



This article was first published in issues #44 & #45, April/June 1996



CP/M-85 Primer

Ref: CP/M-85 Introductory Guide &
CP/M-85 Reference Guide
from Digital Research Corp.

CP/M is a registered trademark of
Digital Research Corp

CP/M is an acronym for the Control Program for Microcomputers, an operating system that quickly became an industry standard because of its convenience and wide range of practical applications.

"The CP/M Introductory Guide" explained beginning concepts necessary to use CP/M with Heath/Zenith hardware and provided step-by-step procedures for starting up and preparing CP/M software.

"The CP/M Reference Guide" provided a comprehensive description of each command included in your CP/M software package.

THE OPERATING SYSTEM

An operating system is a computer program that controls both the components of your hardware environment and subordinate "application" programs that perform specific tasks.

It provides a vital link between your keyboard and your application programs, and between your application programs and your peripheral hardware. Thus, an operating system is essential whenever an application program is to be used.

The Heath/Zenith CP/M operating system is divided into two software components: the "system kernel" and the "BIOS files". A disk that contains both of these components is said to contain the CP/M Operating System.

CP/M-85 Primer

By Steven W. Vagts
Editor, "Z-100 LifeLine"

Such a disk can also be said to be "bootable", which means that it contains an image of the operating system that can be inserted into the computer and used.

The "System Kernel" is a set of programs that reside on the reserved system tracks of a disk. Special data transfer utilities (usually SYSGEN and/or MVCPM207) are used to copy the system kernel from one disk to another. The system kernel manages files, translates the commands you enter at the keyboard, and performs other functions that do not depend upon specific hardware characteristics.

The "BIOS" stands for Basic Input/Output System, which is the part of the operating system that enables CP/M to work in a Heath/Zenith hardware environment. The BIOS used with this CP/M release consists of the files BIOS85.SYS and BIOS88.SYS, which reside on the file tracks of the disk. The BIOS files can be manipulated by the same methods used to manipulate other files that are stored on the disk. The portion of the BIOS stored in the file BIOS85.SYS uses the 8085 processor of the Z-100. The portion stored in BIOS88.SYS uses the 8088 processor of the Z-100.

To use the operating system, you must transfer a copy of it from a disk to the computer by performing an activity called "bootstrapping", "booting up", or "cold booting". Both the system kernel and the BIOS files MUST be together on a disk for that disk to be usable for booting (the activity that puts CP/M in the computer).

The CP/M Operating System on your distribution software is ready to accommodate a standard configuration of hardware devices. But if your hardware configuration differs from these standards, you must adjust the system before it will accommodate all of your devices. Your distribution software includes utilities to help you adjust the system for your devices.

CAUTION: Your CP/M-85 distribution software contains a serial number to prevent the mixing of this software with software from other CP/M

distribution packages. Therefore, do not try to use this CP/M system with other CP/M systems or utilities, and do not try to use these utilities with other CP/M systems.

APPLICATION PROGRAMS

An application program is a set of instructions that tells your microcomputer how to perform a specific function. An application program might consist of several files that automatically access each other under certain circumstances.

Whenever an application program file is needed to perform a specific task, an image of this file is copied from the disk to computer memory, from where it is executed. When finished, a new application program may be loaded and executed.

CP/M's file naming conventions are identical to those of MS-DOS.

Disk drive names are also assigned as in MS-DOS, with the exception that, whether booting to a floppy drive or to the hard drive, the drive used (the default drive) will always be drive "A" and the prompt is represented as "A>". As you recall, in MS-DOS the floppy drives and hard drives are assigned specific drive letters.

SWITCHING DISKS BETWEEN DRIVES

When you reference a disk drive (by entering a command to a default drive, by changing default drives, or by logging in a non-default drive with a command), the CP/M system remembers some of the characteristics of the disk in the referenced drives. Switching disks between drives can cause problems unless you tell the system to forget the old characteristics.

You can make the system forget about old disk characteristics by performing a "Warm Boot" (by holding down the {CTRL} key and pressing the {C} key. This key sequence is often referred to as {CTRL}-{C} or just CTRL-C, and is usually entered at the A> system prompt. It tells the operating system to forget what it knew about the disks that used to be in the drives. Then it redisplay the system prompt.

Note: This CP/M 'Warm Boot' is not the same as a Z-100 Computer 'Warm Boot', which is completed when you press the key combination, {CTRL}-{RESET}, which then reboots the computer. As other documents from Digital Research may use the term 'Warm Boot', I can not change it. But, just remember the difference.

It is recommended that you perform a warm boot whenever you remove a disk from a drive and replace it with another disk (unless the system prompts you to insert a different disk).

CP/M may be used on one-drive systems through the use of "logical" drives in a manner similar to MS-DOS. It will prompt you when to change disks in the single drive to simulate two drive operations:

PUT DISK x IN DRIVE y: AND PRESS RETURN

Where x is the logical name that is assigned to a particular disk for the duration of the operation; and

Where y: is the physical name given to the single drive, which is temporarily set to read data from or write data to a particular disk.

When such a prompt appears, you swap the disks in the drive as directed. When switching disks in response to this prompt, you do not need to perform a warm boot, just press {RETURN} after the proper disk is inserted.

COMMANDS

There are two kinds of commands that can be executed in a CP/M operating environment: Resident Commands and Transient Commands.

The "Resident Commands" reside within the CP/M operating system; therefore, CP/M does not have to refer to a disk to know how to execute these commands. These commands are:

DIR Displays the file directory of a disk
ERA Erases specified files from a disk
REN Renames a specified file on a disk
SAVE Saves the contents of computer memory to a disk
TYPE Displays the contents of a file on the terminal
USER Enables you to divide the space on a disk into separate areas for different users.

"Transient Commands" are application programs that are supplied with the CP/M Operating System on your CP/M distribution software. These application programs help you to manipulate the operating system and to perform several other useful activities.

These commands or utilities are stored on the system disk as files with the ".COM" extension. When you issue a command that makes reference to one of these files, CP/M copies an image of this file from the appropriate disk into memory, then begins execution of the transient command.

The following list shows the files containing all of the transient commands offered by Heath/Zenith for CP/M Version 2.2.100:

| | | | | | |
|--------|------|----------|------|----------|------|
| ASM | .COM | BSYSGEN | .COM | CONFIGUR | .COM |
| DDT | .COM | DUMP | .COM | DUP | .COM |
| ED | .COM | FORMAT | .COM | LIST | .COM |
| LOAD | .COM | MVCPM207 | .COM | MVCPM217 | .COM |
| PIP | .COM | PREL | .COM | STAT | .COM |
| SUBMIT | .COM | SYSGEN | .COM | XSUB | .COM |

Distribution Disk II contains the source code files used to produce BIOS85.SYS and BIOS88.SYS. These files are supplied for the benefit of assembly language system programmers, and they will not be needed by most users. Many of the files on Disk II are written in 8086 assembly language and require the Digital Research 8086 assembler (not included) for reassembly.

COMMAND LINE ENTRY

The CP/M Operating System is very precise in the way it accepts command lines. You must spell all components of a command line correctly and include the names of non-default disk drives whenever a referenced file is not on the default disk. If you do not, CP/M will not be able to execute your command and will respond by redisplaying the invalid command line with a question mark (?). However, CP/M does allow some flexibility in the way you may respond to the system prompt, as the following special entry explanations show.

Note: In this text, CTRL followed by a hyphen and a letter (CTRL-S, CTRL-C, CTRL-P for example) indicates that you should hold down the key marked CTRL (control key) while pressing the key marked with the letter. To alleviate confusion between typed out commands and pressing certain keys, I shall use the convention of placing brackets around individual key names, e.g., {CTRL}-{C}, {DELETE}, {RETURN}, {ESC}.

The following list explains the single keys and combination of keys that perform CP/M functions:

{BACK SPACE} - Removes the character typed previously. Also removes any "DELETED" characters that were echoed in the line immediately to the left of the cursor. An alternate command is {CTRL}-{H}.

{COLON} - Same as the semicolon, except that you should not begin a comment line with two consecutive colons.

{CTRL}-{E} - Enables you to see the entire display of a command line that is longer than your screen is wide. When you type this entry, the remaining portion of your command line will be displayed on the next line. It is not essential that you enter {CTRL}-{E} when typing a command line that exceeds the console display range because CP/M will process your command line even if it does not fit on one screen line. However, a command line cannot exceed 127 characters in length.

{CTRL}-{H} - Same as "BACK SPACE".

{CTRL}-{I} - Same as "TAB".

{CTRL}-{J} - Same as "RETURN".

{CTRL}-{M} - Same as "RETURN".

{CTRL}-{P} - Causes CP/M to send everything it displays on the console device to the list device [LST:] (normally a printer) at the same time. Making this entry a second time will stop listing the display to the list device.

This entry is useful when you want to record the displays that appear during the execution of a command on paper. {CTRL}-{P} will not work if the list device is turned off, is in a local mode, is off line, or the system is not configured properly to the list device.

{CTRL}-{R} - Redisplays the edited version of a command line below the "scratch pad" version of the line without any of the "DELETED" characters that might have been echoed in the line. Also displays the "#" character at the end of the "scratch pad" version.

{CTRL}-{S} - When these keys are pressed once, the display of data to the console is interrupted or paused; pressing the keys a second time allows CP/M to resume data display. This entry is useful when data scrolls by on a console too quickly for you to read.

{CTRL}-{U} - Effectively removes all characters typed in the command line, and leaves you to try again on the line beneath the old line. It leaves the display of the old command line on the console, and displays the "#" character at the end of this old line to label it as a nullified entry.

{CTRL}-{X} - Removes all characters typed in the command line, as if you used "BACK SPACE" all the way to the beginning of the line.

