



Z-100 DISKPACK User's Manual

DiskPack Floppy Disk Device Driver

For the Heath/Zenith Z-100 Series Computer

- Original documentation (C) 1989 by William E. Flanagan
- Marketed by Paul F. Herman Inc.
- Updated February 2021 by Steven W. Vagts, Editor

Introduction

DiskPack was a software package which allowed a person to use many popular floppy disk formats on the Heath/Zenith Z-100 series computer, including PC compatible formats. Aside from the floppy drives and cables, no additional special hardware was required - all formats supported by DiskPack were provided using the standard Z-207 floppy disk controller already installed in your Z-100.

The DiskPack driver supported all popular 8", 5" (actually 5.25"), and 3" (actually 3.5") floppy disk formats. This included the standard Z-100 formats, as well as IBM AT compatible 1.2 Mb (megabyte) 5" drives, and 720 Kb (kilobyte) 3" formats. Additional formats were also available, some of which provided even greater capabilities.

The standard Z-100 floppy disk controller allows up to four standard density drives and four high density drives, for a total of up to 8 floppy drives on each controller, and two controllers may be used in the Z-100. Therefore, the DiskPack driver could support up to 16 floppy drives (including the 4 owned by the BIOS) of various types and capacities in your Z-100 system.

System Requirements

In order to use the DiskPack driver and other programs, you must have a Heath/Zenith Z-100 series computer with at least one standard Z-207 floppy disk controller card. The driver requires ZDOS version 2 (Zenith's version of MS-DOS for the Z-100) or above. One or more floppy disk drives and the appropriate cables are needed.

Note: The following describes operation of DiskPack using ZDOS versions 2 or 3. Under the newer ZDOS version 4 and ZROM v4, any combination of drives may be used without regard to the 48 tpi / 96 tpi DIP switch setting on the Z-207 Floppy Controller or type of drive, as the new DRIVECFG utility

configures the system drives and modifies the IO.SYS directly. DiskPack could still provide the capability of handling less popular disk formats, however.

Why are we talking about DiskPack, developed in 1989, now?

Good question. But, with the arrival of the new Floppy Disk Emulator, suddenly DiskPack becomes very relevant for some users again.

In the last issue, #133, of the "Z-100 LifeLine", I introduced the Floppy Disk Emulator. The Floppy Disk Emulator uses a USB flash drive as the storage media to replace up to 1000 1.44 Mb 3.5" floppy disks and is useful for transferring files from a PC. There are two popular models available at the present time. The first and earliest model featured a two-digit 7-segment display on the front panel which displays 00 to 99 for a total of 100 floppies that could be recognized. It also featured a Green LED for power and a Red LED for access, and two push buttons - one to increment the floppy being accessed, another to decrement - and a USB port for memory stick storage.

The later model featured a three-digit 7-segment display on the front panel which displays 000 to 999 for a total of 1000 1.44 Mb floppies that could be recognized. It also featured a Green LED for access, and two push buttons - which increment the floppy unit being accessed - and a USB port for memory stick storage.

Almost any USB flash drive seems to work, but it should be 256Mb or larger for using the maximum capacity of the first model and 2GB for the later model. If the memory stick is smaller, the number of 1.44 Mb floppy partitions is reduced.

There is an auto-format function that automatically formats the USB flash drive into the maximum number of 1.44 Mb partitions. However, you may format the partitions you wish to use and read/write to them like you would when using the computer with real floppy disks.

The Gotek emulator generally comes with Firmware that ONLY supports the 3.5", 1.44 Mb MFM format. This means that this emulator is suitable for IBM/PC (and compatibles) using high density 3.5" floppy emulation only. However, there are Firmware updates that now permit the emulator to emulate up to 39 different disk drive sizes and formats.

Use in the Z-100 Computer:

The Gotek Floppy Emulator sounds like just the ticket for those who desire fast, solid-state booting capability from nonvolatile RAM without the expense of an IDE Controller Board. This sounds like it would work just as well as an IDE hard drive, and with 1000 1.44 Mb partitions, there would still be plenty of storage. The only disadvantage is the limited storage of one floppy, when compared to the much larger IDE hard drive partitions. But who really needs several megabyte partitions when using the Z-100? Another advantage to the USB drive is the capability of transferring files between a Z-100 and a PC.

The only thing really missing when compared to an IDE Controller Board would be the real-time clock capability. But with my new, inexpensive ZCLK2 clock board, this could easily be remedied.

I have created an updated Gotek User's Manual and published it with LifeLine issue #133. It is available on the *Z-100 LifeLine* website.

So, let us get back to why the DiskPack Driver becomes relevant again.

Pre-ZDOS v4, the Z-100 could only use standard density, 48 tpi, 5" drives **OR** 96 tpi, 5" drives on the 34-pin connector of the Z-207 Floppy Controller Card, depending upon the setting of section 0 of the DS1 DIP switch on the controller; set to 0 (On) for 48 tpi, and to 1 (Off) for 96 tpi operation.

Furthermore, **high density** media must use the 500K rate of the 50-pin interface (8" media). Therefore, **DiskPack would enable use of the new Floppy Emulator models on a standard, unmodified Z-100!**

The Gotek Floppy Disk Emulator firmware upgrade (there are two similar firmware upgrades available) promises several improvements over the standard 1.44 Mb 3" disk emulator. It supports:

- Hundreds of computers, keyboards/samplers/synthesizers and milling machines!
- 3", 5", and 8" disks.
- Single and double sided disks.
- FM / SD, MFM / DD, or MFM / HD densities.
- 125Kb/s, 250Kb/s, 300Kb/s, 500Kb/s bit rates.
- 300 RPM, 360 RPM, and 600 RPM.
- 128/256/512/1024/2048/4096 byte sector sizes.
- All track skew and sector interleaving.
- Up to 255 tracks per disk.
- And more!

So, while the standard ZDOS only supported the normal 5" and 8" drives, the DiskPack driver can control, the other sized drives and formats that you may need with minimal expense.

However, there are some limitations to DiskPack:

- Difficulty getting the cabling correct:
 - Requires a special cable to change drive unit to 2 or 3 because the BIOS uses units 0 and 1 and most drives do not have DS2 or DS3.
 - The 50-pin Z-207 interface requires a Ready Signal, not available on many newer drives.
- Can NOT Boot to any drives not owned by the BIOS.
- Can NOT be used with any operating system other than ZDOS.

So, let us take a closer look at DiskPack. It has never been reviewed before and it is overdue.

DISKPACK Floppy Disk Device Driver

Distribution Disk Contents

The DiskPack distribution disk contains the following files:

DSKPAK.SYS	Floppy disk device driver
DPCNFG.EXE	Device driver configuration utility
DPFMT.COM	Floppy disk formatting utility
DPINFO.COM	Floppy disk format information utility
READ.ME	If, present, will contain additional info

Be sure to make a working copy of the DiskPack distribution disk. Then put the original away in a safe place.

The DPCNFG Configuration Program

The DiskPack device driver must be configured before you attempt to use it. Configuration is accomplished by using the DPCNFG.EXE program, included with DiskPack. This program builds the required table information for the disk drives that you will be using on your system. DPCNFG.EXE must be run before you first install the DiskPack driver, and any time you wish to change your disk drive configuration. The information provided is stored in the DSKPAK.SYS system file, which is loaded during the system boot-up process.

