

ZDS-1 Series



For Hardware Development

The ZDS-1 Series provide the tools necessary for the designer to easily debug and test Z80 and Z80A Microprocessor-based hardware. Functions normally supplied by many pieces of special purpose design equipment are integrated in by the ZDS-1 Series Development Systems.

As an Emulator the Development System connects directly to the prototype system and behaves like the Z80 or Z80A Microprocessor, only under direct control of the user, to enable:

- Setting or resetting of all registers and RAM memory locations.
- Setting or resetting of interrupt controls.
- Single or multiple instruction stepping.
- Setting or displaying of I/O Port information.
- Start/Stop control of program execution.

As a Logic Analyzer the Development System monitors and stores pertinent CPU bus activity and terminates program execution in an orderly manner when user defined events occur. Some examples are:

- Storage of address, data and control bus information during each bus transaction.
- Storage of only user defined bus transactions; e.g. Memory read, I/O write.
- Suspension of program execution on occurrence of a user defined event.
- Display a history of the last n bus events, where n equals from one through 256 events.

As a ROM/RAM Simulator the Development System lends its memory to the user's prototype system with the memory mapping features.

- Memory Mapping allows the user to describe the physical nature of the memory to be used by the prototype system. A memory map is defined by the user to specify which addresses exist in the prototype and which address locations are to be "borrowed" from the Development System.
- The ZDS-1/40 Development System also permits memory address to be declared write protected or non-existent so that if any access is attempted a break in program execution will occur.

As a PROM/EPROM Programmer the Development System provides the functions necessary to write, read, verify and duplicate PROMS. This is an optional feature provided by the ZDS/PPB and ZDS/PPB/16 PROM Programmer Option, or ZDS/CIB Printer and Prolog PROM Programmer Interface.

For Software Development

ZDS-1 Series Development Systems provide a standalone microcomputer system utilizing the versatile RIO Operating System for the creation, editing, assembly and debug of the software for the powerful Z80 Micro-processor.

The RIO Operating System with relocatable modules and I/O management is a general purpose computing system with an architecture that is designed to facilitate the development process, provide straight-forward linkage to various systems routines, and enable expansion of system features to meet the particular needs of individual users.

Features

■ OS EXECUTIVE

- ❑ Map requests for operations on logical units to specific device handling programs.
- ❑ Commands may be issued to OS from the console or by an executing program.
- ❑ Any number of user defined commands may be added to the system.
- ❑ Command sequences may be recorded in files and executed as a group.

■ ADVANCED ASSEMBLER

- ❑ Relocatable or absolute object code format.
- ❑ External Symbol references.
- ❑ Global Symbol definitions.
- ❑ Macros and conditional Assembly.
- ❑ Paged symbol table permits assembly of arbitrarily large programs in standard memory.
- ❑ Include directive permits additional files to be merged into the source program at assembly time.

■ LINKER

- ❑ Assigns absolute addresses to program modules.
- ❑ Resolves External references.
- ❑ Permits overlays or memory gaps.
- ❑ Produces memory map and Global address table.

■ TEXT EDITOR

- ❑ Paged work space permits any size of file to be edited.
- ❑ Automatic backup file creation for protection.
- ❑ Access to other files during edit session.
- ❑ All edit operations for locating and modifying lines within a file are based on character string matching.

■ PROM MONITOR

- ❑ Low level device handlers for system console and floppy disc.
- ❑ Bootstrap loader for ease in system initialization.
- ❑ Machine language Debug package.

■ ZDOSII DISC FILE HANDLING PACKAGE

- ❑ Allocates all disc space automatically on an "as needed" basis.
- ❑ Supports sequential access to file or direct access to any specific disc address.

■ TABBING

- ❑ Supported throughout the system.
- ❑ Compact storage of tabbed text on disc.

Utilization of a standalone microcomputer permits all program preparation and verification to take place in the same environment. Smooth and rapid transition from program input, to assembly and linkage to execution and debug provide a rapid and inexpensive solution for software development.

Optional software languages include a BASIC interpreter, a Fortran compiler and Zilog's own powerful PLZ family of system programming language processors.

All software is supplied on floppy diskette media. Each package comes complete with object software, technical and user documentation, and instructions for use with the ZDS-1 System.