{DELETE} - Removes the previous character typed from the command line. Depending on how your Operating System is adjusted, the removed characters might be echoed (repeated in reverse) on the video console display, or erased from the display.

{RETURN} - Ends the command line, sends the command to the system for execution, and displays nothing on the console. After execution of the command, CP/M redisplay the system prompt. Alternate commands are {CTRL}-{J} or {CTRL}-{M}.

{SEMICOLON} - Enables you to enter comments not intended for execution without receiving error feedback from CP/M. To cause CP/M to ignore a comment, you must make the ";" the first character entered at the system prompt.

Comments can consist of any characters you wish, typed after the ";" entry, and followed by a {RETURN}.

{TAB} - As with a regular typewriter, this key enables skipping up to eight spaces without pressing the space bar each time. The cursor skips to some column numbered in multiples of eight.

START-UP PROCEDURE

This procedure consists of several steps to help you start up the CP/M Operating System in the Z-100.

Note: The Z-100 computer is equipped with an automatic bootstrap feature via a setting of a DIP switch (SW-101) on the motherboard. This enables the computer to automatically boot to the default drive or partition upon power-up. Check your user's manual to adjust this microswitch as necessary.

1. Place the CP/M Distribution Disk I into the lefthand (Lo-Pro Model) or uppermost (All-In-One Model) floppy drive, and close the drive door.

2. At the hand prompt, you have a choice of several options so that you can boot from any of the drives in your computer.

The bootstrap command syntax is:

```
{B}oot [dev][unit][S][:partition[;system]]
```

Where:

- {B} is the required input that the computer completes with "oot";
- [dev] is an optional function key that determines which type of drive is to be used:
 - {F1} for 5-1/4 inch floppy,
 - {F2} for 8 inch floppy, or
 - {F3} for a hard drive partition
- [unit] is a value from 0-3 identifying the drive from which you want to boot;
- [S] specifies that an optional secondary device controller is to be used;
- [:partition] identifies the partition from which you want to boot up; and
- [;system] is the system name of the boot partition, if the partition name is not unique.

Note: The [partition] and [system] parameters are meaningful only if you press {F3} for the [dev] parameter (the hard drive).

BACKUP PROCEDURES

This procedure requires you to perform the following activities in sequence:

```
Booting Up
FORMAT
DUP
CONFIGUR
```

Note: Users of single-drive systems will have to swap disks between the Distribution Disk and the new blank disk when prompted to do so by the computer.

To begin, boot up with Distribution Disk I in the computer's leftmost or top disk drive. Prepare two blank disks, labeled as Backup Disks I and II, and FORMAT them as follows:

The FORMAT activity helps you prepare blank disks for data storage.

1. At the A> system prompt, type **FORMAT** and press {RETURN}. This entry invokes FORMAT, which displays the following:

```
CP/M-85 Format Version 2.x.xxx
```

```
This program is used to initialize a disk.
All information currently on the disk will
be destroyed. Is that what you want? (y/n):
```

2. Press {Y}. FORMAT will display:

```
Which drive do you wish to use for this
operation?
```

3. Press {B}. FORMAT will display:

```
Number of sides? (1 =single, 2 =double):
```

4. Press {2}. FORMAT will display the following message:

```
Put the disk you wish to be formatted in
drive B. Press RETURN to begin, anything
else to abort.
```

5. Continue to follow the instructions from the computer until FORMAT is complete.

With Distribution Disk I in the drive, proceed with DUP. The DUP utility copies all of the data from one disk to another disk of the exact same type.

Note: Creating a backup copy of your Distribution Disk on a system having only one drive is an extremely lengthy process taking 15 to 20 minutes during which time you will be instructed to replace the disks more than 320 times. Should you mix up the source and distribution disk during this procedure, DUP will display an error message.

1. At the A> prompt, type **DUP** and press {RETURN}. DUP will display the following:

```
Disk Utility Program
Version 2.x.xxx
```

```
Do you want to:
```

```
A copy and verify
B copy only
C verify only
```

```
Z exit to operating system
```

```
Selection:_
```

2. Press {A}. DUP will display:

```
Source unit:
```

3. Press {A}. DUP will display:

```
Destination unit:
```

4. Press {B}. DUP will display:

Put source disk in drive A.
Put destination disk in drive B.

Press {RETURN} to begin:

5. If there is no physical drive B, ignore the "Put destination disk" prompt. You will be prompted when to change the disks in drive A, for example: "Put Disk B in Drive A: and Press RETURN."

6. Continue to follow the prompts of the computer until the operation is complete.

Similarly, duplicate Distribution Disk II.

Proceed with the **CONFIGUR** utility, which adjusts the CP/M Operating System on Backup Disk I for your hardware.

Note: You should skip the CONFIGUR activity and proceed to the "Working Disk Procedure" if you do not have a printer or modem, or if you have one of the following:

* A serial printer that runs at 4800 baud, accepts 8 bits per character with no parity bit, handshakes with RTS pin number 4, is ready when handshaking signal is High, and has no protocol.

* A modem that runs at 300 baud, accepts 8 bits per character with no parity bit, and uses no handshaking.

If you have a printer and/or modem that is different from that described, then begin CONFIGUR:

1. Type **CONFIGUR** and press {RETURN}. CONFIGUR will display the following menu:

```
CP/M-85 System Config. Utility version 2.x.xxx  
Copyright (C) 1982 by Zenith Data Systems
```

```
*** MAIN MENU ***
```

```
P - Printer Configuration  
M - Modem Configuration  
C - Command Configuration  
I - I/O Map Configuration  
? - Brief Help Message  
  
X - Exit
```

```
Selection [P,M,C,I,X or ?]:_
```

2. Press {P} and {RETURN}. CONFIGUR will display the following menu:

```
*** Printer Configuration ***  
  
1 - MX-80 or other PARALLEL Centronics-  
  interface printer  
2 - H/Z-25  
3 - H-14 or TI-810(WH-24)  
4 - Dec LA-34 or LA-36  
5 - Diablo 620  
6 - Diablo 630,1610,1620,1630, or 1640(WH-44)  
7 - MX-80 Serial  
8 - Votrax Type 'n Talk  
9 - User-defined SERIAL Printer
```

Please choose the number that corresponds to your printer:

3. Enter the number to the left of your printer's name and press {RETURN}. CONFIGUR will display a message listing some characteristics of your printer. If these characteristics do not match those of your printer, either change your printer settings or specify characteristics of a user-defined printer. Most printers today are parallel Centronics compatible.

When completed, CONFIGUR will redisplay the "MAIN MENU".

To adjust the system for a printer that is not listed by name on this menu, or a listed printer on which the switch settings have been changed since shipping, press {9} and {RETURN}.

CONFIGUR will then prompt you to specify characteristics of your printer such as: baud rate, bits per character, parity, handshaking pin, handshake polarity and protocol.

BAUD RATE is the speed at which data is transmitted to and from your printer, roughly corresponding to the number of bits per second transmitted. Since the CP/M Operating System coordinates the transmission of data to and from your printer, CP/M must know how fast your printer is set to send and receive data.

BITS PER CHARACTER is the number of significant data bits your printer expects to receive in order to decipher one character (byte) from the stream of bits that are transmitted. Start bits, stop bits, and parity bits are not included in this number. (Most printers accept 7 or 8 bits per character.)

PARITY is a method by which data is checked to make sure it hasn't changed during transmission.

When **odd** parity is used, a parity bit is sent along with each character that is sent to the printer. Before transmission, this bit is set to either one or zero to ensure that the sum of all of the transmitted bits is an odd number. If the printer receives a byte of data bits and a parity bit that do not all add up to an odd number, then an error must have occurred during transmission.

Even parity works the same way, except that the parity bit is set to either one or zero to ensure that the sum of the transmitted bits is an even number.

When **no** parity is used, no parity bit accompanies each transmitted character.

The **HANDSHAKE PIN** is the circuit through which the printer and the computer signal each other to determine when data should be transmitted. This circuit is one of many bundled together in the RS-232C cable that connects the computer with the printer. The printer uses this cable circuit to signal that the printer is ready to receive more data.

The **DTR** (Data Terminal Ready) handshake pin is normally number 20 and the **RTS** (Request To Send) handshake pin is normally number 4.

POLARITY is the voltage level at which the printer signals the computer that the printer is ready to receive more data. This signal can be sent at either a high voltage level or a low voltage level. It travels through the handshake pin that was specified at the previous prompt.

PROTOCOL is a method of coordinating the transmission of data between your computer and your printer by assigning one device or both devices to transmit specific characters as a signal to the other device.

With **XON/XOFF** protocol, the printer transmits the DC3 character (Device Control 3 turns the transmitter OFF) when the printer must momentarily stop receiving data, either because the buffer is full or because the printer is off line. The printer sends the DC1 character (Device Control 1 turns the transmitter ON) when the printer is ready to receive more data, as when the buffer has empty space and the printer is on line.

With **ETX/ACK** protocol, the computer transmits the ETX character at the end of the transmission of a unit of data (to signify the End of TeXt). When the printer processes the unit of data (buffer size) with the ETX character, the printer transmits the ACK character back to the computer (to ACKnowledge that the data unit was received). If this option is selected, you will be asked to enter the device buffer size. Enter a number between 10 and 255 to reflect the largest amount of data that your printer can handle.

In order to adjust the system to accommodate your modem, press {M} and {RETURN} at the "Configur Main Menu". Then CONFIGUR will display one or more prompts that help you to specify either the model number or characteristics of your modem. The first modem prompt looks like this:

*** Modem Configuration ***

Standard Heath/Zenith Modem?
(WH-13, WH-23, WH-33, WH-43) [Y/N]:_

Your response to this prompt depends on whether your modem is listed in the prompt by its model number. Pressing {Y} and {RETURN} assumes the factory settings.