The DPCNFG.EXE program is invoked with the command:
DPCNFG [*drivename*]{RETURN}

where;

[*drivename*] = An optional argument that gives the complete pathname of the device driver file to be configured.

If [*drivename*] is absent from the command line, the DPCNFG program will assume that the driver is named DSKPAK.SYS and it is found in the default directory. In the simplest case, the DSKPAK.SYS driver from the DiskPack distribution disk would be in the same directory as the DPCNFG .EXE program. Then the configuration program may be invoked simply by typing:

DPCNFG{RETURN}

The '**Transfer Rate**' refers to the data transfer rate between the floppy disk controller and the disk drive. The Z-100 controller supports two transfer rates; 250K and 500K. A different connector is provided on the Z-207 controller board for each transfer rate. The 34-pin connector is for drives which use the 250K transfer rate. The 50-pin connector is for drives which use the 500K transfer rate.

The Heath/Zenith literature generally refers to the 34-pin connector as the 5" floppy interface, and the 50-pin connector as the 8" floppy interface. However, the DiskPack driver allows you to use alternate drive types. The important thing is that the drive must be connected to the connector which provides the appropriate transfer rate, regardless of the type of connector which is provided on the drive itself.

The '**Drive Select Code**' shown in the Disk Drive Address Screen is a letter assignment used by the configuration program to select a particular controller location and address. The letters shown (from A thru P) are menu options used to select a particular position for the drive you are configuring. THE DRIVE SELECT CODE LETTERS DO NOT HAVE ANYTHING TO DO WITH THE DISK DRIVE LETTERS USED TO REFERENCE A DRIVE UNDER DOS.

The '**Drive Select Address**' is programmed on the drive by using jumpers, a DIP switch, or a jumper block. Most drives allow select addresses of 0, 1, 2, or 3, although some drives may refer to the drive select address as 1, 2, 3, and 4. The drive select address which is programmed on the drive must correspond with the drive select address configured in the driver.

The '**Drive Owner**' shows whether a particular drive is controlled (or owned) by the BIOS, or by the DiskPack DSKPAK.SYS driver. Present versions of the BIOS assume drives 0 and 1 of the 250K interface of the primary controller are 5" floppies (drives A: and B:) and drives 0 and 1 of the 500K interface of the primary controller are 8" drives (drives C: and D:) all owned by the BIOS. Ownership of these drives cannot be changed without making modifications to the Z-100 BIOS. A summary of the drive owner abbreviations is shown near the bottom of the Disk Drive Address Screen.

If you have two controllers in your system, it is important to note that the BIOS-owned drives will be on the controller used for boot-up. This means that the BIOS-owned drives can be on the Primary or the Secondary controller. The Disk Drive Address Screen shows the BIOS-owned drives on the primary controller since this would be the normal default configuration. Normally, re-configuring the BIOS-owned drives would have no affect, since the DiskPack driver does not interface with the BIOS drives. However, if you will be booting from the secondary controller, then the drives shown as BIOS-owned in the Disk Drive Address Screen will actually end up being owned by the DiskPack driver, and should, therefore, be configured properly for the type of drives you are using.

If you have only one controller, you can still configure a drive for the secondary controller. This drive will be considered as a phantom (or imaginary) drive, and must have a similar type drive controlled by the DiskPack driver on the primary controller. Then whenever you make reference to the phantom drive, the DiskPack driver will request you to insert the disk for the phantom drive into the real drive on the primary controller.

The '**Drive Status**' simply shows whether a drive has been configured for each controller address. A 'C' indicates that a drive specification has been configured for the address specified by that column.

Configuration Program Menus

The main menu of the DPCNFG configuration program allows you to select the following options:

A - Add Drive Definition to Driver

This option is used to add a new drive specification to the driver. You will be led through a series of menus which enable you to define the drive type, controller, transfer rate, hardware select address, and step rate.

B - Copy a Drive Definition to Another Drive

A convenience option which allows you to copy an existing drive definition to another controller drive position. This makes it easier to configure multiple disk drives of the same type.

C - Display Drive Configuration Information

This option allows you to display the configuration for a single disk drive, or for the entire system.

D - Move a Drive Definition

A convenience option which allows you to move a drive definition to another controller or drive select address. The drive definition at the original controller position is removed and shown as available.

E - Remove a Drive Definition from the Driver

Use this option to delete a drive definition from the driver configuration. This may be necessary if you make a mistake during configuration, or if you remove a drive from your system.

F - Update a Drive Definition in the Driver

Changes an existing drive definition. The drive must have already been defined using the Add, Move, or Copy options listed above, or be a default BIOS drive. If you are using 96 tpi BIOS-owned drives, you should change the default from 48 tpi to 96 tpi, by setting section 0 of the DS1 DIP switch (set to 0 (On) for 48 tpi, and to 1 (Off) for 96 tpi operation) on the controller so the driver knows the drive type.

G - Set Preferred Sequence for Drives

This option may be used to specify the sequence of the drive letters to be used with the DiskPack driver-owned drives.

H - End Configuration Process

Select this option when you are done configuring the DiskPack driver, or when you want to abort the program. You will be given an opportunity to create a backup copy of the original (unmodified) driver before quitting.

Installing the Driver

To install the DiskPack floppy disk device driver, several steps are required.

- Copy the DSKPAK.SYS device driver file to a working disk, along with the DPCNFG.EXE configuration program.
- Run the DPCNFG.EXE program to configure the driver for your particular system.
- Copy the modified DSKPAK.SYS driver file to the disk and directory where you want it installed. Typically, this would be the Root directory or your boot disk, but you may also place it in a subdirectory of your boot disk.
- Edit or create a file named CONFIG.SYS and add a line in the file like this:

DEVICE = DSKPAK.SYS

This tells MS-DOS to load the DiskPack DSKPAK.SYS driver during the boot-up process. If the driver is not located in the Root directory of the boot disk, then you must include the full path name of the driver file. In other words, if the DSKPAK.SYS file is located in the DOS subdirectory of the Root directory, then the line in your CONFIG.SYS file must be as follows:

DEVICE = DOS\DSKPAK.SYS

- Now you are ready to use the driver (assuming that the disk drives have been installed).

The driver will be loaded the next time you boot up. It will not be necessary to keep the DPCNFG program on your system disk, since it is only needed to change the driver configuration. However, you may want to copy the DPFMT and DPINFO programs to your working disks.

The DPFMT Format Program

The DPFMT program is a disk format program to be used with all drives controlled (or owned) by the DiskPack DSKPAK.SYS driver. DPFMT may also be used to format the BIOS-owned drives on your system. This program prepares a disk to be used by DOS by initializing the disk media and preparing the various tables used by DOS for file storage.

Note: The DPFMT program may ONLY be used to format floppy disks, and will not format hard disks.

Note: DO NOT use the DPFMT program on a network disk drive, or a drive that has been Mapped, Assigned, or SubstEd (see your MS-DOS manual for more information on these commands).

The DPFMT program is invoked with the following command:

DPFMT d: [/switches ...]

where:

d: = A drive letter designating which disk drive is to be formatted. If the drive letter is omitted, the DPFMT program will display a sign-on screen and then return you to the DOS prompt.

[/switches ...] = Optional command line switches.

More than one switch may be used (as long as they are not contradictory). Here is a list of the valid command line switches:

/1 Format disk with single-sided format. This switch is valid for all drives or media that support single side formats.