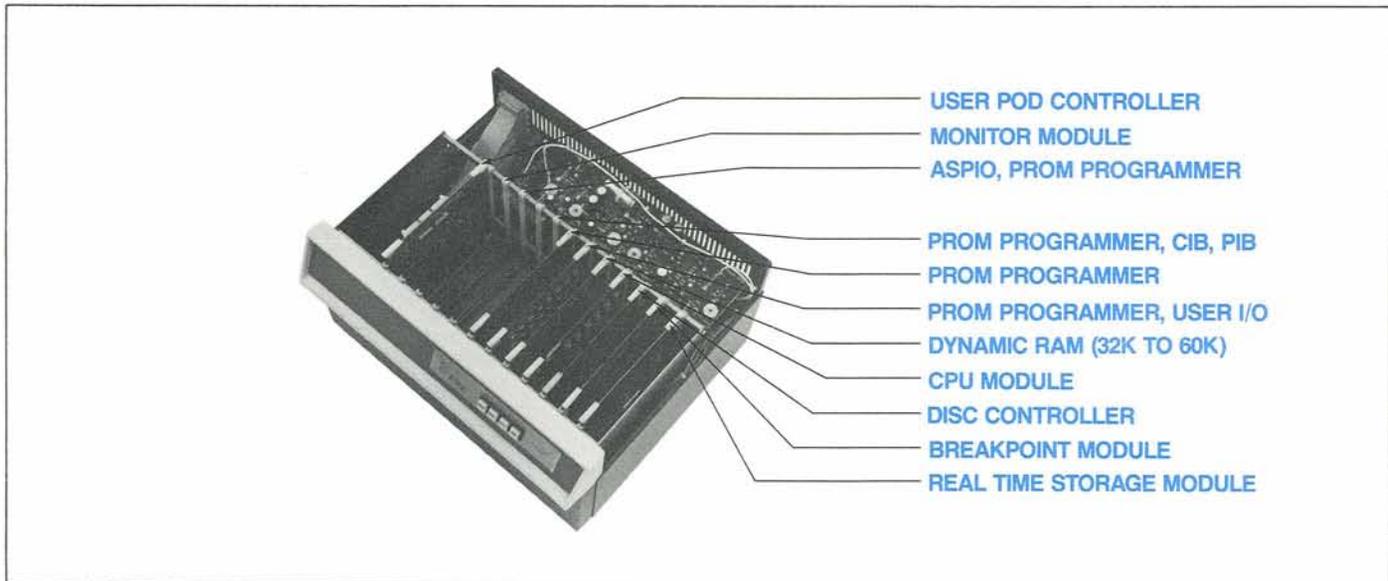
ZDS-1/40 4MHz Development System

The ZDS-1/40 provides a powerful standalone development system for use with the Z80 or Z80A Microprocessor and associated 4MHz peripheral components. Precise emulation is provided by using two microprocessors, a Z80A-CPU which is inserted into the prototype system and a second Z80-CPU which is internal to the development system. This method allows precise emulation up to clock frequencies of 4MHz.

While similar in functional capability to the ZDS-1/25 Development System the ZDS-1/40 provides special features, shown below.

- External Z80A-CPU for direct connection to the user's prototype system.
- Precise emulation for systems employing clock rates to 4MHz.
- Memory mapping and protection in blocks of 1024 bytes.
- User memory refresh, even when an emulation is suspended.
- Verification of user clock integrity.
- Memory address translation.
- Detection of memory accesses to write protected or non-existent blocks.
- New powerful disc-based debug software.





Features

- **Z80A-CPU**—The 4MHz version of the powerful Z80 micro-processor.
- **Main Memory**—Capacity of up to 60K bytes on a single board, allowing more room for I/O options. Standard system includes 60K bytes.
- **Floppy Disc Drives**—Each providing for the storage of up to 300,000 bytes on an inexpensive, removable diskette.
- **RIO Operating System**—With Relocating Assembler, Linker, Text Editor and Logical File Structure.
- **Real-Time Storage Module**—Enables the monitoring of specific address, data and control bus lines during selected operations, e.g. Memory Read, Memory Write, I/O Read, I/O Write.
- **Breakpoint Module**—Enables the monitoring and testing of specific address, data and control bus states to stop program execution or to create a scope sync.
- **Memory Mapper**—Enables the user to describe the physical nature of the memory to be used. The user may utilize the memory in the Development System, his own system or a combination of both in blocks of 1024 bytes. In addition each block may be assigned properties of being non-existent or write protected to aid in the debug and development of the prototype system.
- **Memory Address Translator**—Is an additional feature provided with the Memory Mapper. It enables a block (1024 bytes) of memory in the user system to be physically located at a different address in the Development System.
- **User Clock Integrity**—is verified to insure the clock supplied by the user does not fall below 250KHz.
- **Refresh**—will always occur even when emulation is suspended, to maintain data integrity if dynamic RAM's are employed.
- **I/O Ports**—The user may have access to all I/O Ports when in User Mode and User Clock is selected.

Specifications

CPU

Standard Z80-CPU and Z80A-CPU Emulator.

Memory

3K bytes ROM/1K bytes static RAM dedicated to system monitor; 60K bytes general purpose RAM.

System Clock

Crystal controlled at 2.5 MHz

AC Power Requirement

50/60 Hz, 115 VAC, 200 Watts; optional 230 VAC Power Supply.

Environmental Characteristics

Operating Temperature: 0° to 50°C.

Physical Characteristics

Two separate chassis. One contains the disk drives and power supplies, while the other contains all other elements.

Approximate Weights and Dimensions

Apply to both chassis:
Size: 19"W x 9"H x 15"D
Weight: 35 lbs.

Electrical

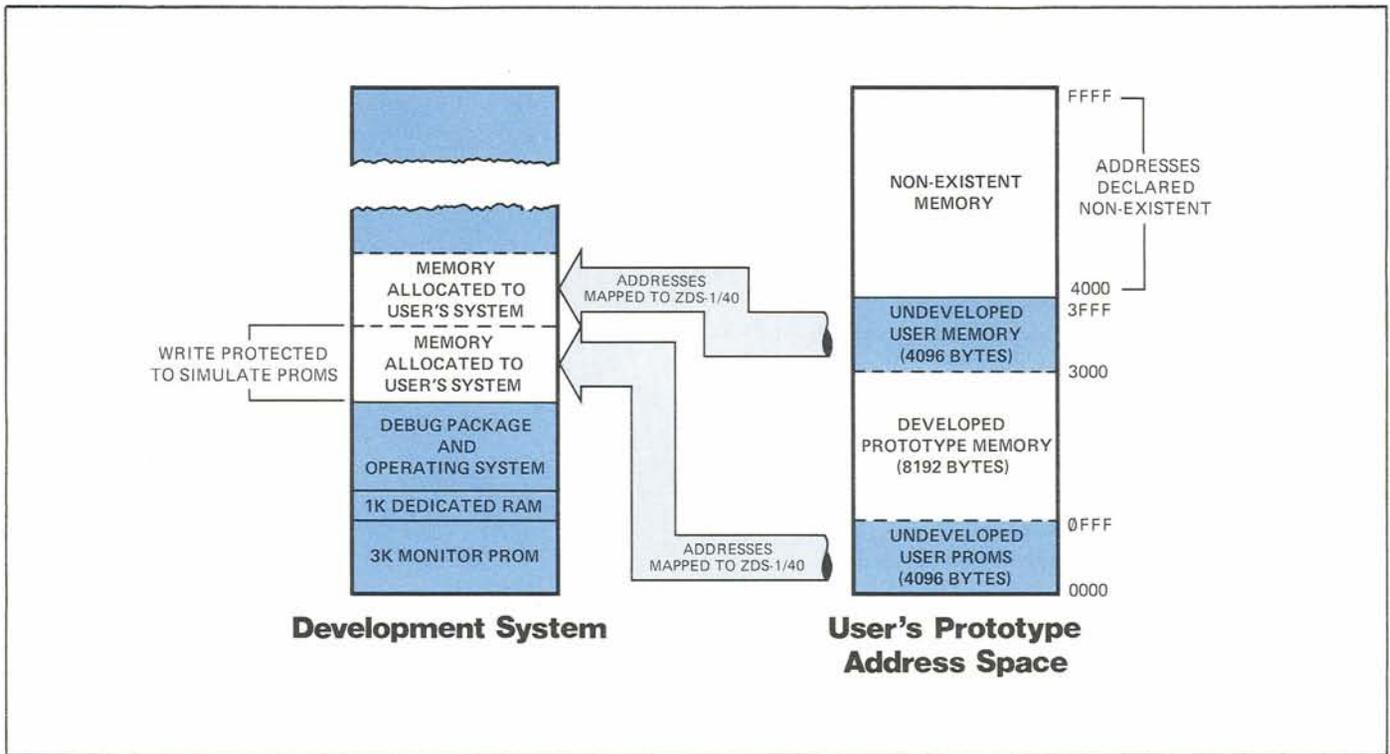
Integral Power Supplies provide all necessary voltages, plus 4 amps of +5V ($\pm 5\%$) is available in the CPU chassis for user cards.