Pressing {N} and {RETURN} causes CONFIGUR to prompt you for the specific characteristics - baud rate, bits per character, parity, and handshaking pin - in a manner similar to the serial printer above.

Automatic Command Specification adjusts the system to accommodate your preference in automatic commands. Pressing {C} and {RETURN} causes CONFIGUR to display:

*** Command Line Configuration ***

C - Cold Boot Command Line =
W - Warm Boot Command Line =
? - Brief Help Message

X - Exit

Selection [C,W,?, or X]:_

This menu enables you to specify a command that could be automatically invoked upon a Z-100 System's Cold or Warm Boot, each independently.

Any valid resident command, transient command (utility), or application program is acceptable in the automatic command line. However, any file that the command line refers to must reside on the disk specified in the command line. For example, if the line reads:

C - Cold Boot Command Line = B:SC

Then the file "SC.COM" (part of the SuperCalc application program) must reside on the disk in drive "B" for the command to work. If any data referenced in the command line cannot be found in the specified drives, then command execution will be aborted and an error message will be displayed.

Note: Some transient commands (utilities) and application programs finish their execution by performing a system Warm Boot. Therefore, you should not enter both an automatic cold and warm boot command line for a program that performs a warm boot after execution, because an endless execution of the same program might result.

Finally, this menu enables you to display a screen of helpful comments about this menu and its use, if you enter a {?}.

Press {X} and {RETURN} to exit.

I/O DEVICE SPECIFICATION

In order to adjust the system to send certain types of data through different physical devices, enter {I} and {RETURN} at the "Configur Main Menu". Then CONFIGUR will display the menu:

*** I/O Map Configuration ***

C - Console Currently: CRT
R - Reader Currently: PTR
P - Punch Currently: PTP
L - List Currently: TTY
? - Brief Help Message

X - Exit

Selection [C,R,P,L,? or X]:_

This menu shows the physical device names that are currently assigned to each logical drive category.

Note: If you have already chosen a printer during this CONFIGUR run, then the List device will have been automatically assigned to the proper physical device.

EXPLANATION

Data can be sent to or from a wide variety of peripheral hardware devices. CP/M groups these devices under the logical device categories "Console", "Reader", "Punch", and "List" (as shown on the left side of the menu).

However, these categories provide CP/M with only a general reference to the actual device being used. CP/M is capable of controlling data transfer through many different devices that are included in these categories.

Therefore, a more specific name must be assigned to each logical device category to inform CP/M of the kind of devices it must control. The specific name assigned to each logical device category is the "physical device name". Each physical device name helps CP/M to recognize and control a particular kind of peripheral hardware.

The physical device name currently assigned to each logical device category is listed on the right side of the menu.

Note: Pressing {?} will cause a screen display of helpful comments.

The following table matches physical and logical device names:

| Logical Device Cat: | Physicl Device Name: | Description and/or Catalog Name of the Actual Hardware Device Used on the Z-100: |
|---------------------|----------------------|--|
| C Console | TTY: | A printing terminal attached to serial port outlet A. |
| | CRT: | A video display terminal & keyboard. |
| | BAT: | A batch pseudo-device using RDR: for input and LST: for output. |
| | UC1: | A modem attached to serial port B. |
| R Reader | TTY: | A printing terminal attached to serial port outlet A. |
| | PTR: | Not implemented. |
| | UR1: | A modem attached to serial port B. |
| | UR2: | A video display terminal & keyboard. |
| P Punch | TTY: | A serial printer attached to serial port outlet A. |
| | PTP: | Not implemented. |
| | UP1: | A modem attached to serial port B. |
| | UP2: | A video display terminal & keyboard. |
| L List | TTY: | A serial printer attached to serial port outlet A. |
| | LPT: | A parallel printer attached to the parallel port. |
| | CRT: | A video display terminal & keyboard. |
| | UL1: | A modem attached to serial port B. |

To select a logical device category to be changed, you must first type the letter [C, R, P, or L] listed to the left of the category in the menu. CONFIGUR will display one of the following prompts, respectively:

```
Console: [T(ty), C(rt), B(at) or 1(UC1)]:_
Reader:  [T(ty), P(tr), 1(UR1), 2(UR2)]:_
Punch:   [T(ty), P(tp), 1(UP1), 2(UP2)]:_
List:    [T(ty), C(rt), L(pt), 1(UL1)]:_
```

Each of these prompts show the logical device category on the left, and all of the physical device names that could be assigned to this category on the right. To assign a physical device name to a logical device category, type the single character of the name that is displayed outside the parentheses.

Note: These logical/physical device assignments can be temporarily changed outside of CONFIGUR by using the STAT utility.

EXIT

Pressing {X} and {RETURN} at the Main Menu enables you to exit from the CONFIGUR utility. If you have specified any changes to the system during this CONFIGUR activity, CONFIGUR will display the following menu:

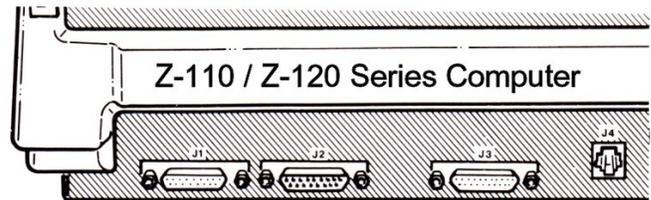
```
*** EXIT OPTIONS ***

T - Make changes temporary (to memory only)
P - Make changes permanent (to memory & disk)
Q - Make no changes

? - Brief Help Message

Choice [T,P,Q, or ?]:_
```

Before returning control to the CP/M Operating System, CONFIGUR will display a diagram of the Z-100's back panel, showing which devices are plugged in at each port.



| | | | |
|---|---|--|-------------------|
| J1 - DCE DB-25(F) Serial Printer | J2 - DTE DB-25(M) Serial Modem | J3 - Parallel DB-25(F) Centronics Printer | J4 - Light Pen |
|---|---|--|-------------------|

If you have a printer, test it by pressing {CTRL} and {P} keys simultaneously, then press {RETURN} a few times. Your printer (if properly configured and connected) should print system prompts just as they are displayed to the video screen. Press {CTRL}-{P} again to discontinue this printer test.

WORKING DISK PROCEDURES

A working disk is a disk that contains both a customized CP/M Operating System and useful application programs.

Most users find working disks to be convenient, if not essential, because combining the operating system and application programs on the same disk makes it quicker and easier to access vital programs and/or data.

This procedure requires you to perform the following activities in sequence:

Booting Up FORMAT SYSGEN PIP

As with the previous Backup procedure, prepare one or more blank disks, labeled as needed, and FORMAT them.

Use the SYSGEN utility to copy the CP/M Operating System to the working disk from the Backup Disk.

1. At the A> prompt, type **SYSGEN** and press {RETURN}. The SYSGEN utility will display the message:

```
CP/M-85 SYSGEN VER 2.x.xxx  
SOURCE DRIVE NAME (OR RETURN TO SKIP):
```

2. At this prompt, press {**A**}. SYSGEN will display:

```
SOURCE ON A, THEN TYPE RETURN
```

3. Press {RETURN}. SYSGEN will display:

```
FUNCTION COMPLETE.  
COPY BIOS88.SYS & BIOS85.SYS (Y/N):_
```

4. Type {**Y**}. SYSGEN will display:

```
FUNCTION COMPLETE.  
DESTINATION DRIVE NAME (OR RETURN TO REBOOT):_
```

5. Type {**B**}. SYSGEN will display:

```
DESTINATION ON B. THEN TYPE RETURN_
```

6. Press {RETURN}. If you are using a one drive system, you will be prompted when to swap disks in the drive. Follow SYSGEN's prompts carefully until you are returned to the System Prompt.

With Backup Disk I in the drive, proceed to PIP.

The **PIP** activity will help you to copy application program files from Application Program Disks to your working disk.

1. Type **PIP B:=A:PIP.COM[V]** and press {RETURN}. Note the difference between the format of this command and that of the COPY command in MS-DOS. Here the destination disk is put first, similar to BASIC's LET command, LET B=A, let the files of disk B equal those of disk A.

2. Insert "Disk A" (Backup Disk I) and "Disk B" (Working Disk) as prompted, until the system prompt appears.

3. Reset the computer (Warm Boot) by pressing {CTRL}-{RESET}.

4. Boot up with the new working disk.

5. Type **PIP A:=B:{filename.ext}[V]** and {RETURN}, where {filename.ext} is the name of a file you wish to copy from the Application Program Disk to the working disk. The usual "?" and "*" wild cards work as in MS-DOS.

6. Repeat step 5 as necessary for other application files.

GENERIC CP/M ON THE Z-100

CP/M-80 software was widely used throughout the small business computer industry on computers that use the 8080 or Z80 microprocessor, such as Heath/Zenith's H/Z-88, H/Z-89, H/Z-90 computers. The H/Z-100 was built with the intent of retaining this CP/M capability, allowing a person to upgrade their computer hardware without the need to purchase all new application software - at significant savings.

Heath/Zenith CP/M software was distributed on 5-1/4" hard-sectored disks, 5-1/4" soft-sectored disks, and 8" soft-sectored disks, depending upon the computer system. As the Z-100 uses soft-sectored disks, the files contained on most CP/M-80 disks must first be converted to a Z-100 compatible format, especially if you wish them to be bootable.

There are three main procedures to conduct this conversion:

PROCEDURE ONE - files from 5-1/4" (48 TPI) or 8", soft-sectored disks.