Eight inch drives use single side and single density media. For 96 tpi standard density drives owned by the BIOS, this switch is invalid. No 3" or 5" HD formats are single sided.

/8 Formats a disk with 8 sectors/track on all standard density drives. The default is 9 sectors/track.

This switch is invalid for all high density type disks, and formats not specifically supporting 8 sectors/track.

/B[=d:] Boot table request control.

This switch is provided for two purposes.

- For 8" single sided, single density media, it informs the format program to create a disk with 4 reserved sectors instead of one.
- On other disk types, this switch may be used to copy a boot sector from the specified '=d:' drive, to the new disk.

If you are creating a bootable system disk for another system, you should use the /S switch. But if you simply want to create a data disk which may be made bootable at some later time, use this /B switch.

- /F=y** Formats the disk with a specific format. The 'y' parameter is used to identify the specific formats, as follows:
- E** Use DiskPack extended format with 1024 byte sectors. This format is invalid for real 8" drives, or BIOS-owned drives connected to the 50-pin connector.
 - U** UCI format for 5" 96 tpi drives.
 - Z** This format causes 96 tpi drives to use the Zenith format instead of the MS-DOS format. This switch may also be used to format AT type drives with the standard 8" format.

If the BIOS owns the drive, the formats specified by this switch are the default.

Note: All DiskPack driver-owned drives will default to standard DOS formats. All BIOS-owned drives will always use Zenith Data Systems' formats.

- /L[=volumename]** Causes the DPFMT program to prompt for a volume label after it has completed formatting the disk. If the optional '=volumename' argument is provided, 'volumename' will be used as the volume label without further prompting.

The volume label may be up to 11 characters. Embedded blanks may be used if the entire volume label is enclosed in single or double quotes. Legal characters for a volume label are the same as those which may be used for a file name. See the MS-DOS manual for a list of illegal characters.

- /N** Causes the DPFMT program to suppress all screen prompts. Only a sign-on message will be displayed. This may be useful when you invoke DPFMT from a batch file.
- /S[=d:]** Causes the system files (including the boot sector) to be transferred to the newly formatted disk, in effect making it a bootable disk.

Note: Only formats known to the BIOS are bootable on the Z-100.

However, the [=d:] option, if included, specifies the drive from which the system files (and boot sector) are to be copied, making it possible to make a disk which is bootable on another type of computer. Use this switch to make an IBM System Disk. If [=d:] is omitted, the system files will be copied from the default drive.

- /V** Causes the DPFMT program to verify the disk after it is formatted.

If no command line format switches are used, the default format for the drive type will be used. The default format is shown in **Appendix C: DiskPack Driver Format Tables**.

If the DPFMT program is invoked with only a question mark as a command line argument, a help screen will be displayed.

The DPINFO Disk Format Information Program

The DPINFO program included with DiskPack is a utility that tells you the present format of a disk in any floppy drive. The program is invoked with:

DPINFO [d:] [/switch]

where:

- [d:]** = The drive holding the disk you want to check. If the drive letter is not specified, the default drive will be used.
- [/switch]** = An optional command line switch specifying what type of information to display. The valid command line switches are:
 - /F** Display detailed media format information. This switch provides all the statistics about the format of the media in the specified drive.
 - /M** Display a map of a floppy disk's format. This switch shows the organization of the sectors on the disk.
 - /L** Display the logical map of a hard disk partition. This switch shows the same information as switch /M, but is used for hard drives. Sector numbers are referenced to the start of the partition.

/P Display a physical map of a hard disk drive and the specified partition. Same display as /L switch, but sector numbers are numbered beginning at the start of the physical drive.

If no switches are included, DPINFO will simply display the media size. If DPINFO is invoked with only a question mark for a command line argument, a help screen will be displayed.

This program is provided to help you in dealing with the variety of disk drives. If you have a system with 5" 48 tpi and 96 tpi drives, it is very easy to get the disks mixed up. DPINFO should help.

Note: Disks formatted with one drive may not be readable in another type drive. A good example of this is when a high density disk is read in a standard density drive. This will return an error just as though the disk were not formatted. So care should be taken to keep high and standard density disks plainly marked. One tip is to put a dot of white-out on the corner of ALL high density disks. This identifies the disk as high density, and it does not come off easily.

You may want to use DPINFO to determine a disk's format before copying files to another disk. Or if you want to make an exact copy of a disk, you may use DPINFO to determine the source disk format. Running DPINFO will tell you in a matter of seconds. But be sure the disk is not an unreadable format before proceeding.

Floppy Disk Drive Types Supported

This section gives a description of the drive types supported by DiskPack, along with some tips and suggestions on how to effectively use the DiskPack device driver.

8" Drives

Eight inch drives that are double sided and capable of both single (FM) and double (MFM) density are supported by the BIOS, and unless you want to support more than two, you would be just as well to allow the BIOS to support these drives. Support for more than two 8" drives, or more than one floppy controller, however, will require the DiskPack driver. All 8" drives must be connected to the 50-pin drive interface (500K transfer rate) on the Z-207 Floppy Disk Controller.

5" (5.25") Drives

Five inch drives can be 48 tpi (tracks per inch) **OR** 96 tpi, single or double sided, and standard or high density. There is no real advantage to using single sided drives, as the cost savings between one and two heads is nominal, and the main reason for adding drives to a system is to gain additional storage capacity. However, if you already own a single sided drive, you may use it with the DiskPack driver.

Single sided drives will work with both the BIOS and the DiskPack driver. However, the BIOS assumes that all disks are double sided. Therefore, if you use a single sided drive with the BIOS and you insert a disk that is formatted as double sided, when the BIOS tries to change sides, it will generate an error reading. Therefore, it would be better to use the DiskPack driver, instead of the BIOS, with single sided drives. In the driver's configuration process, if you indicate that the drive is single sided (one head), the driver will only allow single sided formats.

If you want to use a **mix of 48 tpi and 96 tpi drives** on the same system, you may want to allow the DiskPack driver to own the 96 tpi drives, since the driver supports more 96 tpi formats than the BIOS. In any case, since the BIOS reads a DIP switch on the controller board to indicate 48 tpi or 96 tpi operation, the BIOS must own either all 48 tpi or all 96 tpi drives.

Note: Section 0 of the DIP switch, DS1, on the Z-207 Floppy Controller must be set to 0 for 48 tpi operation or to 1 for 96 tpi operation.

The DiskPack driver supports all standard 48 tpi formats, and two extended 48 tpi formats, as well. This gives you the standard 160Kb and 180Kb single sided, and 320Kb and 360Kb double sided formats. The extended 48 tpi formats provide 200Kb on a single sided drive and 400Kb on double sided drives.

On 96 tpi disks, the DiskPack driver supports all the 48 tpi formats as the BIOS does. It also supports the standard MS-DOS 96 tpi formats. The Zenith Data Systems (ZDS) 96 tpi formats are NOT standard MS-DOS formats. This is not because Zenith wanted to be different, but because ZDS supported 96 tpi formats before Microsoft did. The standard DOS formats supported for 96 tpi disks are the same formats supported for 3" media. The main advantage to allowing the DiskPack driver to own the 96 tpi drives, instead of the BIOS, is that you will gain compatibility with other MS-DOS systems that support 96 tpi disks. The DiskPack driver also offers an extended format for 96 tpi drives which gives 400Kb for single sided and 800Kb for double sided media.