ZDS-1/40 Debug Command Package

The ZDS-1/40 Debug Command Package provides powerful disc based software to support the Z80A-CPU Emulator module. The command repertoire not only includes debug commands for monitoring and analyzing execution of programs under test, but also includes initialization commands to describe the nature of the memory storage being utilized. Once the memory configuration has been established, it may be saved on disc and reloaded as required, thereby eliminating the need to define the memory configuration prior to each emulation session. The following list describes the available commands:

MAP	Creates a map that describes the physical nature of the user's memory address space. This enables the user to define memory in blocks of 1024 bytes to exist in either the user system or the Development System. In addition, the properties of each block may be specified as either non-existent or write protected.	NEXT	Provides a means to single step from 1 to n instructions. All CPU registers are displayed after the selected number of instructions has been executed.
DMAP	(Display Map) —Allows the user to display a single entry, range of entries or the entire map and allows the user to edit the map.	OS	Is used to exit the Debug environment and cause the operating system to be loaded and initialized.
WRITE	Determines whether a break occurs when a memory write is attempted to an address which has the property of being write protected.	PORT	Enables a single character to be input from any selected port and displayed. The user may then enter a data byte to output to the specified port.
BREAK	Is used to specify when the Real-Time execution of a user's program will be terminated. The option fields of the command may establish a break to occur on any combination of the following events: —Address Compare —Data Compare or Compare with Mask —Memory Read or Write —Port Read or Write	PROM	Transfers program control to the resident Debug software contained in the Development System PROM's.
DISPLAY MEMORY	Is used to display the contents of a memory location or group of locations. If single bytes are displayed the user may change the contents of each byte.	PULSE	Is similar to the BREAK command except that program execution is not suspended. Instead a Sync pulse is generated. This pulse is available as a sync for external test equipment.
FILL	Causes a given data string to be stored in all locations specified by the command.	QUIT	Is used to enable return from current program to the calling program.
GET	Loads a memory image file from disc and loads its entry address into the program counter.	REGISTER	Enables the examination and modification of any or all CPU registers.
GO	Begins execution of the user's program. An address may be specified from which program execution will begin. If the program had previously terminated execution due to a break, then the GO command will restore EMULATOR status and resume execution.	SAVE	Enables the contents of memory to be copied onto disc along with the entry address. These files may then be retrieved using GET command. The contents are assigned a file name to permit each of location on the disc.
HISTORY	Enables the display of the last n events stored in the Real-Time Storage Module, where n equals from 1 through 256. Data displayed will be: —Address Bus (16 bits) —Data Bus (8 bits) —Control Bus (7 bits)	SET	Enables the sequential storage of data bytes into memory at the address specified by the user.
INTERRUPT MODE	Provides for the display, setting and resetting of the interrupt mode of the emulator (mode 0, 1, 2 or disable).	STATUS	Displays the emulation status, e.g. interrupt mode, clock source, break and trace arguments.
		TRACE	Allows the user to specify which types of bus transactions will be stored in the Real-Time Storage Module. The user may specify any one or a combination of the following types of operations: —Memory Reads —Memory Writes —Port Reads —Port Writes

Memory Mapping and Address Translator



ZDS-1/40 MEMORY MAPPING

The memory Mapping and Address Translator features allow configuration of the memory used during an emulation. Memory is divided into blocks, each containing 1024 bytes. Under software control, these blocks of memory may be assigned to be existent in either the prototype system, or the ZDS-1/40 Development System, to be non-existent or write protected.

This feature allows the allocation of blocks of development system memory to the prototype system thereby emulating PROM programs or utilizing memory addresses not yet developed in the system.

The address translator allows blocks of user memory to be located around existing software so that resident software will remain unaffected.

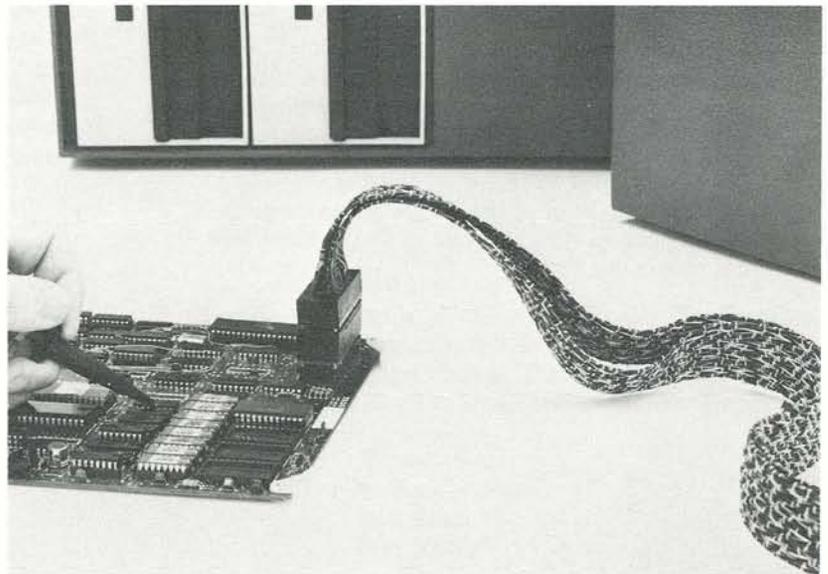
The illustration above shows a typical use of the mapping feature and the address translation feature. Undeveloped PROM's are allocated address space in the development system so that software may be debugged prior to fusing the PROM's. This area has been assigned properties of being write protected to simulate PROM behavior. If a write is attempted, it will be inhibited and a break will occur if the "Break-on-Write" option has been selected. Addresses 4000–FFFF are non-existent and may be declared as such so that any attempted access will also cause a break in program execution.