PROCEDURE TWO - files from 5-1/4", hard-sectored, 48 TPI disks.

PROCEDURE THREE - files from 5-1/4", soft-sectored, 96 TPI disks.

PROCEDURE ONE - files from 5-1/4" (48 TPI) or 8" (if the Z-100 is so equipped), soft-sectored disks - may be used directly on the Z-100 without modification. However, you will not be able to boot from these disks, as the Z-100 can not use the CP/M-80 operating system. You will need a CP/M-85 working disk, or hard drive partition, containing the following files:

FORMAT.COM, BSYSGEN.COM, and PIP.COM.

To simplify the procedure, I'll assume you have CP/M-85 installed on your Z-100 on a hard drive partition and have booted up to it. This is drive A and the first floppy drive is drive C. If you only have a dual floppy system, the drive you boot to is drive A, and the second drive is drive B, instead of C.

These procedures also assume you are transferring files to the Z-100's 5-1/4" drives only.

[] Label a sufficient number of blank disks for all your copying needs and format each in drive C by typing:

FORMAT C: [N] {RETURN}

Note: Brackets {} around a word simply mean to press the single key whose name is in the brackets, for example {RETURN} means to press the RETURN key only. {CTRL}-{C} means to press a two key combination - first press and hold the {CTRL} key, press the {C} key, then release both.

The Z-100 Computer will format the 5-1/4" diskette for double-density use on both sides. If you do not use the [N] switch in the FORMAT command, you will be cautioned about losing all data on the disk and then be prompted to continue. Format all of your blank disks now.

[] Prepare each of your new disks with BSYSGEN. BSYSGEN is the CP/M utility that puts the operating system on each disk. To do so, insert the newly formatted blank disk in drive C and type the following:

BSYSGEN C:=A: [B,N] {RETURN}

BSYSGEN will copy the CP/M system files, ALTCHAR.SYS, BIOS88.SYS and BIOS85.SYS, from drive A to the blank disk in drive C (caused by the B switch). Again, the N switch eliminates the prompts. Do this with all of your new disks now.

[] Copy the Peripheral Interchange Program (PIP.COM) to each Disk. Insert the new Disk in drive C and type the following:

PIP C:=A: PIP.COM [V] {RETURN}

PIP will copy the program (itself) to each new disk. (If you wish to copy additional CP/M programs (files) to the new disk, use PIP again, changing the appropriate file names.

[] Transfer files from the CP/M-80 disks to the new CP/M-85 disks using the PIP command:

PIP D:=C: *.* [V] {RETURN}

CP/M uses the wild card, *.* , to transfer all the files from one disk to another. In one floppy drive systems, you will be asked to swap destination and source disks at the appropriate time (many times). In two floppy drive systems, the wild card *.* causes all the files from disk C to be copied to disk D, except any system files that may exist. The switch [V] verifies the data exchange.

[] If you have more files to transfer to the CP/M-85 system, repeat the process using another new CP/M-85 disk.

Note: If you are copying files from an 8" disk to a 5-1/4" disk, all the files may not fit on a single 5-1/4" CP/M-85 disk. If this is the case, an error message will be generated and you will

have to split the files from your 8" disk among two or more 5-1/4" disks.

You have completed procedure one for transferring CP/M-80 files to run on your Z-100.

PROCEDURE TWO

To transfer CP/M-80 software on 5-1/4", hard-sectored disks, you will first have to transfer it to 5-1/4", 48 TPI (tracks per inch) soft-sectored disks and then to Z-100 disks (only if you wish to boot from them). To do this, you will need the following equipment:

* A H/Z-89 or H/Z-90 Computer.

* A H-88-1 disk controller card installed in the Z-89 with at least one 5-1/4", 48 TPI hard-sectored disk drive connected.

* A Z-89-37 controller card installed in the Z-89 with at least one 5-1/4", 48 TPI soft-sectored disk drive connected to it.

* CP/M-80 (version 2.2X03, 2.2.03, or 2.2.04) working disks properly configured for the system that contains FORMAT.COM, SYSGEN.COM, and PIP.COM.

* A Z-100 Desktop Computer.

* CP/M-85 configured working disk containing the following files: FORMAT.COM, BSYSGEN.COM, and PIP.COM.

* 5-1/4", soft-sectored, blank disks in sufficient quantity to accommodate at least twice the number of files you wish to transfer.

To transfer the software, do the following:

[] Prepare labels for about half of the 5-1/4" disks, using the names of the files that you will transfer to them, and the name "Conversion Disk".

[] Boot the Z-89 system using CP/M-80.

[] Format each of your Conversion Disks. To do so, insert a blank disk in the 5-1/4", 48 TPI soft-sectored drive and type the following:

FORMAT x: [DD,2S,N] {RETURN}

Where x: is the soft-sectored drive letter for your system, DD specifies double density, 2S specifies double sided, and N eliminates the prompts. Using the same procedure, format all of your Conversion Disks.

[] Place the operating system on each of your Conversion Disks. To do so, insert each disk in the drive and type the following:

SYSGEN {RETURN}

When the following is displayed:

SOURCE DRIVE NAME (OR RETURN TO SKIP):

press the key that corresponds to your source disk. At the next prompt,

COPY BIOS.SYS (Y/N):

press {Y}. The computer then displays:

DESTINATION DRIVE NAME (OR RETURN TO REBOOT):

Press the key that corresponds to the name for that drive. This will be the same drive name you used for the FORMAT command. Use this same procedure and put the system on the rest of your Conversion Disks.

[] Place the Peripheral Interchange Program (PIP) on each of your Conversion Disks with PIP. Use the proper drive names for the source and destination. (They will be the same as you used in the previous step.) Do this for each of your Conversion Disks.

[] Copy your CP/M-80 software to the Conversion Disks. To do so, reboot your system to the Conversion Disk. This becomes drive A. Place each of your CP/M-80 software disks in the hard-sectored drive, now drive B, and use PIP to copy the files to the Conversion Disk(s):

PIP A:=B:*.*

Note which CP/M-80 software files you transferred to the Conversion Disk.

You have completed your work with the Z-89 Computer. Follow Procedure One to complete the transfer process (if you want Z-100 bootable disks) of your CP/M-80 files.

Note: Whenever Procedure One calls for a CP/M-80 disk, use your Conversion Disk instead.

PROCEDURE THREE

CP/M-80 software supplied on 5-1/4", soft-sectored 96 TPI (tracks per inch) disks must be transferred to 48 TPI disks for use on the Z-100. You will need the following hardware and software:

* A H/Z-89 or H/Z-90 Computer

* A Z-89-37 disk controller card installed in the Z-89; one 5-1/4", 48 tracks per inch disk drive connected to one port of the Z-89-37 disk controller card; and one 5-1/4", 96 TPI disk drive connected to the other port of the Z-89-37 controller card.

* CP/M-80 (version 2.2X03, 2.2.03, or 2.2.04) with PIP.COM, SYSGEN.COM, and STAT.COM, properly configured for the two different types of drives.

* A Z-100 Desktop Computer.

* CP/M-85 configured as a working disk containing the following files: FORMAT.COM, BSYSGEN.COM, and PIP.COM.

* Enough 5-1/4", soft-sectored, blank disks to

accommodate at least twice the number of files you wish to transfer.

In spite of the fact it is not a supported configuration, you can transfer files from 96 TPI disks with little or no difficulty once CP/M-80 is properly configured.

To transfer the CP/M-80 software perform the following steps:

[] Boot the Z-89 system using CP/M-80 from the 5-1/4", 96 TPI disk drive.

[] Format each of your Conversion Disks. To do so, insert a blank disk in the drive and type the following:

FORMAT x:[DD,2S,N] {RETURN}

Where x: is the 48 TPI soft-sectored drive letter for your system, DD specifies double density, 2S specifies double sided, and N eliminates the prompts. Using the same procedure, format all of your Conversion Disks.

[] Place the operating system on each of your Conversion Disks. Enter the following:

SYSGEN{RETURN}

When the following is displayed:

SOURCE DRIVE NAME (OR RETURN TO SKIP):

press the key that corresponds to your source drive. At the next prompt,

COPY BIOS.SYS (Y/N):

press {Y}. When the following is displayed:

DESTINATION DRIVE NAME (OR RETURN TO REBOOT):

press the key that corresponds to the name for that drive. This will be the same drive name you used for the FORMAT command. Use this same procedure and put the system on the rest of your Conversion Disks.

[] Place the Peripheral Interchange Program (PIP) and STAT Program on each Conversion Disk with the PIP program. Use the proper drive names for the source and the destination (they will be the same as you used in the previous step). Do this for each Conversion Disk.

[] Copy your CP/M-80 software to the Conversion Disks using PIP. Since you will probably not be able to fit all the files on one Conversion Disk, carefully use the "?" and "*" wildcards to limit the number of files transferred at a time. Note the files transferred to each Conversion Disk. Do this for all of your CP/M-80 software disks.

You have completed your work with the Z-89 Computer. Use Procedure One to complete the transfer process of your CP/M-80 files. When Procedure One calls for a CP/M-80 disk, use your corresponding Conversion Disk.

CP/M-80 TO CP/M-85 CONVERTIBLE SOFTWARE

The following list of CP/M-80 software products may be converted and/or run on the Z-100 Computer while you are using CP/M-85.