In addition to the standard 48 and 96 tpi drives, the DiskPack driver also supports high density 5" disk drives - when they are connected via a special cable to the 50-pin interface on the controller board. These drives are the 1.2Mb drives normally used on IBM-AT systems. When you use these drives, you may use either the standard MS-DOS AT format (80 tracks), or the Zenith standard 8" format (77 tracks). The Zenith 8" format is available with drives owned by the BIOS too, however BIOS owned drives will lose the compatibility with IBM-AT systems.

The DiskPack driver also supports two extended capacity formats. The extended high capacity driver formats give 1.44Mb with 512 byte sectors (format similar to 1.44Mb 3" media) and 1.6Mb 1024 byte sectors. However, neither of these

extended formats is compatible with any MS-DOS or IBM-PC/AT standard.

If you choose to use these high capacity disk drives, you should remember that these drives use the 8" (500K transfer rate) interface on the Z-207 controller board. This means that a special cable is required to connect the 34-pin connector on the disk drive to the 50-pin interface on the controller. A wiring diagram for this special cable can be found in the **Appendix D: Hardware Installation**.

3" (3.5") Drives

The DiskPack driver supports 135 tpi (96 tpi equivalent) 3" drives. The formats supported with these drives are the standard MS-DOS formats, which include either 8 or 9 sectors per track, 80 tracks, and single or double sided.

High density 3" drives (PS/2 compatible) are also supported by the DiskPack driver. These drives offer formatted capacities up to 1.44Mb using the IBM standard format.

The DiskPack driver also offers extended formats which are the same as those for 5" 96 tpi media.

These drives will work with the BIOS most of the time, but the BIOS will NOT provide the PC compatible formats. Another problem that arises when using 3" drives owned by the BIOS is that the system may hang if a drive is accessed when no disk is in the drive. This is not a fault of the BIOS or the hardware; it is caused by a functional difference between the 3" and 5" disk drives.

BIOS Versus DiskPack Driver

Note: The following describes operation of DiskPack using ZDOS versions 2 or 3. Under the newer ZDOS version 4 and the newer ZROM v4, any combination of drives may be used without regard to 48 tpi or 96 tpi DIP switch setting or type of drive, as the new DRIVECFG utility configures the system drives and modifies IO.SYS directly. DiskPack would still provide the capability of handling less popular disk formats, however.

The DiskPack driver does not attempt to redefine disks owned by the BIOS. The driver has disks it exclusively owns, and the BIOS has disks it exclusively controls. The decision as to who owns the disk drive is up to the user and how the drives are installed and addressed.

The BIOS will ALWAYS own the drives at drive select 0 and 1 of both the 250K and 500K interfaces on the controller used for booting. The 250K interface has a 34-pin connector, and the 500K interface has a 50-pin connector. These were originally assumed to support 5" and 8" drives, respectively. Therefore, the normal BIOS configuration is two 5" drives (Drives A: and B:) and two 8" drives (drives C: and D:).

Important Note: The BIOS ALWAYS owns drive select 0 and 1 of the 34-pin interface and drive select 0 and 1 of the 50-pin interface on the controller used for booting.

The MTR-100 monitor ROM (often referred to as the ZROM, and located at position U149 on the motherboard) will support more than two 5" or 8" floppy drives, as well as more than one controller card, even if the BIOS will not. If you use the default boot sequence, or boot from a hard drive, the BIOS will own the drives (normally A:, B:, C:, and D:) on the primary controller. But if you have two controllers, you may boot from the secondary controller, in which case the BIOS will own these same drives on the secondary controller.

The BIOS does NOT support disk drives other than drive select 0 or 1. Therefore, if you attempt to boot from a drive with a drive select address of 2 or 3, it will cause the system to hang. The ZROM will actually load the boot code from the requested drive and will execute it, but when control is turned over to the BIOS, everything will crash.

The Z-207 Floppy Controller board uses a DIP switch to inform the BIOS whether the 5" drives are 48 tpi or 96 tpi. If you want a mix of 48 and 96 tpi drives on the same system, put all the 48 tpi or all the 96 tpi drives under control of the BIOS, and set the DIP switch appropriately. If you wish the ability to have the BIOS use 48 tpi sometimes, and 96 tpi drives other times, then you will need to install two controller boards, one configured for 48 tpi operation and the other for 96 tpi operation.

Appendix A: A Primer on Floppy Disk Technology

Floppy disk drives come in varying sizes, the most common of which are 3" (3.5"), 5" (5.25"), and 8". The 8" drive was the first to appear, followed by the 5" varieties, then the 3" and even 2" for laptops, before all became obsolete. However, all versions of the floppy are still available (mostly used) and since we are still using mid-80's computers, we should understand the differences between these common drive types of the period and have some knowledge of the hardware.

Read/Write Heads

One easily understood component of different disk drives is the number of read/write heads the drive uses. The earliest drives only had one read/write head and only recorded information on one side of the media. By adding a second head and using the other side of the media, you double the capacity of the drive. In the 8" drive, there was another difference between double sided and single sided drives; the index hole in the media was in a different location, based on whether the media was single or double sided.

Recording Mode

Another difference in capacity is in the way data is recorded on the media, or the way it is encoded. Early drives were recorded with an encoding technique called FM (Frequency Modulation).

Later, a method called MFM (Modified Frequency Modulation) came along. These two recoding techniques are equivalent to the terms single density (FM) and double density (MFM).

The change from single density again doubled the storage capacity of the drive, but it also added a level of confusion. A drive that was single sided and capable of recording in double density had the same capacity as a double sided single density drive.

Track Density

The next area of improvement, and the third factor to consider, was track density. Each disk is recorded in a series of concentric circles (tracks), with each adjacent circle getting closer to the center of the disk. The earlier drives could record at a track density of 48 tpi (tracks per inch). At this track density, good track definition and separation could be maintained.

As innovations came along, stepper motors improved and the accuracy of the read/write heads improved. As a result, drives were introduced with a 96 tpi density, and therefore doubled the drive's storage capacity again. This introduced an additional confusion factor when talking about disk capacity. A single sided, 96 tpi disk had the same capacity as a double sided, 48 tpi disk, when both are in the same recording mode.

Transfer Rate

Yet another consideration in disk performance is in the transfer rate. Drives recorded as double density (MFM) have a greater transfer rate than drives recorded as single density (FM). The Z-207 Floppy Disk Controller used a transfer rate of 500K for the 8" drives, and 250K for the standard density 48 tpi or 96 tpi 5" drive for default formats. In high capacity drives (like the high density 5", 96 tpi, AT format drive), the transfer rates are the same as for the 8" drives.

Track Capacity and RPM

Due to improvements in read/write heads and disk media, higher densities have also been achieved by packing data closer together on each track. This enhancement gave birth to the 96 tpi, 5" IBM-AT type disk drive. These drives typically operate at a speed of 360 rpm, just like an 8" disk drive (versus the normal 300 rpm for a 48 tpi, or standard density 96 tpi 5" drive).

These high density drives record at the same track density as the 8" drives, but in a smaller track circumference. Some high capacity disk drives literally double the track density and still operate at 300 rpm.