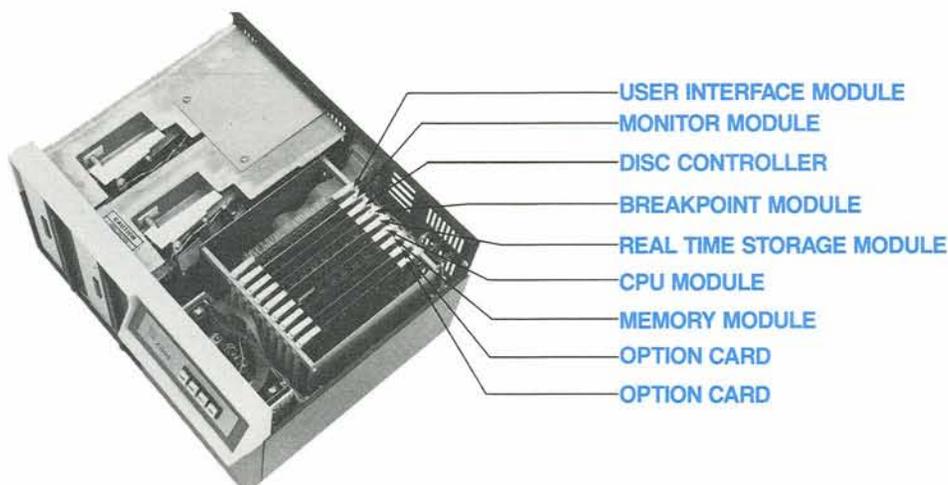
ZDS-1/25 2.5 MHz Development System

The ZDS-1/25 Development System is a low cost standalone development system used with the Z80 CPU for prototype development of systems employing clock rates not exceeding 2.5MHz. This system provides a standalone microcomputer with dual floppy discs, optional peripherals and interfaces to assist the user in the software and hardware development activity of a Z80-based system. Two modes of operation are provided to accomplish this: User Mode and Monitor Mode. In Monitor Mode the system enables the user to utilize the RIO operating system to develop, edit and modify his software. In User Mode, the system memory and peripheral devices may partially or totally be allocated to the prototype system, with the exception of I/O ports E0H through EFH. This enables the user to execute his program in a real time environment.

Features

- 60K bytes of RAM memory.
- 3K PROM Monitor Program with 1K of dedicated Monitor Scratch pad area.
- Programmable Hardware Breakpoint Module enabling the suspension of instruction execution at a given address or activity (Memory or I/O). Programmable Real Time Storage Module to store CPU activity (address, data and control bus), for up to 256 events, to monitor memory or I/O Port activity.
- In-circuit emulator with three foot connection cable to user system.
- Memory mapping to describe the nature of the memory used, either user or ZDS memory. Mapping occurs using blocks containing 256 bytes each.
- Dual Floppy disc drives having a combined storage capacity of 600,000 bytes.
- RIO operating system providing a combination of a Text Editor, Assembler, Linker and ZDOSII File Management System.





Specifications

SYSTEM CHARACTERISTICS:

CPU

Standard Z80-CPU

System Clock

Crystal controlled at 2.5 MHz.

Memory

3K bytes ROM/1K bytes static RAM dedicated to system monitor; 60K bytes general purpose RAM.

AC Power Requirement

50/60 Hz, 115 VAC, 200 Watts; optional 230 VAC Power Supply

Environmental Characteristics

Operating Temperature: 0° to 50°C.

Physical Characteristics

Single chassis, containing the disk drives, disk power supplies, and system logic.

Approximate weight and dimensions:

Size: 19"W x 9"H x 15"D

Weight: 65 lbs.

Electrical

Integral Power Supplies provide all necessary voltages, plus 4 amps of +5V (±5%) is available for user cards.

DISC DRIVE CHARACTERISTICS:

Capacity

Unformatted

Per Disk 3.2 megabits

Per Track 41.7 Kilobits

Transfer Rate 250 Kilobits/sec

Latency (average) 83 ms

Access Time

Track to Track 10 ms

Average 260 ms

Settling Time 8 ms

Head Load Time 35 ms

Functional Specifications

Rotational Speed	360 rpm
Recording Density	3200 bpi
(inside track)	
Flux Density	6400 fci
Track Density	48 tpi
Physical Sectors	32
Index	1
Encoding Method	FM

Physical Specifications

Environmental Limits

Ambient Temperature	= 50° to 100°F.
Relative Humidity	= 20% to 80%
Maximum Wet Bulb	= 78°F.

AC Power Requirements

50/60 Hz ± 0.5 Hz
100/115 VAC Installations
90 to 127 V .4A typical
200/230 VAC Installations
180 to 253 V .2A typical

DC Voltage Requirements

+24 VDC ±5% 1.3A typical
+ 5 VDC ±5% 0.8A typical
- 5 VDC ±5% .05A typical
(option -7 to -16 VDC)