SUPERCALC: A work sheet planning program used for viewing and calculating business data. Performs math operations on all relevant statistics throughout the 630 column, 254-row range of the work sheet. Provides screen-displayed help messages. Distributed on CP/M-80 disks as:

HSC-817-1 5-1/4", Hard-sectored
HSC-837-1 5-1/4", Soft-sectored
HSC-847-1 8", Soft-sectored

GENERAL LEDGER II: A low-cost accounting program for small business bookkeeping. Enables you to enter a custom chart of accounts. Verifies equality of all double-entered debits and credits. Displays balance statements. Requires BASIC-80 Interpreter. Distributed on CP/M-80 disks as:

SF-9104 5-1/4", Hard-sectored
SFS-9104 5-1/4", Soft-sectored

SMALL BUSINESS INVENTORY: An inventory program that displays and prints twelve useful statistics describing up to 1,000 parts that you stock. Distributed on CP/M-80 disks as:

SF-9105 5-1/4", Hard-sectored
SFS-9105 5-1/4", Soft-sectored

MICROSTAT: A flexible, menu-driven program for statistical analysis and data manipulation. Performs math operations, time series analysis, and several other functions. Distributed on CP/M-80 disks as:

HES-817-1 5-1/4", Hard-sectored
HES-837-1 5-1/4", Soft-sectored
HES-847-1 8", Soft-sectored

WORDSTAR: A word processing program with advanced text editing/printing capabilities and screen-displayed help messages. Distributed on CP/M-80 disks as:

HMP-817-2 5-1/4", Hard-sectored
HMP-837-2 5-1/4", Soft-sectored
HMP-847-2 8", Soft-sectored

MAILMERGE: A WordStar supplement that enhances WordStar's printing capabilities by enabling you to print form letters merging variable text and to automatically print a number of files in succession. Requires WordStar. Distributed on CP/M-80 disks as:

HMP-817-3 5-1/4", Hard-sectored
HMP-837-3 5-1/4", Soft-sectored
HMP-847-3 8", Soft-sectored

MAGIC WAND: A word processing program with video screen text editing and form letter printing capabilities. Distributed on CP/M-80 disks as:

HRS-817-9 5-1/4", Hard-sectored
HRS-837-9 5-1/4", Soft-sectored
HRS-847-9 8", Soft-sectored

Note: Some Magic Wand functions are performed by pressing keys that are different on the Z-100. Refer to the "Magic Wand Function Key Guide" in this document.

SUPERSORT: A data organization program that quickly sorts and combines data from several different files. Accepts input data in many different record formats. Provides screen-displayed help messages. Distributed on CP/M-80 disks as:

HMP-817-1 5-1/4", Hard-sectored
HMP-837-1 5-1/4", Soft-sectored
HMP-847-1 8", Soft-sectored

DATASTAR: A data organization program that enables you to generate forms for entering data, to store the data in files, and to retrieve the data quickly. Can perform math operations on data. Provides screen-displayed help messages. Distribution on CP/M-80 disks as:

HMP-817-4 5-1/4", Hard-sectored
HMP-837-4 5-1/4", Soft-sectored
HMP-847-4 8", Soft-sectored

WORDMASTER: An economical text editing utility that enables you to compose text files and store them on disk for program development and/or correspondence. Distributed on CP/M-80 disks as:

HMP-817-5 5-1/4", Hard-sectored
HMP-837-5 5-1/4", Soft-sectored
HMP-847-5 8", Soft-sectored

DESPOOL: A time saving utility that coordinates printing operations. Distributed on CP/M-80 disks as:

HDR-817-3 5-1/4", Hard-sectored
HDR-837-3 5-1/4", Soft-sectored
HDR-847-3 8", Soft-sectored

SID/ZSID: A debugging utility that enables you to display and change portions of an assembly language program by accessing assembly mnemonics in source files or hexadecimal addresses in object files. Distributed on CP/M-80 disks as:

HDR-817-2 5-1/4", Hard-sectored
HDR-837-2 5-1/4", Soft-sectored
HDR-847-2 8", Soft-sectored

MACRO ASSEMBLER: An advanced assembler utility that makes program development and assembly easier, but still uses the assembly language designed for the ASM utility. Includes many useful macro libraries. Distributed on CP/M-80 disks as:

HDR-817-1 5-1/4", Hard-sectored
HDR-837-1 5-1/4", Soft-sectored
HDR-847-1 8", Soft-sectored

MACRO-80: Fast and versatile utilities for assembling 8080 or Z80 source code, building libraries of relocatable modules, linking modules from several languages into executable object code, and cross referencing variable names and line numbers. Distributed on CP/M-80 disks as:

HMS-817-52 5-1/4", Hard-sectored
HMS-837-52 5-1/4", Soft-sectored
HMS-847-52 8", Soft-sectored

CBASIC: A non-interactive BASIC programming tool that includes a pseudo-code compiler and a run-time interpreter. Includes programs to convert source files into executable object files and a program to list all variables used in a source file. Distributed on CP/M-80 disks as:

HCM-817-1 5-1/4", Hard-sectored
HCM-837-1 5-1/4", Soft-sectored
HCM-847-1 8", Soft-sectored

PEARL III: A menu driven program generator that enables users without programming experience to design and produce custom application programs that can manipulate up to three files at a time. Prompts the user to specify what the program should do, and converts these specifications into a C-BASIC source file. Distributed on CP/M-80 disks as:

HCP-837-1 5-1/4", Soft-sectored
HCP-847-1 8", Soft-sectored

MAGIC WAND FUNCTION KEY GUIDE

Some of the features of the Magic Wand word processor are performed on the Z-89 by pressing keys that do not exist on the Z-100 keyboard. This guide will help you to determine which keys are available for these functions on the Z-100.

The following table shows the Z-100 keys you can press to duplicate the Magic Wand functions you performed on the Z-89 terminal. And, if these are not satisfactory, you can remap the Z-100's character set to rearrange these.

~~~~~  
**MAGIC WAND FUNCTION KEY GUIDE**

| <u>Z-89 Key</u> | <u>Z-100 Key</u>        | <u>Function</u>                                                 |
|-----------------|-------------------------|-----------------------------------------------------------------|
| f1              | F1                      | backward line scroll                                            |
| f2              | F2                      | forward line scroll                                             |
| f3              | F3                      | backward page scroll                                            |
| f4              | F4                      | forward page scroll                                             |
| f5              | F5                      | page feed                                                       |
| blue            | F6                      | search/replace                                                  |
| red             | F7                      | repeat search                                                   |
| white           | F8                      | block marker                                                    |
| IC/7            | DCHR/ICHR               | toggle char insertion mode on or off                            |
| DC/9            | Shift-DCHR/ICHR         | delete word                                                     |
| IL/1            | DEL LINE/INS LINE       | toggle insert on or off                                         |
| DL/3            | Shift-DEL LINE/INS LINE | delete line                                                     |
| Up Arrow        | Up Arrow                | cursor up one line                                              |
| Down Arrow      | Down Arrow              | cursor down one line                                            |
| Left Arrow      | Left Arrow              | cursor left one character                                       |
| Right Arrow     | Right Arrow             | cursor right one character                                      |
| HOME            | HOME                    | cursor to left of line, then to top or bottom of screen on left |

~~~~~

CP/M COMMANDS AND UTILITIES

The following is a brief description of each of the CP/M-85 Commands. For more information, please see Appendix A.

ASM

Utility that creates an Intel Hex File and a Print-Out File from an Assembly Language Program File.

BSYSGEN

Utility that copies CP/M between disks.

CONFIGUR

Utility to customize CP/M for your hardware and/or preferences; described in depth earlier in this document.

DDT

The Dynamic Debugging Utility, similar to MS-DOS Debug Utility.

DIR

Directory Listing Resident Command.

DUMP

Utility to display a Hex File.

UP

Utility to duplicate and/or verify entire disks.

ED

Line editing utility that creates and edits text files.

ERA

Resident Command for erasing files.

FORMAT

Utility to prepare a disk's surface for data.

LIST

Utility that sends text file contents to the printer.

LOAD

Utility that loads a Hex file for execution.

MVCPM207

Utility that customizes a CP/M System Kernel for memory size. MVCPM207 is for floppies, MVCPM217 is for the Winchester Hard Drive.

PIP

Utility to copy data between files, disks, and/or hardware devices.

PREL

Utility that creates a relocatable file from two Hex output files.

REN

Resident command that renames files.

SAVE

Resident command that copies data from memory to a disk file.

STAT

Utility that reports disk statistics and assigns status.

SUBMIT

Utility that triggers automatic execution of CP/M commands.

SYSGEN

Utility that transfers the operating system from memory to a disk.

TYPE

Resident command that displays file contents on the console.

USER

Resident command that controls user access to disk areas.

XSUB

Utility that batches commands within utility programs for automatic processing.

If you have any questions or comments, please email me at:

z100lifeline@swvagts.com

Cheers,

Steven W. Vagts



Appendix A:

CP/M-85 COMMAND SUMMARY

ASM.COM

The ASM utility reads assembly language source files (files with the "ASM extension) from a disk and translates them into two output files:

* An output file with the "HEX" extension, containing 8080 machine code in Intel hexadecimal form. This file can be loaded into the computer (using the LOAD utility) and executed under the CP/M Operating System.

* An output file of ASCII characters, with the "PRN" extension, that can be printed out or displayed on a console screen.

Invocation: **A>ASM {Filename}.{x}{y}{z}**

Where {**filename**} is the primary name of an assembly language source file whose extension is assumed to be "ASM"

Where {**x**} is the drive letter where the assembly language source file is located

Where {**y**} is the drive letter where the "HEX" output file will be written

Where {**z**} is the drive letter where the "PRN" output file will be written

Example: A>ASM PROGRAM.ABB

Notes:

- * CP/M assumes that the source file has the extension "ASM". Therefore, the "ASM" extension does not have to be entered in the invocation command.
- * If a Z is entered in the place of a drive letter for the "HEX" or "PRN" output file, that file is not produced.
- * If an X is entered in place of a drive letter for the "PRN" file, the listing is displayed at the console. To send such a display to the printer as well, enter {CTRL}-{P} before invoking ASM.
- * If you omit any of these parameter letters from the invocation command line, the ASM utility will try to read from, or write to, the default drive.