For example, an 8" drive using MFM recording mode, will average about 10,400 bytes per track (depending on which track), yielding an unformatted capacity of 1.6Mb for the entire disk. The AT type disk drive uses the same capacity as the 8" drive, and also has an average of 10,400 bytes per track. The drives that are double the normal capacity of normal 96 tpi, 5" drives, have an average track density of 12,500 bytes per track, giving an unformatted capacity of 2.0Mb per disk.

Media Size

Because of improvements in stepper motors, read/write heads, and media, it was possible to introduce smaller 3" drives. Most manufacturers introduced these drives with capacities comparable to their 5" cousins. However, this further confused the issue because a different track density had to be used to accommodate the smaller media. The two formats established for the 3" disk are 67 tpi and 135 tpi. The 67 tpi is equivalent to the 48 tpi density of 5" disks, while the 135 tpi density is equivalent to 96 tpi 5" disks. Of these two 3" densities, only the 135 tpi is commonly used.

Although we have track densities of 48, 67, 96, and 135 tracks per inch, the actual recording formats are limited to 40, 77, or 80 tracks per disk. The 77 track format is only used for 8" disks. A 135 tpi 3" disk has a storage capacity equal to a 96 tpi 5" disk, if all other factors (number of heads, recording mode) are the same. The reason is because they both have 80 tracks; the tracks on the 3" drive are just closer together.

Super High Capacity

There are some super high capacity floppy disk drives that were available. These drives may have unformatted capacities as high as 6Mb. Some of them relied on secondary 'fine-tuning' steppers to maintain alignment, or other tricks which have not enjoyed widespread acceptance. Other floppy drives were developed with capacities over 20Mb, relying on special optical or laser sensors for tracking. These disk drives used special preformatted media, and had more characteristics in common with removable hard disk drives than floppy drives. None of these super high capacity drives were supported by DiskPack.

Appendix B: DOS Disk Format Information

A floppy disk or hard disk must be formatted before use by DOS. The DOS FORMAT program builds required information tables on the disk so that files can be located. For floppy disks, the FORMAT program also writes address information onto each data track of the disk so that sectors can be located and verified. For hard disks, the FORMAT program may only initialize the table information, or verify data sectors for the purpose of flagging bad sectors so DOS will not try to use them. The actual data track address formatting for hard disks is usually done by a 'low level' format program (called PREP by Heath/Zenith).

Boot Sector

The first of the table areas written by the FORMAT program is a small program called the boot loader, which is used to load DOS into the system at the time the system is booted. The boot loader code is written into the boot sector, which is always the first sector on the disk. There is also a table which explains certain media information to the monitor ROM (ZROM) boot code, and the boot loader code on disk. This table will have the sector size, number of sectors per allocation unit, number of reserved sectors, etc.

The format of the boot loader table, from Zenith Data System's (ZDS) viewpoint, comes in two varieties. One is the IBM version for IBM compatibles, and the other is the Z-100 type.

One of the main purposes of the boot loader table is to inform the boot code about the format of the media. The data contained in the boot loader table is pertinent to the disk format and does not relate to the disk drive's capabilities. Some additional information is defined by ZDS which relates to the boot process and not the format, therefore it was not included in the following table descriptions.

IBM Format for the Boot Loader Table

Offset	Length	Description
0	8 bytes	OEM name
8	2 bytes	Bytes per sector
10	1 byte	Sectors per allocation unit
11	2 bytes	Number of reserved sectors
13	1 byte	Number of FAT tables
14	2 bytes	Maximum Root directory entries
16	2 bytes	Number of sectors per media
18	1 byte	Media descriptor
19	2 bytes	Sectors per FAT
21	2 bytes	Sectors per track
23	2 bytes	Number of heads
25	2 bytes	Number of hidden sectors

ZDS Format for the Z-100 Boot Loader Table

Offset	Length	Description
0	1 byte	Version number
1	2 bytes	Bytes per sector
3	1 byte	Sectors per allocation unit
4	2 bytes	Number of reserved sectors
6	1 byte	Number of FAT tables
7	2 bytes	Maximum Root directory entries
9	2 bytes	Number of sectors per media
11	1 byte	Sector size shift count (log2 of size)
12	1 byte	Sectors per track
13	2 bytes	First data sector number
15	1 byte	Cluster factor shift count (log2 of factor)
16	2 bytes	First directory sector

File Allocation Table (FAT)

The next table created by the FORMAT program is the file allocation table. This table maps the status of each data cluster available on the disk. A cluster is a minimum grouping of sectors which can be manipulated as a unit. The FAT will map the location of all the clusters which are linked together to make up a data file. Bad sectors are also indicated by the FAT, so they will not be used by DOS. This table is written to the disk twice in the standard DOS format. Only one table is actually required. The other table is presumably a backup copy which is maintained in case the first table becomes corrupt. However, it is questionable if DOS ever actually consults this second table in the event of a corrupt original FAT table, but nevertheless it remains a DOS requirement to have both copies.

Directory Table

The last table built by the FORMAT program is the Root directory table, which is used to identify all the files and subdirectories in the Root directory of the disk. The directory area holds information about the file length, attributes, and the starting cluster for each file or subdirectory. DOS can then consult the FAT table to determine the location of the rest of the file. Note that all subdirectories under DOS are merely data files with a special attribute set. No space is reserved on the disk for directories, except for the Root directory.

Data Space

The remainder of the disk space is considered as data space. This data space is divided into clusters, each of which is pointed to by a particular FAT entry. In other words, the first FAT entry corresponds with the first data cluster, the second FAT entry with the second data cluster, and so forth.

Before DOS writes a file to disk, it finds the first available data cluster by consulting the FAT. Data from the file is then written into the cluster, and the FAT is updated to show the cluster as being used for this file. Then another available data cluster is found, and the process repeats until the entire file is saved on disk. As new FAT entries are reserved for the file, the previous entries are changed to point to the present entry so as to form a chain of pointers from one cluster to another. After the file is saved, the last cluster number is indicated in the FAT so that DOS will know where to stop when the file is read.

Later versions of MS-DOS (including our ZDOSv4) handle the allocation of data clusters a little differently. They search for a contiguous string of available clusters large enough to hold the file, and then put the file in the 'hole'. Only if a large enough hole CAN NOT be found, does it then revert back to the original method of cluster allocation. This newer scheme has the advantage of storing files as contiguous pieces, which can be read faster.

Appendix C: DiskPack Driver Format Tables

The tables in this section show everything needed about the specific formats which are available with the DiskPack driver. The formats shown are those that might commonly be created, read, or written using the DiskPack driver. The DPFMT utility switches used to format a disk with the desired format are shown for each table entry.

Note: DiskPack can actually read or write other formats, as long as the boot loader table data is valid (see **Appendix B: DOS Disk Format Information**).

Note: The default format shown for each type of disk drive is the default for DiskPack driver-owned drives. BIOS-owned drives will always default to the ZDS standard format for that drive type, if listed.

Appendix C - DiskPack Driver Format Tables

Important Note: The default format shown for each type of disk drive is the default for driver-owned drives. BIOS owned drives will always default to the ZDS standard format for that drive, if listed.