Heat Dissipation

245 BTU/hr. typical

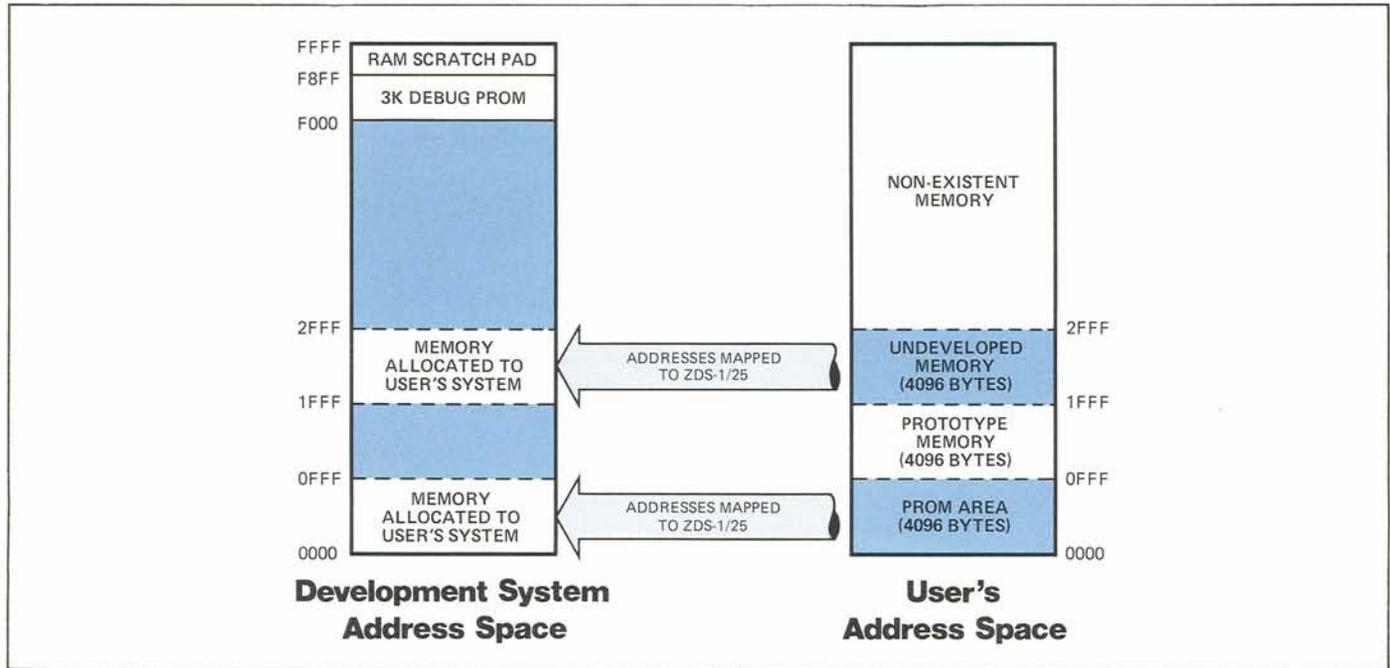
ZDS-1/25 Debug Command Package

The ZDS-1/25 Debug Command Package provides the capability to control, analyze and debug programs which may reside in internal or external prototype memory, or a combination of both. Debug commands are contained in 3K of PROM and use 1K of RAM for a "scratchpad" area. The following is a list of the debug commands and their functions:

BREAK	Sets an automatic hardware breakpoint into the real-time debug module. This break can be on a memory read, memory write, I/O port read, or I/O port write. Addresses, data and data masks can also be specified. A break from the user program can also be caused by pressing the Monitor button on the front panel. In either case, a break causes the state of the user's CPU to be stored so that execution can be resumed later, Control returns to the debug level.	PORT	Permits examination and/or modification of I/O port data.
COMPARE	Allows the user to compare blocks of memory.	PULSE	Is identical to Break except that a pulse is provided, via a BNC connector, each time the specified condition occurs, and the program continues to execute. Pulse can be used to synchronize external test equipment.
DISPLAY	Provides access to memory locations. These memory locations may be displayed as a block or observed one at a time for examination and modification.	QUIT	Returns control to the OS level.
FILL	Allows the user to store a specified data byte throughout a range of memory addresses.	REGISTER	Provides access to the Z80-CPU registers. They may be displayed in their entirety or opened one at a time for examination and change. The register command also allows the user to display and modify the address flag and interrupt mode.
GET	Transfers file images formed by the SAVE command into system memory, ready to be executed by the GO Command.	SAVE	Stores the RAM image of linked programs and subroutines on the user's disk.
GO	Begins execution of the user's program. Execution can begin at any specified address, or it can continue from a previous breakpoint. A programmed or manual break is required to return control back to the debug level.	SET	Stores data entered from the terminal into specified or memory locations.
INTERRUPT STATUS	Allows the user to display and modify the state of the interrupt enable flip-flop.	TRACE	Specifies if memory read, memory write, port read and/or port write conditions are to be stored in the Real-Time Debug Module during execution of the user's program.
HISTORY	Is normally issued after a break from a user program. This instruction lists on the terminal the state of the address, data and control busses of the CPU during the execution of up to 255 bus transactions that occurred in the user's program just prior to a break.		
JUMP	Transfers control to a starting address of program, but the system remains in the Monitor Mode.		
MOVE	Allows the user to transfer a block of memory of any size from any location to any other location.		
NEXT	Executes one or "N" instructions and prints the contents of the CPU registers after each instruction.		



Memory Mapping Features



ZDS-1/25 MEMORY MAPPING

The Memory Mapping Feature allows various segments of memory, called blocks, to exist in either the ZDS-1/25 or in the prototype system. This feature is particularly useful when blocks of the prototype system are composed of ROM, PROM, or EPROM memory. Using the mapper, the blocks of ROM, PROM or EPROM may be described to exist in the ZDS-1/25 RAM memory to facilitate debugging of user firmware, while the remainder of addressable memory may be described to exist in the user's prototype system. Also, the user may choose to map memory blocks into the ZDS-1/25 which have not yet been developed in the prototype system.

The illustration shown portrays a typical memory map used in prototype system development. Mapping occurs using a fixed block length of 256 bytes. The only restriction to using the mapping feature is that the memory in the ZDS-1/25 must exist for the associated block being mapped in the user's prototype system.

Peripheral Equipment

Zilog provides peripheral equipment to enhance the performance of the ZDS systems to get the job done.

ZDS/CRT Terminal

The CRT Terminal provides an alphanumeric display containing 1920 (80 characters/line x 24 lines) with attached keyboard to serve as the system console. Communication between the terminal and the ZDS-1 Series System occurs in a character-by-character mode at transfer rates of 110, 300, 1200, 2400, 4800 or 9600 baud.



