible by the inclusion of a 1,024 bit serial *electrically erasable* EPROM (EEPROM). The device used is contained in an 8-pin IC at location U14. The configuration EEPROM is normally used in two ways: part of it is reserved for *Ampro setup information*; the rest of it is available for *OEM data storage*.

The 1024 bits of the configuration EEPROM have been allocated in two groups: *Ampro data area*, and *OEM data area*. The Ampro data area is used by the board's ROM-BIOS during powerup and reset initialization. The parameters within the Ampro data area are normally initialized and altered using the Ampro SETUP utility. Included as a feature of the SETUP utility is the ability to display and edit the bits of the OEM data area as well.

If you require the use of the OEM (or other) data area *directly from programs*, you will need to write special software routines to read and write configuration EEPROM bits. The software required to read and write bits to this device are somewhat complex, however; for this reason, functions for reading and writing the configuration EEPROM have been included directly within the board's ROM-BIOS. It is recommended that you utilize the configuration EEPROM BIOS functions, rather than direct hardware programming, to access the device.

Note that the configuration EEPROM is not intended for constant writing. It is specified as having a limitation of 10,000 write cycles, which should be more than enough for its intended use as a configuration storage device. One reason to use the ROM-BIOS routines for configuration EEPROM access is that they automatically maintain a checksum on the contents of the EEPROM.

For details on the use of the EEPROM read/write functions contained within the board's ROM-BIOS, refer to application note AAN-8805, available from Ampro.

APPENDIX A

LITTLE BOARD/286 UTILITIES SOFTWARE

APPENDIX A

LITTLE BOARD/286 UTILITIES SOFTWARE

INTRODUCTION

This Appendix contains detailed information on each of the driver and utility programs supplied on the Ampro Little Board/286 Utilities diskette. Each program's description explains what the program does, and how it is used. Program descriptions are in alphabetical order, so this chapter can serve as a handy reference.

Each program is identified by a version and revision level. When the program is run, its version number (and a revision level) generally appear in a sign-on message such as the following:

**Ampro Hard Disk Format Utility
Copyright (C) 1986 Ampro Computers, Inc.
Version 2.1**

In this case the program is version 2, revision 1. Versions of a program which have the same "version" number operate in the same manner. When a change is made to a program which necessitates a new description, its version number is changed, indicating that the old description is no longer accurate.

Note

Read the contents of the ".DOC" files on your Little Board/286 Utilities diskette (filenames of the form "NAME.DOC"). They contain information on recent program revisions, enhancements, or additions relative to the published user's manual.

Additional programs and features, offered on an on-going basis, will be available to you through software updates, at a nominal charge.