Disk Formats for 3.5", 135 Tpi Disk Drives, Standard Density

Sec. Size	Sec. trk.	Trk. Side	No. Head	Sec. Cls.	Sec. FAT	Dir. Ent.	FAT ID	Media Size	Std.	Sup.	Format Switches
512	8	80	1	2	1	112	FA	320KB	DOS	Yes	/1/8
512	8	80	2	2	2	112	FB	640KB	DOS	Yes	/8
512	9	80	1	2	2	112	F8	360KB	DOS	Yes	/1
512	9	80	2	2	3	112	F9	720KB	IBM	Yes	default
1024	5	80	1	1	1	128	F8	400KB	DP	Yes	/1/F=E
1024	5	80	2	1	2	160	F8	800KB	DP	Yes	/F=E
512	9	80	2	1	5	224	F9	720KB	#1	Yes	
512	9	80	2	1	5	112	F9	720KB	#2	Yes	

Note: All formats use double density MFM recording mode.
All formats use 2 FAT tables.
All formats have 1 Reserved Sector.

1: This format is used by IBM for distribution of DOS 3.3
2: This format is used by IBM for distribution of DOS 4.0

Disk Formats for 3.5", 135 Tpi Disk Drives, High Density

Sec. Size	Sec. trk.	Trk. Side	No. Head	Sec. Cls.	Sec. FAT	Dir. Ent.	FAT ID	Media Size	Std.	Sup.	Format Switches
512	18	80	2	1	9	224	F0	1440KB	IBM	Yes	default
1024	10	80	2	1	3	256	FB	1600KB	DP	Yes	/F=E

Note: All formats use double density MFM recording mode.
All formats use 2 FAT tables.
All formats have 1 Reserved Sector.

Disk Formats for 5 1/4" 48 Tpi Disk Drives

Sec. Size	Sec. trk.	Trk. Side	No. Head	Sec. Cls.	Sec. FAT	Dir. Ent.	FAT ID	Media Size	Std.	Sup.	Format Switches
512	8	40	1	1	1	64	FE	160KB	IBM	Yes	/1/8
512	8	40	2	2	1	112	FF	320KB	IBM	Yes	/8
512	9	40	1	1	1	64	FC	180KB	IBM	Yes	/1
512	9	40	2	2	2	112	FD	360KB	IBM	Yes	default
1024	5	40	1	1	1	96	F8	200KB	DP	Yes	/1/F=E
1024	5	40	2	1	1	160	F9	400KB	DP	Yes	/F=E
512	9	40	2	1	3	112	FD	360KB	#1	Yes	

Note: All formats use double density MFM recording mode.
All formats use 2 FAT tables.
All formats have 1 Reserved Sector.

1: This format is used by IBM for distribution of DOS 4.0.

Disk Formats for 5 1/4" 96 Tpi Disk Drives, Standard Density

Sec. Size	Sec. trk.	Trk. Side	No. Head	Sec. Cls.	Sec. FAT	Dir. Ent.	FAT ID	Media Size	Std.	Sup.	Format Switches
512	8	80	1	2	1	112	FA	320KB	DOS	Yes	/1/8
512	8	80	2	2	2	112	FB	640KB	DOS	Yes	/8
512	9	80	1	2	2	112	F8	360KB	DOS	Yes	/1
512	9	80	2	2	3	112	F9	720KB	DOS	Yes	default
512	8	80	2	4	1	144	FB	640KB	#1	Yes	/8/F=Z
512	9	80	2	4	2	144	F9	720KB	ZDS	Yes	/F=Z
1024	5	80	1	1	1	128	F8	400KB	DP	Yes	/1/F=E
1024	5	80	2	1	2	160	F8	800KB	DP	Yes	/F=E
512	9	80	2	2	5	112	FF	720KB	UCI	Yes	/F=U

Note: All formats use double density MFM recording mode.
All formats use 2 FAT tables, and 1 reserved sector.

1: Standard ZDS format. FAT ID byte is FD for Z-DOS.
All formats found in 5 1/4" 48 Tpi are available in read only mode.

Disk Formats for 5 1/4" 96 Tpi Disk Drives, High Density (10400)

Sec. Size	Sec. trk.	Trk. Side	No. Head	Sec. Cls.	Sec. FAT	Dir. Ent.	FAT ID	Media Size	Std.	Sup.	Format Switches
512	15	80	2	1	7	224	F9	1200KB	IBM	Yes	default
1024	8	77	2	1	2	192	FE	1232KB	#1	Yes	/F=Z
1024	8	80	2	1	2	192	F8	1280KB	DP	Yes	/F=E

Note: All formats use double density MFM recording mode.
All formats use 2 FAT tables.
All formats have 1 Reserved Sector.

1: This is the standard IBM 8" format used in the 5 1/4" drive. It is provided for owners of this type drive who have used them with the H/Z-100's BIOS. This is the default format for BIOS owned drives of this type.

Disk Formats for 5 1/4" 96 Tpi Disk Drives, High Density (12500)

Sec. Size	Sec. trk.	Trk. Side	No. Head	Sec. Cls.	Sec. FAT	Dir. Ent.	FAT ID	Media Size	Std.	Sup.	Format Switches
512	18	80	2	1	9	224	F0	1440KB	#1	Yes	default
1024	10	80	2	1	3	256	FB	1600KB	DP	Yes	/F=E

Note: All formats use double density MFM recording mode.
All formats use 2 FAT tables.
All formats have 1 Reserved Sector.

1: This is the standard IBM 3 1/2" format supported in this drive.

Disk Formats for 8" 48 Tpi Disk, Standard Density

Sec. Size	Sec. trk.	Trk. Side	No. Head	Sec. Cls.	Sec. FAT	Dir. Ent.	FAT ID	Media Size	Std.	Sup.	Format Switches
128	26	77	1	4	6	68	FE	250KB	IBM	Yes	/1
128	26	77	1	4	6	104	FD	250KB	ZDS	Yes	/1/B
1024	8	77	2	1	2	192	FE	1232KB	IBM	Yes	default

Note: All formats use 2 FAT tables.
All formats with FD for a FAT ID use 4 Reserved Sectors. Others have 1 Reserved Sector.
All 128 byte sector size formats use single density FM recording mode.

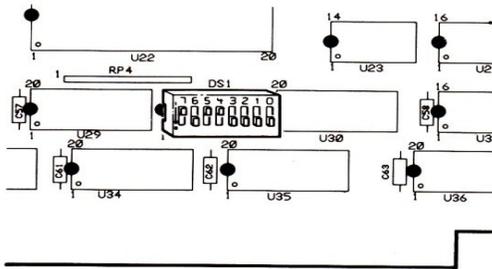


Appendix D: Hardware Installation

The Z-207 Floppy Disk Controller board has two interface connectors used to attach floppy disk drives. The 34-pin interface is used to connect disk drives with an MFM recording mode transfer rate of 250K. The 50-pin interface is used to connect disk drives with an MFM recording mode transfer rate of 500K. This controller will support up to four disks through each interface.

The Z-100 computer will further support two Z-207 Floppy Controllers, giving a maximum capacity of 16 floppy disk drives for the system. Each Z-207 Floppy Disk Controller has a DIP switch, DS1, located toward the bottom center of the board. This DIP switch is used to set the port address of the card, the Precomp Select for 5" drives, and, for 5" drives, sets the drives for operation as 48 tpi or 96 tpi. See the figure.

Section	Description
7	} Set port (modulo 8)
6	
5	
4	
3	
2	not used
1	Precomp select for 5-1/4" disk drives
0	0 = 48 tpi, 1 = 96 tpi (5-1/4" drives)



Setting the Z-207 Floppy Controller DIP Switch, DS1

DS1 Floppy Disk Controller Switch Settings

With the board upright (gold pins at bottom), DS1 is viewed with section 7 at the left end; switches down (0) are OFF, switches up (1) are ON.