Specifications

SCREEN FORMAT

Characters per line	80
Lines per display	24
Character set	96 character ASCII
Character format	5 x 7 dot matrix

MECHANICAL

TERMINAL

Size	12" high, 17" wide, 15" deep
Weight	35 pounds

KEYBOARD

Size	3" high, 17" wide, 8" deep
Weight	5 pounds

ENVIRONMENTAL

Operating Temperature	0° to 50°C.
Storage Temperature	-30° to 70°C.
Humidity	0 to 95% non-condensing

ELECTRICAL

Power Consumption	150 watts
Domestic Power	105-130 volts; 60 Hz
Export Power	105-130, 210-260 volts; 50 Hz

ZDS/Printer Terminal

The ZDS/Printer provides a low cost, highly reliable character printer for use with the ZDS-1 Series Development Systems. This printer features excellent print quality with sharply defined characters. Its small size, mobility and attractive appearance make it well suited to the laboratory or office environment.

Both upper and lower case characters of the full 96-character set are impact printed within a 9 x 7 dot matrix to maintain a printing speed of 120 characters per second.

Bidirectional printing allows the printer to develop the shortest printing path. A read/write memory stores up to 132 printable characters that will comprise the next line, allowing "look ahead" for the next print position. The result is faster throughput and increased reliability.



Specifications

Print Speed	120 character/second (bidirectional) 55 lines/min. (132 characters/line) 250 lines/min. (10 characters/line)
Characters/Inch	10
Characters/Line	132
Lines/Inch Vertical	6
Paper Width (Maximum)	15 inches
Character Set	96 USASCII
Forms Length Selector	2 Channel VFU
Forms Specification:	
Type	Continuous fanfold, edge perforated
Width	4 variable 4 inches to 15 inches
Copies	Original and 5 copies
Weight	15 lb. single part, 12 lb. with 7 lb. carbon multipart
Thickness	0.003 to 0.020 inches
Ribbon	Cartridge with continuous loop (0.25" x 15 yds.)

Hardware Expansion Options

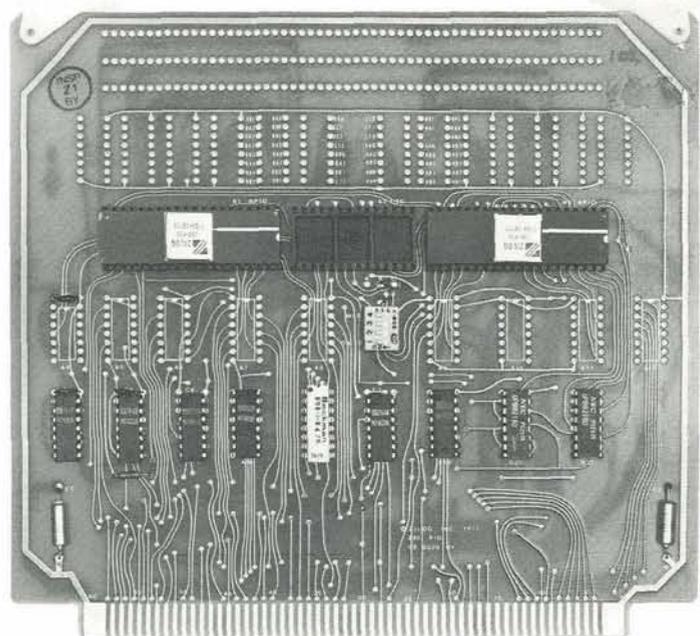
Zilog Offers a variety of hardware expansion features to tailor ZDS-1 Series Development Systems to best suit the users' microprocessor development needs. A standard bus interface structure allows the use of modular boards, each packaged with very high density LSI components for maximum reliability and performance. In addition, software is provided with standard interface to enable immediate use the interface.

ZDS/PIB

PARALLEL INTERFACE BOARD

The PIB provides a basic parallel interface between the ZDS-1 Series Development System and external devices provided by the user. Two Z80-PIO components are provided with supporting logic to give the user 32 bidirectional I/O bits. The card also contains plated through holes for insertion of a Z80-CTC to provide four counter-timer channels. The uncommitted PIO's and/or user supplied CTC can be interfaced with the system's daisy-chain priority interrupt structure. Unused space is provided with 16-pin dip locations (Vcc on pin 16, and GND on pin 8) to allow the addition of logic at the user's discretion. A four-bit dip switch enables I/O port address selection while additional control logic directs port transaction with the system bus.

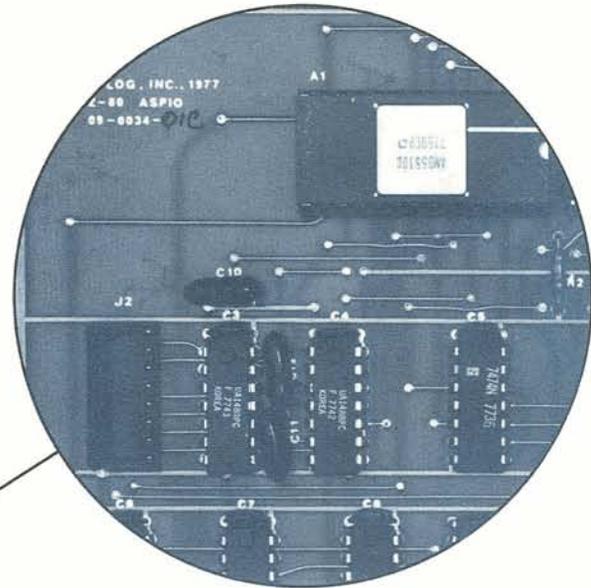
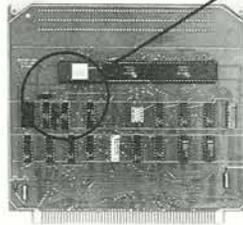
The flexibility of the PIB is demonstrated by Zilog in using it to design our own family of interface boards; ASPIO, CIB and RXB.