BSYSGEN.COM

The BSYSGEN utility is used to transfer either part or all of the CP/M operating system to a disk, depending on the circumstances. Unlike the SYSGEN utility, it can not be used to copy the system kernel directly from memory to a disk after running the MVCPM207 utility, although it can copy a file that was recorded on a disk by the SAVE command after a run of MVCPM207.

Invocation: **A>BSYSGEN or A>BSYSGEN {dest}={source}{[options]}**

Where {**dest**} is the name of the drive (A:, B:, C:, or D:) containing the formatted disk that you wish to receive the copy of the system

Where {**source**} can be either a drive name, a file name, or both

Where {**[options]**} represents one or more letters enclosed in square brackets and separated by a comma to specify how the BSYSGEN operation should be conducted:

B - The ALTCHAR.SYS, BIOS88.SYS, and BIOS85.SYS files are copied from the specified source to the specified destination.

N - No prompts will be displayed during this operation.

Example: A>BSYSGEN A:=B:[B,N]

CONFIGUR.COM

The CONFIGUR utility helps you to change the CP/M Operating System so that it will accommodate a particular printer or modem. It also enables you to set the system to automatically invoke commands upon cold boots and/or warm boots, and to assign physical devices to logical devices.

CONFIGUR is usually run during the first session of CP/M use in a particular hardware environment. But it should also be run whenever a hardware component is added or changed, or whenever you wish to change an automatic command line.

Invocation: **A>CONFIGUR**

DDT.COM

The Dynamic Debugging Tool (DDT) utility enables you to debug machine language programs (files with the extension "HEX" or "COM"). DDT loads a program into the Transient Program Area of the computer's memory. During a debugging session, the utility exposes and manipulates the hexadecimal, assembly language, and ASCII forms of the loaded program when you implement the special DDT commands. You can copy the results of the debugging session to a disk file by exiting from DDT and implementing the SAVE resident command.

The DDT commands are:

| | | |
|----------|--------------------|---|
| A | Assembly: | Assembly language mnemonics are inserted |
| D | Display: | Display memory contents in Hex and ASCII form |
| F | Fill: | Fill a block of memory with a specified data constant |
| G | Go: | Go to specified address to run a program |
| H | Hex: | Hexadecimal computation of sums and differences |
| I | Insert: | Insert file name into file control block |
| L | List: | List assembler mnemonics of a program |
| M | Move: | Move a data block to a different memory area |
| R | Read: | Read a program file into memory from disk |
| S | Substitute: | Substitute hexadecimal values at an address |
| T | Trace: | Trace program execution |
| U | Untrace: | Untrace program execution |
| X | eXamine: | Examine or change registers or flags |

Invocation: **A>DDT or A>DDT {filename}**

Where {**filename**} is the complete name of a program file residing on the default disk with a "HEX" or "COM" extension. This file is automatically loaded into the computer's memory beginning at address 100H (the Transient Program Area start).

DIR

The DIR resident command is issued to determine the presence of:

- all of the files on a disk
- a specified file
- a group of specified files

After command entry, DIR displays file names to the console, except that the system files are not accessed and listed. Files assigned to an unlogged user area are also inaccessible to a DIR command unless a USER command is issued before the DIR command.

Invocation: **A>DIR {drive}{filename}**

Where {**drive**} is the drive whose contents are to be listed
Where {**filename**} is the file being searched for on a certain drive

Examples:

```
A>DIR                lists files on default drive
A>DIR B:            lists files on drive B:
A>DIR B:*.BAK       list all the .BAK files on drive B:
```

DUMP.COM

The DUMP utility is invoked with the name of a file to display the Hex contents of each address in a file. The file contents appear in lines containing 16 bytes of data each. To the left of each line is the address of the first byte in each line. This display will continue to the end of the program, unless suspended using {CTRL}-{S} (press any char to resume) or aborted using a {CTRL}-{C}, which executes a warm boot (Not system warm boot; only a disk warm boot) and aborts the program.

Invocation: **A>DUMP {filename}**

Where {**filename**} is the complete file name of the disk file that you wish to examine in a hexadecimal display.

DUP.COM

The DUP utility can be used to duplicate ALL of the data from one disk to another disk. It can also compare the two disks to verify whether the data recorded on one disk correspond exactly to the data recorded on another disk. If desired, DUP will even perform both operations consecutively, to ensure accurate duplication of a disk.

Invocation: **A>DUP or A>DUP {dest}:={source}:{[options]}**

Where **{dest}** is the drive letter containing the blank disk to receive the data

Where **{source}** is the drive letter containing the data disk to be duplicated

Where **{[options]}** represents one or more of the following:

C - Copy only

V - Verify only (either disk may be source or destination)

N - No inquiry prompt

Without the C or V options, DUP will copy all of the data from the source disk to the destination disk, then automatically compare the two disks to verify the data.

Example: A>DUP B:=C:[C,N]

ED.COM

The ED utility allows you to compose, alter, and manipulate files containing ASCII characters. The files composed by ED are often referenced or manipulated by other system utilities and commands. When invoking ED, you either create a new file or summon an old file into the computer. Files being EDited can be moved, altered, or displayed when you enter various ED commands.

Invocation: **A>ED {filename}**

Where **{filename}** is the complete name of a file that you wish to compose or edit.

Example: A>ED B:THISFILE.TXT

ERA

The ERA resident command frees the space once occupied by disk files, permitting the storage of new files. ERA will erase a single file or a group of files through the use of wild cards, ? (for single letters) or * (for the file name or extension).

Invocation: **A>ERA {filename.ext}**

Where **{filename.ext}** is the complete name of the file you wish to delete.

Examples: A>ERA B:DATA??.*

FORMAT.COM

The FORMAT utility prepares a floppy disk for storing data by establishing storage areas on the disk surface. At the same time, FORMAT erases any data that remains on the disk from prior use, and sometimes inspects the recording surface for imperfections that could impair data storage or transmission. FORMAT also enables you to determine how much data you will be able to store on the disk.

Invocation: **A>FORMAT or A>FORMAT {drive}:{[options]}**

Where **{drive}** is the drive letter that contains the disk you wish to format.

Note: This drive does not necessarily have to be a physical drive. For instance, if you have only one physical 5.25" drive, then you can still specify drive B. You will later be prompted to put the appropriate disk in the drive.

Where **{[options]}** may be one or more of the following:

SD 8-inch disk formatted to Single Density

DD 8-inch disk formatted to Double Density

1S 5.25-inch disk formatted on only one side

2S 5.25-inch disk formatted on both sides

F Fast formatting, because the disk surface is not tested

N No prompt displayed between FORMAT command entry and FORMAT execution

Examples:

A>FORMAT A:[2S,N]
A>FORMAT B:

Format disk in drive A on both sides, without prompts
Format disk in drive B

Note: All 5.25" disks are automatically formatted at double density. All 8" disks are automatically formatted on the number of sides for which the disk has been certified by the manufacturer. (FORMAT detects this certification by checking the position of the small hole in the disk cover next to the center spindle hole.)

LIST.COM

The LIST utility enables you to obtain paper copies of files by entering a command for one or more files to be printed. Special printout characteristics can be set when you enter the command with LIST parameters. You can stop a LIST printout in progress by pressing ANY keyboard character while the printout is in progress. Only files containing ASCII characters should be LISTed.

Invocation: **A>LIST or A>LIST {argument}**

Where {argument} is the name of the file(s) to be LISTed.

Example: A>LIST PRINTOUT.DOC B:PROGRAM.PRN B:REPORT.DOC

You can specify parameters in a LIST argument to alter the characteristics of a standard printout. These parameters allow selecting date, number of copies, the width of tabs, etc:

LIST PARAMETERS

| <u>Parameter Name:</u> | <u>Keyboard Entry:</u> | <u>Description of Characteristic:</u> | <u>Default Value:</u> |
|------------------------|------------------------|--|------------------------------|
| Date | [D xxx...x] | First 10 chars of specified date printed on upper right corner of each page | no date |
| Heading | [H xxx...x] | First 60 chars of specified heading printed at left of top line on each page | file name (1st copy only) |
| No Heading | [N] | No heading or date printed | filename |
| Lines/Page | [L nn] | Each page has the specified number of lines. [L 0] causes printing w/o page breaks | 60 lines/page |
| Tab Stop | [T nn] | Each tab stop within text is expanded or contracted to specified number of spaces | 8 spaces |
| Page Number | [P nn] | Page numbering sequence begins with specified number on first file page | page 1 |
| Upper Case | [U] | All letters printed upper case | upper & lower |
| Copies Desired | [C nn] | Specified number of copies are printed | 1 copy |
| Erase | [E] | File is erased from disk after LIST operation is completed | file retained on disk |

LOAD.COM

The LOAD utility puts an assembled hexadecimal file into memory and translates the file into a "COM" file, which is executable under the CP/M operating system. To be LOAded, a file must contain valid Intel hexadecimal format records and have a "HEX" extension. The ASM utility can be used to produce such a file from a file with the "ASM" extension.

Invocation: **A>LOAD {hex file}**

Where **{hex file}** is the primary file name of an Intel hexadecimal file. The file extension is omitted from this entry because LOAD always assumes the extension "HEX".

Example: A>LOAD B:BETA

MVCPM2x7.COM

The MVCPM2x7 utility will adjust the CP/M system kernel so that it has the proper memory size for your purposes, and so that it recognizes the type of disk (floppy or Winchester partition) it will be recorded on. It can change the system's size within a range of 48 through 64 kilobytes of Random Access Memory (RAM). This utility should be followed immediately with another utility or command, such as SYSGEN or SAVE.