Sections 7-3: Base Port Address of Z-207 Floppy Controller. These sections set the base port address of each Z-207 Floppy Disk Controller. The Z-100 series computers use port addresses B0 & B1 (hex) as shown.
 Pri. controller = Port B0 = 1011 0
 Sec. controller = Port B1 = 1011 1

To Set Controller Address:	First * (Primary)	Second (Secondary)
Section 7:	ON (1)	ON (1)
Section 6:	OFF (0)	OFF (0)
Section 5:	ON (1) = B0	ON (1) = B1
Section 4:	ON (1)	ON (1)
Section 3:	OFF (0)	ON (1)

Section 2: This section is NOT USED.

Section 1: This section selects precompensation for 5.25" disk drives. Position this section ON (1) if you want precompensation. Position this section OFF (0) if you do NOT want precompensation. *

Section 0: This section selects the 5.25" disk drive type. Position the switch section to ON (1) to use 96 tpi disk drives. Position the switch section to OFF (0) to use 48 tpi disk drives. *

Note: The asterisk denotes the normal position, and for using one controller.

Check the Terminating Resistor

Disk drives connected to the controller on the same interface are said to be daisy chained together. That is, the cable comes off the controller and connects to the first drive, then on to the next drive until all drives (up to four) are connected to the controller. The last physical drive in the chain must have a terminating resistor installed. All other drives must have the terminating resistor removed.

The termination method will vary between manufacturers. Some will use a terminating resistor pack, which will need to be removed from all but the last drive in the chain.

Generally, the 5" floppy drives will use a 16-pin DIP (dual in line) pack in a socket. Other drives may use an 8 or 9-pin SIP (single in line) pack in a socket. Others may use a jumper block or a DIP switch, and all jumpers or switch sections would need to be removed or turned off to remove the resistance.

Note: Newer 3" drives do not have terminating resistors, so if you experience any difficulty, try changing the drive's position on the cable; that is, move it to last or to the center of the cable.

Note: Some drives allow for radial connection as well as daisy-chaining. Be sure that the drives are set for daisy-chain connection.

Note: One last thing you need to consider is the location of the last drive, and its power source. This last drive will have the

terminating resistor and it needs to be ON in order for the string to be properly terminated. If all drives in the string derive their power from the same source, then there is no problem. However, if the drives are split between separate power sources (as would be the case if the last drive is in an external case), then the string will not be properly terminated when the last drive is turned off. This may cause intermittent I/O failures due to no terminator being found on the string.

Drive Select Address

Each disk drive must have a unique address, and all floppy disk drives have a method of setting the disk drive address. The drive address may be 0 to 3 or 1 to 4, depending on the drive manufacturer and whether they set the address relative to 0 or 1. Z-207 Controller documentation specifies drive addresses relative to 0.

The method used to set the drive select address will also vary by manufacturer, and their specification sheets may need to be consulted. Some use jumper blocks, while others may use an address block strapping chip or DIP switch.

Floppy Drive Cabling

The cabling required to connect drives to the controller is straight through ribbon cable if you are taking a 34-pin drive connector to the 34-pin interface, or a 50-pin drive connector to the 50-pin interface on the Z-207 floppy controller.

If you are connecting a drive with a 34-pin drive connector to the 50-pin interface on the Z-207 floppy controller, a modified cable will be required. Further, if the drive does not have the needed Drive Select numbers, Motor ON signal or Ready Signal jumper, than the addition of 1N34 diodes may be necessary to generate the required signals.

As we discussed earlier, because the BIOS-owned drives are using Drive Select 0 and 1, DiskPack driver-owned drives must use Drive Select (or unit numbers) 2 and 3. A quick, simple fix (for adding one drive) is to place a twist in the cable going to the DiskPack driver-owned drives by separating the flat cable lines 10 thru 14 from the cable end connector and flipping them so line 10 now goes to pin 14 of the connector, 11 to 13, 12 to 12, etc. As you can see from the first diagram, Drive Select 0 on the drive now becomes Drive Select 2. Unfortunately, Drive Select 1 remains Drive Select 1 on the Drive. But for adding one DiskPack drive, this would be an easy fix.

Finally, while the 34-pin interface does not use a Ready signal, the 50-pin interface of the Z-207 card requires one. This can be generated by installing 1N34 signal diodes, as we will see in the diagrams later.

First, let us assume a drive has Drive Select setting 2 or 3 for use with DiskPack, and a Ready Signal Jumper. The cabling configuration for a simple 34-pin drive connector to the 50-pin Z-207 interface becomes:

Simple 50-pin to 34-pin Adapter Cable

	Z-207 50-pin 500K Interface		34-pin Drive Connector	
Ground	1	-----	1	Ground
Two Sided	10	-- Ground		
Side 1 Select	14	----->	32	Side 1 Select
Head Load	18	----->	16	Motor ON
Index	20	<-----	8	Index/Sector
Drive Ready	22	<-----	34	Dsk Chg (drv)
Drive Select 0	26	----->	10	Drive Select 0
Drive Select 1	28	----->	12	Drive Select 1
Drive Select 2	30	----->	14	Drive Select 2
Drive Select 3	32	----->	6	Drive Select 3
Direction Select	34	----->	18	Direction Select
Step	36	----->	20	Step
C. Write Data	38	----->	22	C. Write Data
Write Gate	40	----->	24	Write Gate
Track 0	42	<-----	26	Track 0
Write Protected	44	<-----	28	Write Protect
C. Read Data	46	<-----	30	C. Read Data

Notes:

All odd number pins on both connections are assumed to be ground. Other pins not shown are no connection (NC). All lines active low. Loop a wire from pin 9 or pin 11 to pin 10 (to ground pin 10).

50-pin to 34-pin Adapter Cable With Motor ON Diodes

	Z-207 50-pin 500K Interface		34-pin Drive Connector	
Ground	1	-----	1	Ground
Two Sided	10	-- Ground		
Side 1 Select	14	----->	32	Side 1 Select
Index	20	<-----	8	Index/Sector
Drive Ready	22	<-----	34	Disk Chg (drv)
Head Load	18	----->	16	Motor ON
Drive Select 0	26	--x V		
Drive Select 1	28	--x V		
Drive Select 2	30	----- ---->	10	Drive Select 0
Drive Select 3	32	----->	12	Drive Select 1
Direction Select	34	----->	18	Direction Select
Step	36	----->	20	Step
C. Write Data	38	----->	22	C. Write Data
Write Gate	40	----->	24	Write Gate
Track 0	42	<-----	26	Track 0
Write Protected	44	<-----	28	Write Protected
C. Read Data	46	<-----	30	C. Read Data

Notes:

All odd number pins on both connections are assumed to be ground. Other pins not shown are no connection (NC). All lines active low. Loop a wire from pin 9 or pin 11 to pin 10 (to ground pin 10). If Head Load signal is unreliable, install 1N34 diodes (the V symbol) as shown **with the band towards the Drive Select signal**. The motor(s) should turn ON with either Drive Select signal.