ZDS/ASPIO

AUXILIARY SERIAL I/O BOARD

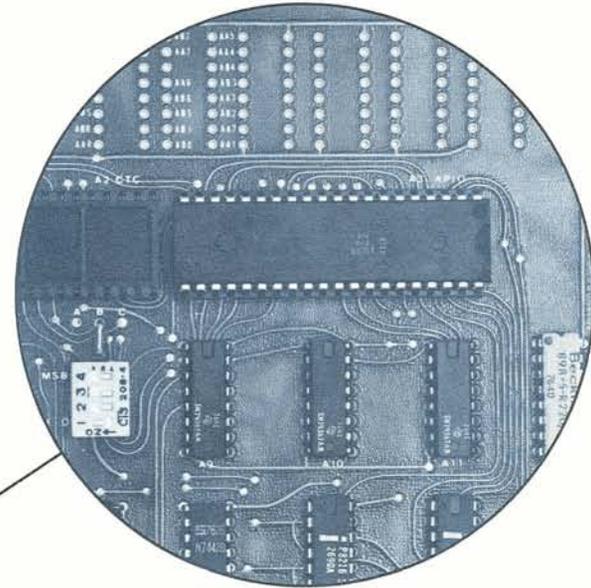
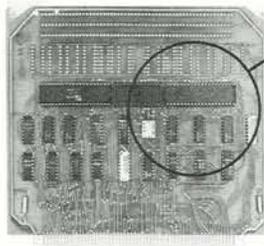
The ASPIO board provides one auxiliary parallel communications interface and one auxiliary serial communications interface for the ZDS-1. A parallel interface is designed for communications with the printer supplied by Zilog. Also a printer employing the interface characteristics of the Centronics Model 306C may be used (e.g. Tally Model 1202 or 1602, Wang 200W). A serial interface, utilizing a Universal Asynchronous, Receiver/Transmitter, is provided to enable synchronous or asynchronous, full duplex communication with external devices employing RS-232C interface characteristics. Utility software is provided for the parallel printer interface and diagnostic (exerciser) software is provided for the serial interface.



ZDS/CIB

CENTRONICS INTERFACE BOARD

Two specialized parallel interfaces are provided, one dedicated to interface with a Prolog 90 Series Prom Programmer and the other for a parallel printer interface previously described under ASPIO. Utility software is provided for each interface eliminating the need for the user to develop his own utilities.



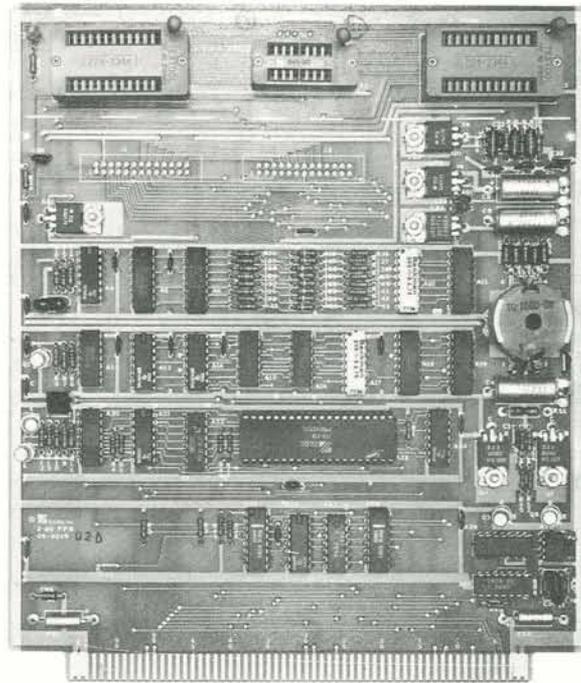
ZDS/PPB PROM Programmer Board

A PPB is a combination PROM/EPROM programmer board which is designed to handle the programming, reading, duplication and verification of the following:

EPROM's — 2704 and 2708

BIPOLAR — 7610, 7611, 7620, 7621, 7640 and 7641

All programming activity is under direct control of the software utility program provided.

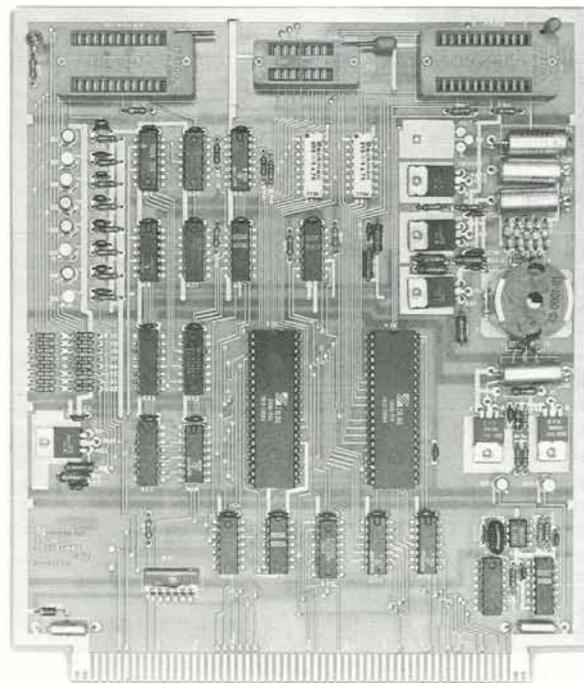


ZDS/PPB/16 PROM Programmer Board

The PPB/16 is a functional equivalent of the PPB, previously described, except it is designed to accommodate the following types:

EPROM's — 2716

BIPOLAR — 82S27, 82S181, 82S131



Software Expansion

In addition to the RIO Operating System Zilog provides a broad range of higher level languages to simplify and accelerate microprocessor software development activity.

BASIC

Zilog's newest Extended BASIC Interpreter sets a new standard in precision and speed performance for the most popular microcomputer language. Two versions of the system—a version of business BASIC with a 13-digit BDC data type—and a scientific version with the conventional 32-bit floating binary data type—guarantee the optimal mix of speed and precision for any application from professional quality business data processing to extensive scientific calculations. In both of these versions, a 16-bit integer data type provides compact storage and rapid computation for status and control variables. Finally, the string data type permits manipulation of variable length text in a uniquely powerful subscripted notation. Both interpreters have extensive access to disk files through Zilog's ZDOSII disk operating system.