Note: The two MVCPM2x7 utilities are provided in separate files because each one is intended for use with a specific disk controller card. MVCPM207 is intended for the Z-207 floppy disk drive controller card, and MVCPM217 is intended for the Z-217-1 Winchester hard drive controller card.

Invocation: **A>MVCPM{xxx} {nn} {d}:{biosfile.ext}**

Where **{xxx}** is the disk controller model number that controls the disk that will eventually receive the system being moved. If xxx is 207, the system will be prepared for a floppy disk. If xxx is 217, the system will be prepared for a Winchester partition.

Where **{nn}** represents the memory size that the transferred system kernel will occupy, in multiples of 1024 bytes (kilobytes). If an asterisk, "*", character or no value is entered, the system kernel will be set to occupy the entire memory capacity of the computer, by default. This value can be less than or equal to the computer's actual memory capacity.

Where **{d}** represents the drive letter containing the BIOS files that are to be matched up with the system kernel being moved. This variable is optional. If omitted, MVCPM207 will assume that the created system kernel should be modified to be compatible with the BIOS files that are currently active in computer memory.

Where **{biosfile.ext}** represents the name of the file containing the system components normally stored in the file BIOS85.SYS. This variable is optional. If omitted, MVCPM207 will assume the file name "BIOS85.SYS".

Example: A>MVCPM207 48 C:

Notes:

- * The "*" character must be entered when you specify NO value for the memory {nn} variable, and DO specify a value for the drive name {d} and/or BIOS filename {biosfile.ext}. The "*" character is required as a "placeholder" so the drive name and/or file name parameters are not interpreted as a memory value.
- * The MVCPM207 utility should be immediately followed by either:
 - The SYSGEN utility, which will transfer the adjusted CP/M system kernel to the system tracks of a specified disk; or by
 - The SAVE resident command, which will transfer the adjusted CP/M system kernel to a file on a specified disk.

PIP.COM

PIP stands for Peripheral Interchange Program, the CP/M utility that can be invoked to copy and link files or parts of files, and to direct input and output between logical devices (see the following table). PIP also allows files to be transferred between different disks when using a one-drive hardware environment.

Invocation: **A>PIP or A>PIP {data dest}={data source}**

Where **{data dest}** can be a file, a disk, or a logical device.

Where **{data source}** can be a file or logical device.

Examples:

```
A>PIP B:NEW.DOC=C:SEND.DOC      Copy & Rename a file
A>PIP B:=C:*. *                Copy all files from drive C to drive B
A>PIP CON:=B:SENDOVER.DOC      Display SENDOVER.DOC on the screen
A>PIP LST:=SENDOVER.DOC        Print a copy of SENDOVER.DOC
A>PIP CREATED.DOC=CON:         Create a file from keyboard text
```

The following table matches physical and logical device names:

| Logical Device | Physical Device | Description of Recommended Input/Output Machine: |
|-----------------|------------------------------|--|
| CON: Console | TTY: CRT: BAT: UC1: | A printing terminal attached to serial port outlet A A video display terminal & keyboard A batch pseudo-device using RDR: for input & LST: for output A modem attached to serial port B |
| RDR: Reader | TTY: PTR: UR1: UR2: | A printing terminal attached to serial port outlet A Not implemented A modem attached to serial port B A video display terminal & keyboard |
| PUN: Punch | TTY: PTP: UP1: UP2: | A serial printer attached to serial port outlet A Not implemented A modem attached to serial port B A video display terminal & keyboard |
| LST: List | TTY: LPT: CRT: UL1: | A serial printer attached to serial port outlet A A parallel printer attached to the parallel port A video display terminal & keyboard A modem attached to serial port B |

All devices in the preceding table can also be referenced using the STAT utility, which assigns physical devices to logical devices on a temporary basis. Use CONFIGUR to match the devices on a permanent basis.

The PIP utility can also be used to transfer the BIOS files, as shown:

A>PIP {dest}:={source}:{filename} [RV]

Where {dest} is the destination disk and {source} is the source disk
 Where {filename} is the system file(s) to copy
 Where [RV] are the PIP parameters used to help you copy a file that has system status, and to help you verify the accuracy of the copy operation.

The PIP utility can also gain access to five additional devices, defined below.

- NUL:** Sends 40 "nulls" (ASCII zeros) to the destination device. The NUL: device is usually accessed at the beginning and/or end of the output when a paper tape punch device is used.
- EOF:** Sends an "end-of-file" character (ASCII CTRL-Z) to the destination device. A CTRL-Z is sent automatically when PIPping a file composed of ASCII chars.
- INP:** Special input source which can be "patched" into the PIP program itself. Through this source, PIP accepts data input char-by-char using a system call to memory location 103H. The data returns from location 109H. The parity bit must be preset at zero.
- OUT:** Special output destination which can be "patched" into the PIP program. PIP transmits data from register C to this destination using a system call to memory location 106H. Note that locations 109H through 1FFH of the PIP memory image are not used, and can be replaced by special purpose drivers using DDT.
- PRN:** This device is accessed for the same purposes as the LPT: device. In its operation, however, tabs are expanded at every eighth character position, lines are numbered, and page breaks occur every 60 lines. Output directed to the LPT: device will be treated identically if the [TNP] or [T8NP60] parameter is used in a LIST command.

When PIPping to the CON: logical device, the copy operation can be suspended by pressing {CTRL}-{S} and resumed by entering any character other than {CTRL}-{C}. The operation can be aborted by entering any character except {CTRL}-{S} while data is being transferred.

PREL.COM

The PREL utility is invoked to take the hex output files of two assemblies and generates a relocatable file from them. The first of the assemblies should have an initial origin of 000h and the second assembly should be one page higher, or 0100h. The hex output of the first assembly must have a filename extension of "HX0", and the other must have "HX1" for its extension. PREL can be implemented as one of the batched commands in a SUB file.

Invocation: **A>PREL {hex filename} {relocatable filename}**

Where **{hex filename}** is the primary name of the two assembled hex files, which have the extensions HX0 and HX1

Where **{relocatable filename}** is the primary name of the output page relocatable file, which will have the extension PRE

Example: A>PREL MYPROG TESTBIOS

In which TESTBIOS.PRE is formed from the two hex files, MYPROG.HX0 and MYPROG.HX1.

REN

The REN command allows you to assign a new name to any disk file. If it cannot RENAME a file, it displays the reason.

Invocation: **A>REN {newname}={oldname}**

Where **{newname}** is the name you wish to assign to the file

Where **{oldname}** is the name of the file before it is RENamed.

Example: A>REN C:MYFILE.DAT=YOURFILE.DAT

SAVE

The SAVE command is used to copy the contents of a span of memory locations to a disk file. The amount of data that SAVE will copy is measured in pages. A page of memory in CP/M is 256 (decimal) bytes, or a span of 0100h (hexadecimal) memory locations. The data that SAVE copies from the computer's memory space can come from a variety of sources.

Invocation: **A>SAVE {pages} {filename}**

Where **{pages}** is the decimal number of pages of memory contents that is SAVED

Where **{filename}** is the name of the file created to store the data

Example: A>SAVE 3 B:PROGRAM.ASM

One useful application is to SAVE a program that has been placed in memory by the DDT utility. To produce the system prompt (A>) and allow entry of the SAVE command, you must exit from the DDT utility with a warm boot (performed by pressing **{CTRL}-{C}**).

Another common application is to SAVE a copy of the operating system onto a file. Part of the system always occupies the area of memory between locations 0h and 100h. However, if the system kernel is loaded into memory by the MVCPM207 utility, it can be SAVED to a file. To do this, you must follow a MVCPM207 command with a command in the following form:

A>SAVE 38 CPMkk.COM

Where **kk** is the number of kilobytes of memory to which the system has been adjusted by the MVCPM207 utility.

STAT.COM

The STAT utility has two functions: to provide statistical information about disk space or file size, and to allow you to change the assignment of various types of status to files and devices.

Invocation: A>STAT

Will list all drives that have been logged into and respond with a display like the following:

```
A: R/W, Space: 3k
B: R/O, Space: 18k
```

Where "A:" and "B:" indicate the drive letters

Where "R/W," stands for Read/Write, meaning that data and files can be written to or read from.

Where "R/O," stands for Read/Only, meaning that the disk is write protected.

Where "3k" or "18k" indicates the amount of free memory on the disk. Memory space is expressed in units of 1024 bytes, or kilobytes (k). One byte holds one character.

Invocation: A>STAT x:

Will list the status of the requested drive, x:, whether logged to or not.

Invocation: A>STAT x:{filename}

Will list the status of an individual file {filename} on the requested drive, x:, providing information of the following form:

```
Recs   Bytes   Ext   Acc
 140    18k     2   R/W x:{filename}
Bytes Remaining on x: nnnk
```

Where "Recs" stands for the number of records within the file. A record is 128 bytes (one-eighth of a kilobyte) of memory space. Thus, one record contains 128 data characters. Any number of characters less than 128 characters is still reported as a record.

Where "Bytes" stands for the amount of memory space the file occupies, expressed in kilobytes (1024 bytes). One kilobyte of space will hold 8 records. Again, any number of records less than 8 is still reported as one kilobyte.

Where "Ext" stands for the number of extents that the file occupies. One extent will process 16 kilobytes of data. However, if a file occupies less than 16 kilobytes, STAT will still report that it requires one extent to be processed.

