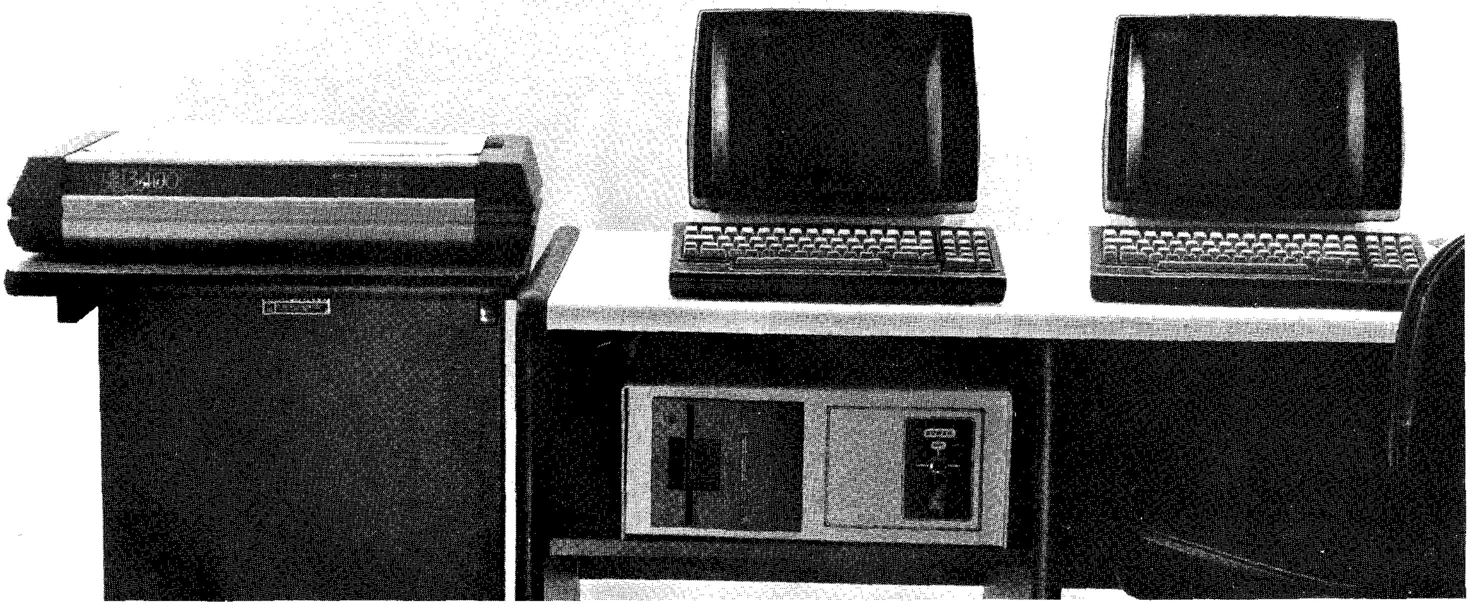
The MDZ™ Computer



Multi-user, Multi-processor, *Fast*

Multi-user - as many as 16 users

Multi-processor - Z80A® or Z80B® for Each User
CP/M® and MicroDoZ® Operating Systems

Thousands of Application Programs Available

Start with a single-user system — add users as your needs dictate.

The MDZ computer, utilizing the MDZ/OS operating system by Micro Mike's, Inc., is a highly versatile and flexible computer system which can be configured to fit the exact needs of the individual or small business.

A company can start with a single-user MDZ/OS system and expand to as many as sixteen users, as needed, simply by adding a slave processor board and a terminal for each user. Expansion is accomplished without the need to scrap the entire system and start all over again. As the size and scope of the system are expanded, the established data base can be maintained and enlarged. Extra disk storage space and printers can be added as needed.

The more expensive peripheral devices, such as printers, hard disk drives, etc., can be shared with other system users, reducing the per-user cost of the system dramatically.

Thousands of Application Programs Available

The MDZ/OS operating system features industry standard CP/M operating system compatibility, supporting many thousands of off-the-shelf application programs, including a wide selection of the most popular database management, electronic spreadsheet/financial modeling and word processing software available.

MDZ/OS systems often qualify for multi-processor software license fee discounts. The per-user cost of most application software is often a fraction of what separate-CPU, single-user software licenses cost.