Internally, the Zilog BASIC Interpreter has advanced the state of the art in microcomputer language both in its mathematical packages and in its method of variable reference. Transcendental functions supplied in the Zilog BASIC package use the powerful technique of rational approximations to limit the number of arithmetic operations performed in their evaluation while guaranteeing accuracy. The fundamental arithmetic operations all use rounding of results rather than truncation to further enhance system accuracy. The Interpreter itself maintains all variable references in an encoded form to eliminate the need for execution time symbol table searches. As a result of these advances, BASIC provides the convenience of an interpretive environment, speed normally not found in an interpreter and random access to disk files through a sophisticated disk operating system.

FORTRAN IV

FORTRAN has long been accepted as the standard for scientific programming and is the "native" language of many part-time programmers. Zilog supports FORTRAN with a compiler conforming to the ANSI 1966 specification. This level of FORTRAN is commonly referred to as FORTRAN IV.

Full conformance, with the exception of COMPLEX data types, to the generally accepted ANSI standard insures that accumulated libraries of FORTRAN programs will be immediately usable in the ZDS-1 environment. There is no need to translate to a special microcomputer language. Zilog FORTRAN opens the door to the richest traditions of scientific programming in a small, inexpensive environment.

PLZ

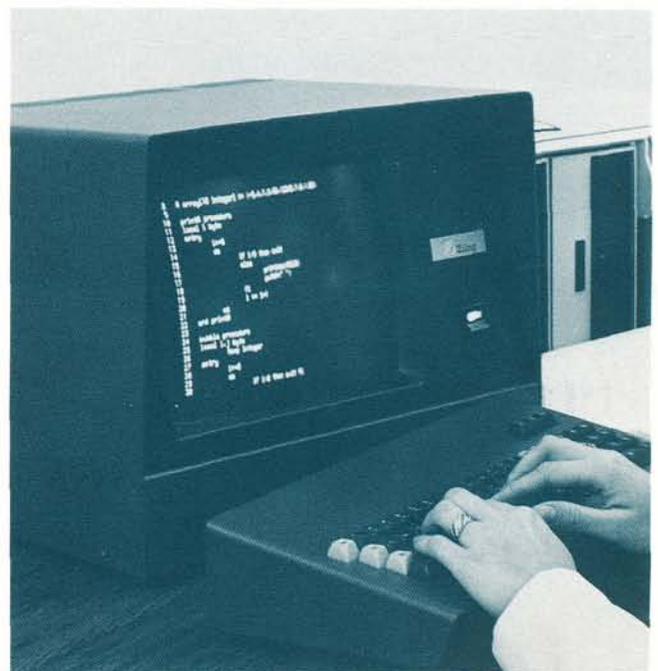
PLZ is Zilog's own family of system implementation languages. Designed specifically with the system programmer in mind, PLZ provides an extremely sophisticated structured programming environment. Implemented at two separate language levels and in distinct translator packages, PLZ offers a complete solution to microcomputer system programming problems.

In its highest level, PLZ is a fully block structured language with an IF-THEN-ELSE clause, a case selector (actually implemented in the syntax of the IF statement), block REPEAT and EXIT statements, procedure references and, of course, RETURN's. Data types include BYTE, WORD, INTEGER, SHORT INTEGER and POINTER as well as ARRAYS of any type and RECORD's composed of groupings of the various basic types, arrays, or even other records.

At this language level, both an Interpreter and a Compiler are available. The interpreter provides a fast and convenient means of program development and debugging, while the highly efficient Compiler generates Z80 Object code.

For applications requiring total control of hardware resources, a second level of PLZ provides all the block structuring and control conventions of the full compiler, but uses assembly language statements rather than compiler level expressions and assignments. Thus, it acts as a PLZ compatible structured assembler and produces code which can be linked directly to modules translated by the compiler.

Regardless of the scope or complexity of a systems programming problem, PLZ will provide a simple, elegant, and complete environment for its solution.



ZILOG SALES OFFICES

<p>EAST</p> <p>Sales & Technical Center Zilog, Incorporated 76 Treble Cove Road North Billerica, MA 01862 Tele: (617) 667-2179 TWX: 710 347 6660</p>	<p>NORTHWEST</p> <p>Sales & Technical Center Zilog, Incorporated 10340 Bubb Road Cupertino, CA 95014 Tele: (408) 446-4666 x261 TWX: 910 338 7621</p>	<p>SOUTH</p> <p>Zilog, Incorporated 2711 Valley View Suite 103 Dallas, TX 75234 Tele: (214) 243-6550 TWX: 910 860 5850</p>	<p>SOUTHWEST</p> <p>Sales & Technical Center Zilog, Incorporated 18001 Sky Park South Suite K Irvine, CA 92714 Tele: (714) 549-2891 TWX: 910 595 2803</p>
<p>Technical Center Zilog, Incorporated Post Office Box 92 Bergenfield, NJ 07621 Tele: (201) 385-9158 TWX: 710 991 9771</p>	<p>MID-ATLANTIC</p> <p>Sales & Technical Center Zilog, Incorporated 110 Gibraltar Road Horsham, PA 19044 Tele: (215) 441-8282 TWX: 510 665 7077</p>	<p>MIDWEST</p> <p>Sales & Technical Center Zilog, Incorporated 890 East Higgins Road Suite 153 Schaumburg, IL 60195 Tele: (312) 885-8080 TWX: 910 291 1064</p>	<p>Sales & Technical Center Zilog, Incorporated 5535 Balboa Boulevard Suite 217 Encino, CA 91316 Tele: (213) 788-2511</p>
<p>WEST GERMANY</p> <p>Zilog GmbH Zugspitzstrasse 4 D-8011 Vaterstetten West Germany Tele: 08106 4035 TELEX: 529110 zilog d.</p>	<p>JAPAN</p> <p>Zilog, Incorporated, Japan Kyoshin Building 13-14 Sakuragaoka-Machi Shibuya-Ku Tokyo 150 Japan Tele: 03-476-3010</p>	<p>UNITED KINGDOM</p> <p>Zilog (U.K.) Limited Babbage House Maidenhead SL6 1DU Berkshire, United Kingdom Tele: (0628) 36131 TELEX: 848609</p>	

10340 Bubb Road, Cupertino, California 95014

03-0004-03

Zilog

Telephone: (408) 446-4666 TWX: 910-388-7621

Printed in U.S.A.
Copyright ©1977 by Zilog, Inc.