Where "Acc" indicates the file's access status - "R/W" or "R/O"

Invocation: A>STAT x:*.*

Will report the following information of the disk in drive x:

```
Recs   Bytes   Ext   Acc
 175    22k     2   R/W x:{filename1}
 160    20k     2   R/W x:{filename2}
 15     2k     1   R/O x:{filename3}
Bytes Remaining On x: nnk
```

Wildcards are recognized, as in the following examples:

```
A>STAT B:*.COM           Lists all .COM files on drive B:
A>STAT B:DATE????.DOC   Lists all .DOC files starting with DATE
A>STAT B:REPORT?.*      Lists all files starting with REPORT
```

The STAT utility also performs functions which allow disk protection, manipulation of file indicators, disk parameter statistics, user number display, and I/O device assignment. A list of these special STAT functions can be invoked by entering the following command:

Invocation: A>STAT VAL:

In response, STAT will display the following list:

```
Temp R/O Disk: d:=R/O
Set Indicator: d:filename.typ $R/O $R/W $SYS $DIR
Disk Status : DSK: d:DSK:
User Status :USR:
Iobyte Assign:
```

CON: = TTY: CRT: BAT: UC1:
RDR: = TTY: PTR: UR1: UR2:
PUN: = TTY: PTR: UP1: UP2:
LST: = TTY: CRT: LPT: UL1:

STAT can be used to temporarily change a disk's access mode:

Invocation: **A>STAT x:={r/w}**

Where **x:** is the desired drive letter

Where **{r/w}** is the disk access mode desired:

R/W allows reading and writing from the disk

R/O allows reading only from all the files on the disk

Example: A>STAT B:=R/O

STAT can be used to change a file's access mode:

Invocation: **A:STAT x:{filename} {r/w}**

Where **x:** is the drive letter containing the desired file

Where **{filename}** is the desired full file name, including extension.

Wild cards "?" and "*" are also recognized for multiple files.

Where **{r/w}** is the file access mode desired:

\$R/W allows reading and writing of the file

\$R/O allows reading only from the file

\$SYS changes the file to the status of a system file

\$DIR removes the system attribute

STAT can list a disk's characteristics, with a display similar to the following:

```
  x: Drive characteristics
2496: 128 Byte Record Capacity
  312: Kilobyte Drive Capacity
  256: 32 Byte Directory Entries
  256: Checked Directory Entries
  128: Records/ Extent
   16: Records/ Block
   32: Sectors/ Track
   2: Reserved Tracks
```

Invocation: **A:STAT x:DSK:**

Where **x:** is the desired disk drive.

If no drive is specified in the command line, then a listing in the above form will be displayed for the disk in each drive that has been logged since the last warm boot.

User access to certain files can be controlled by assigning user areas to the files. (See text on the USER resident command.) The following form of the STAT utility will display a list of the numbered user areas which have files on the currently logged disk.

Invocation: **A>STAT USR:**

Will produce the display similar to the following:

```
  Active User : 0
  Active Files: 0 1 3
```

The first line lists the currently logged user area, which was set at zero either by the last USER command, or a cold boot. The second line lists the other user area numbers that have been established for files on the disk. This list is displayed as "Active Files", and indicates that additional files can be accessed by logging user area one or three.

Finally, STAT can temporarily match logical and physical devices. It enables you to monitor and temporarily control the assignment of physical Input/Output devices to the appropriate logical device names. (Use CONFIGUR to permanently assign the physical device names to the logical device names.)

Invocation: **A>STAT DEV:**

Lists the current assignments of the physical devices to logical devices, similar to:

```
CON: is CRT:
RDR: is UR1:
PUN: is UP1:
LST: is UL1:
```

The current logical to physical device assignment can be changed by entering this STAT command.

Invocation: **A>STAT {logical device}:= {physical device}:**

Where both the {logical device} and the {physical device} are specified by their four-character names, listed in the section describing the PIP command.

Example: A>STAT LST:=LPT:

SUBMIT

The SUBMIT utility enables you to initiate automatic execution of several sequential commands by issuing only one SUBMIT command. The SUBMIT command accesses a file containing a pre-composed sequence of commands. You can also vary the execution of this command sequence with each invocation, by substituting command line parameters such as file names, drive names, or device names for variables inserted into the command file.

A SUB file is a file with the "SUB" extension, which is comprised of a sequence of command lines composed using a text editor, such as ED, described earlier. The SUB file is the source of the command sequence referenced by the SUBMIT command. The command lines within a SUB file can contain prototype parameters to represent command line fields. These prototype parameters appear in the SUB file in the form, **\$n**, where each numeral "n" preceded by a dollar sign ("\$\$") takes the place of a file name, drive name, or device name which you will substitute into the SUB file with an actual parameter entered in the SUBMIT command line. The number of parameters specified in a SUBMIT command line must correspond to the number of different prototype parameters that appear within the SUB file in the form "**\$n**".

The following example shows the text of the SUB file with the name "MULTIJOB.SUB", which contains two prototype parameters:

```
ASM $1
DIR $1
ERA *.BAK
PIP $2:=$1.PRN
ERA $1.PRN
```

The SUBMIT utility can then be used as in the following example:

```
A>SUBMIT MULTIJOB PROGRAMX LST
```

Where {**MULTIJOB**} is the name of the SUB file filled with batch commands, and {**PROGRAMX**} and {**LST**} are parameters which will be substituted into one or more locations in the batch file.

After the entry of a SUBMIT command line, SUBMIT will create a file named \$\$\$SUB, and send it to the Console Command Processor (CCP) portion of the operating system. The CCP recognizes the commands in the SUB file and executes them in sequence. The \$\$\$SUB file contains the actual parameters expressed in the SUBMIT command substituted for the corresponding prototype parameters which are embedded in the original SUB file.

Execution of SUBMITTED commands can be aborted by typing a {**DELETE**}, {**RETURN**}, or {**CTRL**}-**{C}** immediately after the command is displayed.

SYSGEN.COM

The SYSGEN utility is used to transfer the operating system to a disk. Under some circumstances, SYSGEN does this task without the aid of other utilities. Sometimes SYSGEN needs the help of other utilities to put an entire, usable system on the disk.

If you have just run the MVCPM2x7 utility to customize the operating system for memory capacity, then a system kernel still resides in a special location of computer memory. You can copy this system kernel from the computer to the disk by simply pressing {RETURN} at the prompt:

SOURCE DRIVE NAME (OR RETURN TO SKIP).

However, if you are copying the CP/M system from memory to a destination disk (as you would after using MVCPM207), SYSGEN will not be able to copy the BIOS files for you. Since the BIOS files (BIOS85.SYS and BIOS88.SYS) are essential to make a disk bootable, you must copy them using the PIP utility, as shown:

A>PIP {dest} :={source} :*.SYS [RV]

Where {dest} and {source} are the respective drive letters and [RV] are PIP parameters used to help you copy a file that has system status, and to verify the accuracy of the copy operation.

Invocation: A>SYSGEN

TYPE

The TYPE command displays the contents of specified files on the console. Fast scrolling displays can be controlled to make them readable by pressing {CTRL}-{S} to suspend the scrolling text. Pressing any character will continue the scrolling. Pressing any character, other than {CTRL}-{S} will cause the scrolling to terminate.

Invocation: A>TYPE {filename}

Where {filename} is the complete, explicit name of a file. You cannot specify wildcard file names in a TYPE command.

To obtain a listing of the file, enter a {CTRL}-{P} before entering the TYPE command line. Listings will not exhibit special features (boldfaced type, underlining, etc.) that are inserted into some text files by word processing systems. Enter {CTRL}-{P} a second time to end the listing.

USER

The USER command enables you to access only files in a specified area of the disk directory, useful when several users have files stored on the same disk. File directory (DIR) and status checks (STAT) invoked by a particular user will list only the files in the user's specified area.

You can have and log into as many as 16 USER areas by invoking a single USER command line. Whenever a cold boot is performed, the user is automatically logged in to USER area number 0. When in this area, only files in USER area 0 of the directory are accessible. To make a user area other than number 0 accessible, you should respond to the system prompt by entering a command in this form:

Invocation: A>USER {area number}

Where {area number}, from 0 to 15, is the area that contains files to access.

The USER command can be issued whenever the system prompt appears. The currently-logged USER area will be in effect for all of the disks in the hardware environment, until a different USER area is logged or a cold boot is performed.

Files can be pulled into a USER area that contains the file PIP.COM, but to put the file PIP.COM into that USER area in the first place requires a special process.

XSUB

The XSUB utility extends the power of the SUBMIT utility to enable you to invoke utility programs, and then execute commands within these programs, while making only one entry at the terminal.

The XSUB command line is entered as the first line in a SUB file. A SUB file is a text file composed of command lines that are executed in sequence when a SUBMIT command line is entered at the system prompt.

Note: XSUB will not support the submission of input for programs which, when run, prompt for single character input. The only programs that can effectively be used in a SUB file under XSUB are programs that accept input lines ended with a carriage return, such as DDT, PIP, LIST, and ED.

```
Example:      XSUB
              DDT
              I$1.HEX
              R
              GO
              SAVE 4 $2.COM
```

Processing of this SUB file could be initiated by entering the following command line:

```
A>SUBMIT MEGAJOB PRGRMY PRGRMZ
```

The preceding entry creates a \$\$\$SUB file in which the primary file name "PRGRMY" is substituted for the prototype name "\$1" of the SUB file, and "PRGRMZ" is substituted for the "\$2" prototype parameter.

As commands are read from the \$\$\$SUB file, the XSUB utility enters computer memory. Then DDT enters and executes the DDT commands "IPRGRMY.HEX", "R", and "GO", which are also included in the \$\$\$SUB file. The DDT command "GO" has the same effect as a warm boot, which enables the operating system to process the final command in the \$\$\$SUB file, "SAVE 4 PRGRMZ.COM".