MDZ/OS, utilizing state-of-the-art microcomputer technology, is compatible with new, lower-cost programs designed for microcomputers rather than the traditionally much more expensive software for minicomputers or mainframes.

A High-Speed Network in a Box

Unlike traditional multi-user minicomputers which share a central processor among all users, a multi-user MDZ/OS system is comprised of one to as many as eighteen separate computers, all contained in the same cabinet. Each user of an MDZ multi-user system has a dedicated Z80 processor. Each user's processor executes its own application program at full speed, without the system speed degradation common to shared-processor minicomputer systems.

The single-board computers are connected to the system through the industry-standard S-100 bus. The high-speed transfer of information between master and slave processors via the S-100 bus is many times faster than that of "netwoked" desktop microcomputers.

Each Winchester hard disk unit may appear to the user as one drive or as many drives. Only a few or as many as several hundred "drives," of equal or varying sizes, can be created on a single hard disk by drive segmentation.

All users can run programs operating through either the CP/M® or MicroDoZ® operating system, e.g., one user might be running WordStar®, while another user is running a program written in baZic® or in NorthStar™ BASIC, while another user may be running dBASE II®, another running SuperCalc®, etc. As many as 16 separate tasks can be executed simultaneously.

As many as thirty-five printers can be supported on a single system. As many as three printers may be shared by all users. Each of the remaining printers is dedicated to a single user.

By entering the number of an option from a password-protected menu system, a user can select as many as seven segments or drives and be up and running an application program.

Versions of baZic, a high-speed, NorthStar-compatible, Z80 BASIC interpreter, are provided for use with the CP/M and MicroDoZ operating systems.

The Best of Both Worlds

MDZ/OS has the best of both worlds - machine language routines where speed is vital and high-level language routines where flexibility and sophisticated programming techniques are important. The run-time portions of MDZ/OS are written in optimized Z80 code for quick operation. These routines are table-driven so that system parameters can be changed quickly and easily from a high-level language. Use of table-driven routines results in increased efficiency because run-time routines have no unneccessary routines to take up extra time and system utilities simply manipulate tables. Utilities then execute at maximum speed. In addition, all inter-processor system level calls can be made from a high-level language, such as baZic.

Because most of the utility programs are written in baZic, the system can be changed easily by qualified programmers to add new features required by special applications. This versatility is not found in most operating systems which strictly structure use of the system. MDZ/OS can be viewed as a general-purpose hard disk operating system and multi-processor communications system whose flexibility lends itself easily to a wide range of tasks.

Hard Disk Drive Segmentation

Under MDZ/OS, a hard disk can be divided into only a few or hundreds of "drives," of equal or varying sizes. These drives, or segments, can be either MicroDoZ or CP/M segments. A user can have as many as seven of these drives on line simultaneously. As a means of visualizing drive segmentation, imagine a set of books on a shelf. Each book can have an equal or a different number of pages. Each book has a Page 1 and, in fact, many pages which have the same page number as other books on the shelf.

Each of these books is a segment (drive). Page numbers correspond to sector addresses. If you want to see Page 200 of the third book on the shelf, you walk down the shelf, bypassing two books, until you come to the third book, whereupon you find the correct page and receive the needed information.

In a similar manner, MDZ/OS is given an address (page number) from one of the servant operating systems (CP/M or MicroDoZ). Utilizing the defined table, it bypasses all the segments (books) not needed until it comes to the proper segment location. The address is then located in the segment (book) specified and the information is then read or written, depending on the action specified.

Following this analogy you should be able to understand how MDZ/OS can support different servant operating systems, each with its own characteristics. By keeping track of the offsets (how many books down the row you need to go to get to the right book), many segments can be created and used by a servant operating system as if it was in its normal environment.

Powerful Locking System

Disk Table Locking

When as many as sixteen users are on the same disk system, locking procedures are vital to maintain integrity of data. Under MDZ/OS, each hard disk in the system may appear as one drive or as many drives. MDZ/OS allows any user to access as many as seven drives simultaneously. To keep track of who has which disk segments, a Segment disk table is stored in the RAM of the master computer. As a user requires the re-assignment of segments, he issues a system call which fetches the disk table from the master. This fetch "locks" the disk table so that other users in the system cannot modify the disk table until it has been unlocked.