50-pin to 34-pin Adapter Cable With Ready Signal Diodes

	Z-207 50-pin 500K Interface		34-pin Drive Connector	
Ground	1	-----	1	Ground
Two Sided	10	-- Ground		
Side 1 Select	14	----->	32	Side 1 Select
Index	20	<-----	8	Index/Sector
Head Load	18	----->	16	Motor ON
Drive Ready	22	<-----, x-	34	Disk Chg (drv)
Drive Select 0	26	--x V		
Drive Select 1	28	--x V		
Drive Select 2	30	----- ---->	10	Drive Select 0
Drive Select 3	32	-----,---->	12	Drive Select 1
Direction Select	34	----->	18	Direction Select
Step	36	----->	20	Step
C. Write Data	38	----->	22	C. Write Data
Write Gate	40	----->	24	Write Gate
Track 0	42	<-----	26	Track 0
Write Protected	44	<-----	28	Write Protected
C. Read Data	46	<-----	30	C. Read Data

Notes:

All odd number pins on both connections are assumed to be ground. Other pins not shown are no connection (NC). All lines active low. Loop a wire from pin 9 or pin 11 to pin 10 (to ground pin 10). If the Drive does NOT create a Ready signal (no jumper), install 1N34 diodes (the V symbol) as shown **with the band towards the Drive Select signal**. Each drive should be Ready with their Drive Select signal.

50-pin to 34-pin Adapter Cable for Dual 3"+5" Combo Drive With Ready Signal Diodes

	Z-207 50-pin 500K Interface		34-pin Drive Connector	
Ground	1	-----	1	Ground
Two Sided	10	-- Ground		
Side 1 Select	14	----->	32	Side 1 Select
Index	20	<-----	8	Index/Sector
Head Load	18	--x ,---->	16	Motor ON 1
Drive Ready	22	<-----, ,-->	10	Motor ON 0
		V V		
		V V		
Drive Select 2	30	----- ---->	12	Drive Select 1
Drive Select 3	32	-----,---->	14	Drive Select 0
Direction Select	34	----->	18	Direction Select
Step	36	----->	20	Step
C. Write Data	38	----->	22	C. Write Data
Write Gate	40	----->	24	Write Gate
Track 0	42	<-----	26	Track 0
Write Protected	44	<-----	28	Write Protected
Comp. Read Data	46	<-----	30	C. Read Data

Notes:

All odd number pins on both connections are assumed to be ground. Other pins not shown are no connection (NC). All lines active low. Loop a wire from pin 9 or pin 11 to pin 10 (to ground pin 10). Models: Teac FD-505 or Canon MD5511

If the Drive does NOT create a Ready signal (no jumper), install 1N34 diodes (the V symbol) as shown **with the band towards the Drive Select signal**. Each motor will turn ON with their Drive Select signal.

Appendix E: Example Floppy Drive Configurations

Drives:

- 2 5" standard density 48 tpi drives
- 1 5" high density AT type drive

Configuration:

- The BIOS should own the two standard density 5" drives, as drives A: and B:.
- DiskPack should own the high density 5" drive:
 - It should be connected to the 50-pin interface.
 - It should be programmed as Drive Select 2 or 3.
 - DiskPack will assign the drive letter I: to this drive.

Drives:

- 1 5" standard density 48 tpi drive
- 1 3" standard density drive

Configuration:

- The BIOS should own the standard density 5" drive, as drives A:.
- DiskPack should own the standard density 3" drive:
 - Both drives should be connected to the 34-pin interface.
 - The 3" drive should be programmed as Drive Select 2 or 3.
 - DiskPack will assign the drive letter I: to this drive.
 - The drive will provide a PC compatible 720Kb format (& others).

Drives:

- 1 5" standard density 48 tpi drive
- 1 5" standard density 96 tpi drive

Configuration:

- You may elect to let the BIOS own either of these drives.
 - Make sure section 0 of the DIP switch, DS1, on the Z-207 Controller is configured for either 48 OR 96 tpi drive operation.
 - This BIOS drive can be either Drive Select 0 or Drive Select 1 (A: or B:)
- DiskPack should own the other 5" drive:
 - It should be connected to the 34-pin interface.
 - It should be programmed as Drive Select 2 or 3.
 - DiskPack will assign the drive letter I: to this drive.

Drives:

- 1 5" standard density 48 tpi drive
- 1 5" high density 96 tpi drive
- 1 3" high density drive

Configuration:

- The BIOS should own the standard density 5" drive, as drives A:.
- DiskPack should own the high density drives:
 - They should be connected to the 50-pin interface.
 - They should be programmed as Drive Select 2 and 3.
 - DiskPack will assign the drive letters I: and J: to these drives.

Appendix F: Glossary of Terms

Allocation Unit: The smallest amount of sectors that can be allocated, also called a cluster. The number of sectors per allocation unit will always be a power of two (i.e., 1, 2, 4, 8, 16, 32, ...).

Block Device: A type of device designed to transfer data in blocks of more than one character. Examples of block devices include disk drives, tape drives, or laser disks. These devices are assigned drive letters by MS-DOS.

Boot Sector: A sector on the disk (always the first sector under DOS) which contains code and data necessary to load DOS into the system at boot-up time. The executable program code is generally referred to as a boot loader. The boot sector also contains a table which contains information about the disk media.

Cluster: See Allocation Unit.

Cylinder: A cylinder is composed of all tracks on a multi-head disk drive that can be accessed without moving the read/write heads. It may help to think of a hard disk with multiple disks; a cylinder would include both sides of each disk.

File Allocation Table (FAT): A table stored on disk which indicates the status of every cluster on the disk. The FAT shows which clusters are available, which are used by files, and which are bad (not usable). Clusters used by files are chained together in the FAT so that all clusters of a file can be located by DOS, if the starting cluster is known. The starting cluster is obtained from the directory entry for the file.

FAT ID: This is an identification byte found as the first byte in a FAT table. It is used to denote the physical makeup of the disk format. This ID byte, however, does not completely identify the disk format. Many disk formats use the same FAT ID.

Sector: A unit of disk space that consists usually of 128, 256, 512, or 1024 bytes of data (depending on the disk format). Sectors are arranged as segments of a disk track. In other words, a single disk track is comprised of a collection of sectors.

Track: One of many concentric rings around a hub of a disk media surface where data is stored. Each track is composed of one or more sectors. The tracks begin from the outside and higher tracks work toward the center of the disk.



Summary:

DiskPack was an early software package which allowed the use of all popular 8", 5" and 3" floppy disk formats on the standard H/Z-100, including PC compatible formats, such as the IBM AT compatible 1.2 Mb 5" drives, and 720 Kb 3" formats. Aside from the floppy drives and cables, no special hardware was required - all formats supported by DiskPack were provided using the standard Z-207 floppy controller.

Once ZDOSv4 was developed, which allowed the use of most popular drives in the Z-100 through the use of the DRIVECFG utility, the need for DiskPack virtually disappeared.

However, the development of a Floppy Disk Emulator, which uses a USB flash drive as the storage media to replace up to 1000 of the increasingly hard to locate 1.44 Mb 3.5" floppy disks, is intriguing for all of us. But while use of an emulator is easily accomplished under ZDOS4, users of earlier DOS versions would have a more difficult time. All of a sudden DiskPack becomes very relevant again, as an inexpensive option for these users.

I hope this article helps clarify the use of the DiskPack Floppy Disk Driver with this new Gotek Floppy Disk Emulator.

Cheers,

Steven W. Vagts
Editor, Z-100 LifeLine