Segment Locking

Once the user has fetched the disk table, routines are provided in baZic to compare the disk table with the user's copy, which is stored in the executive portion of each slave. The user may assign segments using the functions provided (which insure that no two users are in the same segment) or the user may manipulate the disk table in any manner before writing the disk table back to the master. Restoring the disk table results in the table being "unlocked" so that other users may access it. This system provides a method for segment lockout.

File Locking

The next level of locking is at the file level. Here the user may call for a file in the normal manner of the language being used and no locking will be performed. However, if multiple users are to be "in" a file at the same time a system level call can be made from the high-level language which will "open" the file for multiple access. At this point MDZ/OS keeps track of the physical drive number, the beginning disk address of the file and the disk offset as read from the disk table. To lock the file from all access, the user simply makes another system call which locks the read or write to the file until the unlock command is issued. Another command is provided to "close" the file and remove the current user as an active user of the file.

I/O Device Locking

The I/O device locks are similar to the user-defined locks in that the system supports a 512-byte block of common memory. In this instance the 51 I/O devices supported by MDZ/OS have two bytes associated with each I/O device. The first byte defines the status of the I/O. The I/O can be defined as (1) not in the system, (2) in the system but not shared or (3) in the system and shared by all the users. The second byte is used in shared I/O devices. This byte is zero if the device is not in use or is the user number of the user which "has" the device. System calls to fetch the I/O lock table can be made from a high-level language and lock the table automatically when fetched and unlock the table when it is restored.

User-defined Locking

MDZ/OS retains 512 bytes of common memory which is also supported by a locking system. Any user may fetch this memory which then locks the memory from access by another user. The memory values are defined by the system programmer so that tasks or communications can be coordinated between users. As an example, the user-defined locks could be used to define a simple record lock where the programmer writes the record number in use into a specified series of bytes within the common memory. Once the memory is written back to the master computer, the lock is terminated and other users have access to the common memory.

Automatic Lock Clearing

All locks in the system are cleared automatically at appropriate times by MDZ/OS. Any system reboot clears all lock tables. Slave reboots clear all locks associated with that user. Changing from

MicroDoZ to CP/M results in the appropriate locks being cleared. CP/M warm and cold boots also clear the appropriate locks without user intervention.

Versatile I/O Assignment

Each of the 51 serial and parallel I/O devices supported by MDZ/OS has its own input buffer within the system. All input buffers are sixteen bytes long and are required because the entire MDZ/OS I/O is interrupt-driven. Output buffers are not used because the system is faster without using them. All I/Os are overlayed and reside in RAM. They are set up initially through baZic programs which guide the user through the definition of I/O devices. Once in RAM, the I/Os may be manipulated directly by each slave, providing the user wide flexibility. Baud rate, USART protocols and CRT handling are also established through baZic programs and are overlayed when a slave computer is booted.

All I/Os can support handshaking lines on input and output. Provisions are made for DSR, Break and a user-defined JuMP intruction which can be executed by the slave. Break can be used to provide a software "reboot" of a slave computer of a user who has "committed hari kari."

One to sixteen slaves are supported, each of which can have two serial I/Os and one parallel I/O, which is either input or output. I/Os are supported for the two serial ports and one parallel port on the SysteMaster.

CP/M Support

A standard version of CP/M-80 is supported within each slave. Switching to CP/M is accomplished through a booting program which loads the slave with the appropriate code, establishes a predetermined I/O and "drops" the user into CP/M.

The Transient Program Area (TPA) is 51.25K if the CCP is not overwritten and 53.25K if the CCP is overwritten. MDZ/OS supports a maximum CP/M segment size of 8 Megabytes and a maximum CP/M file size of 4 Megabytes. As many as 7 segments (drives) may be defined at one time by any user. Users may share common CP/M disk segments, but all but the first user to access the segment are set as Read Only (R/O).

CP/M can be set to boot in several modes. A user may boot directly to CP/M or a specific program can be executed on the cold boot. Alternatively, a warm boot program may be specified which will be executed automatically each time that slave does a warm boot. The CP/M I/O file is defined by an easy-to-use baZic program and is overlayed at boot time.

A version of baZic, a high-speed Z80 BASIC interpreter, is included so that programs written in NorthStar BASIC or in baZic can be run under the CP/M operating system.

MicroDoZ and baZic Versatility

The MicroDoZ and baZic combination is the heart of the MDZ/OS system. baZic is a friendly BASIC interpreter which has the capability of executing the full range of disk operating system commands. baZic is designed to execute programs written in NorthStar BASIC with few or no modifications. The MicroDoZ operating system is faster and more flexible than CP/M and is supported by program development aids as well as a large number of application programs. MicroDoZ segments may be as large as each physical (or subphysical) drive. MicroDoZ files may be more than 16 megabytes long.

A menu system, providing most of the functions necessary to control users within the system, is supported under MicroDoZ and baZic. Each menu item can be password-protected. The menu system provides automatic assignment of segments and I/O devices. Errors are reported if users may not access a particular segment or I/O device.

The MDZ/OS operating system combination includes MicroDoZ; CP/M; versions of baZic for the CP/M and MicroDoZ operating systems; the multi-user, multi-processor hard disk operating system and all related utility programs. Shipped on 8" or 5¼" disk.

Specifications:

3820 Mainframe:

free-standing with casters
IEEE-696 S-100 bus
slide-mounted card cage
20 S-100 expansion slots
all signal lines terminated actively
funnel air flow design
heavy-duty 38 amp sequenced power supply
constant-voltage transformer

dual air fans
funnel air flow design, filtered air
dimensions: 26½" high X 18½" wide X 23½" deep
space for:
two full-height 8" flexible drives
or four half-height 8" flexible drives
and one 5¼" Winchester hard disk

2810 Mainframe:

free-standing or desktop
IEEE-696 S-100 bus
10 S-100 expansion slots
all signal lines terminated actively
dual air fans
funnel air flow design, filtered air
heavy-duty 28 amp power supply
dimensions: 20" high X 13½" wide X 21" deep
space for:
one 8" full-height flexible drive
or two 8" half-height flexible drives
and one 5¼" Winchester hard disk

3510 Mainframe:

desktop
IEEE-696 S-100 bus
10 S-100 expansion slots
all signal lines terminated actively
funnel air flow design
heavy-duty 35 amp power supply
dimensions: 7½" high X 17" wide X 24" deep
space for:
one 5¼" full-height flexible drive
or two 5¼" half-height flexible drives
and one 5¼" Winchester drive

Master Processor Board:

Z80A @ 4MHz
64K RAM
two RS232-C serial ports
one output parallel port
one input parallel port
NEC uPD765AC disk controller
(as many as four 5¼" or 8" flexible disk drives)
Z80A DMA controller
IEEE-696 S-100-compatible
time-of-day clock/calendar

Slave Processor Boards:
(one per user)

Z80A (4 MHz) or Z80B (6MHz) processor
64K RAM, expandable to 128K
two RS232-C serial ports
input/output parallel ports (upgradable to RS-422)
2K FIFO
time-of-day clock/calendar

Flexible Disk Drives:

soft-sectored
5¼" drives - 342K each, formatted
8" drives - 1116K each, formatted

Hard Disk Drives:

5¼" Winchester drives - 16, 52 or 112 megabytes each, formatted

16 Megabyte Drive:

5¼" form factor
20 megabytes, unformatted
16.32 megabytes, formatted
4 platters, 8 read/write heads
MTBF 11,000 power-on hours
MTTR 30 minutes
preventive maintenance - none required
rotational speed: 3600 RPM
recording density: 7700 BPI
average access time: 70 milliseconds
transfer rate: 5 million bits per second

52/112 Megabyte Drives:

5¼" form factor
67/143 megabytes, unformatted
52/112 megabytes, formatted
4/8 platters, 7/15 read/write heads
MTBF 11,000 power-on hours
MTTR 30 minutes
preventive maintenance - none required
rotational speed: 3600 RPM
recording density: 11,155 BPI
average access time: 30 milliseconds
track-to-track access time: 8 milliseconds
transfer rate: 5 million bits per second

Cartridge Tape Drives:

8" form factor
capacity, unformatted: 21.6 megabytes
capacity, formatted: 12 megabytes
MTBF: 5,000 power-on hours
MTTR: 30 minutes
tape speed: 90 IPS
start/stop time: 300 ms
recording density: 8000 BPI
recording tracks: 4-track
recording format: Serpentine
recording density: 11,155 BPI
head format: read while write with separate erase
media: ANSI standard X3.55 - 450/600'

MDZ/OS Operating System includes:

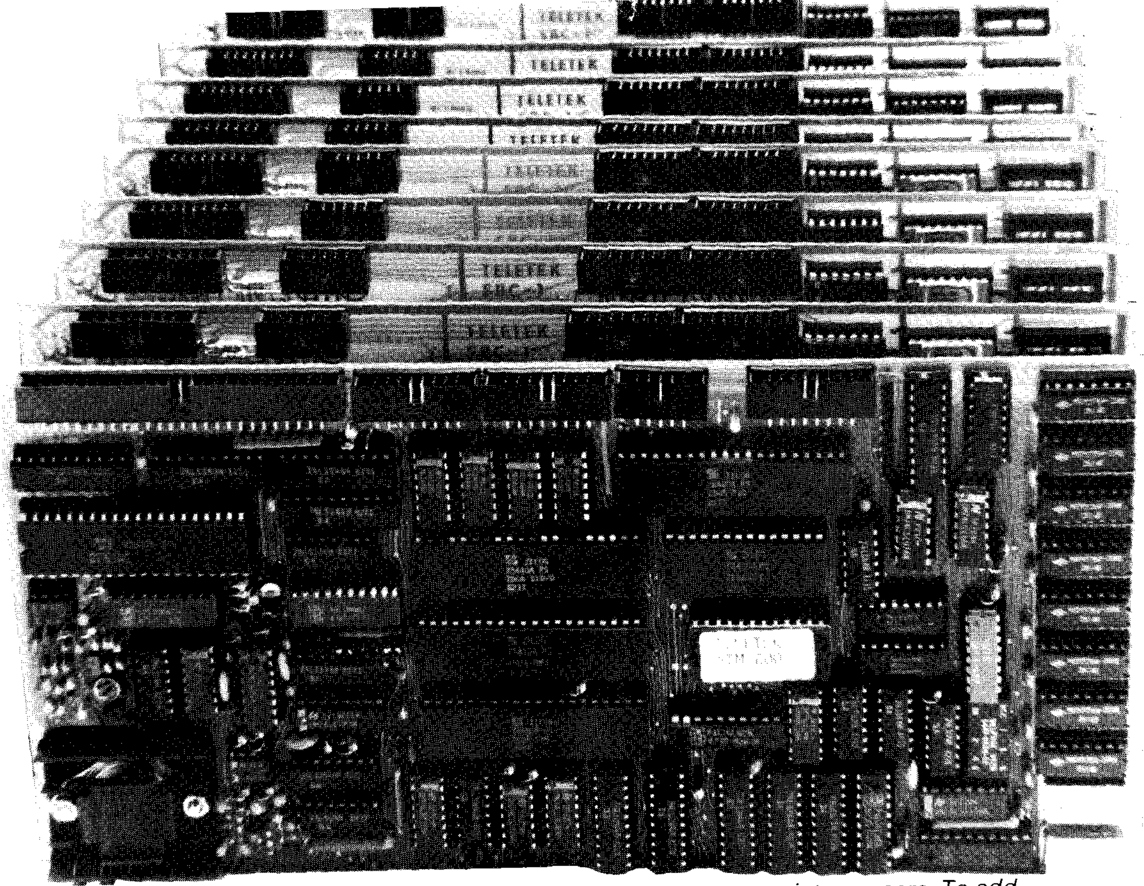
CP/M-80 operating system
MicroDoZ operating system
baZic for CP/M
baZic for MicroDoZ
complete set of utilities

MDZ Systems
P.O. Box 5031
Amarillo, Texas 79117-5031
Phone 806-372-2915

Specifications are subject to change without notice.

CP/M is a registered trademark of Digital Research, Inc.
NorthStar is a trademark of NorthStar Computers, Inc.
Z80 is a registered trademark of Zilog, Inc.
dBase II is a registered trademark of Ashton-Tate.
WordStar is a registered trademark of MicroPro International.
SuperCalc is a registered trademark of Sorcim.
MicroDoZ and baZic are registered trademarks of Micro Mike's, Inc.
MDZ and MDZ/OS are trademarks of Micro Mike's, Inc.
SysteMaster and SBC-1 are registered trademarks of Teletek Enterprises, Inc.

making technology uncomplicated ... for People℠



MDZ Computer systems are expandable to as many as sixteen users. To add a user simply plug in a slave processor board, add cables and a terminal.

The MDZ™ Computer

Multi-user, Multi-processor, *Fast*

Multi-user - as many as 16 users

Multi-processor - Z80A® or Z80B® for Each User

CP/M® and MicroDoZ® Operating Systems

Thousands of Application Programs Available

Start with a single-user system — add users as your needs dictate.