Porting Guide for the
HP BASIC Language Processor

HEWLETT PACKARD

Notice

Corvallis Information Systems
1000 N.E. Circle Blvd.
Corvallis, OR 97330, U.S.A.

Printing History

HP Computer Museum
www.hpmuseum.net

For research and education purposes only.

Contents

Appendix A: Using the CATCOPY Utility to Convert Files

Appendix B: HP BASIC Keyword Differences

Appendix C: Porting Programs That Use CSUBs

1

The HP BASIC Language Processor Release II Software

Will my existing programs run? What do I need to change to make my program work the way I want it to? These are the questions you probably ask first when you start using a new version of a programming language. Release II of the HP BASIC Language Processor software has been designed with two major goals in mind: *increased performance* and *backward compatibility*. Your existing programs should run with few, if any, modifications, whether written for a previous version of the language processor (Release I) or for an HP 9000 Series 200/300 computer running HP BASIC 5.0/5.1. This manual gives some *porting hints* that will help you modify your programs for Release II.

This manual does not tell you how to install or use the HP BASIC Language Processor software. For that information, refer to the manual *Installing and Using HP BASIC in the MS-DOS Environment*, which is also included with the software.

What Is Release II?

The HP BASIC Language Processor Release II software is a significant enhancement of the software provided with the HP BASIC Language Processor. Release II provides BASIC 5.13, the latest version of HP BASIC, but that is only the beginning. Release II provides greatly increased performance compared to Release I — especially for mass storage and graphics display operations — and provides better compatibility with HP 9000 Series 300 computers. At the same time, a very high degree of backward compatibility has been maintained with the previous versions of the HP BASIC Language Processor software (Release I). Most programs written for Release I should run without modification on Release II. However, you may want to make some minor modifications to your programs to make optimum use of the increased performance offered by Release II. Chapter 2 gives some tips on porting programs written for Release I to run optimally in the Release II environment. Chapter 3 covers porting from the Series 200/300 environment to the HP BASIC Language Processor Release II environment.

An Overview of Release II

The following is a summary of some of the main changes that have been made in Release II:

- New boot ROMs are supplied on all HP 82300C BASIC Language Processor boards. These boot ROMs greatly increase the boot speed and provide several boot options. Refer to chapter 3 in *Installing and Using HP BASIC in the MS-DOS Environment* for additional information. A set of these boot ROMs is included with the HP 82323A BASIC Language Processor Upgrade Kit.

- The CRT and graphics binaries have been modified to provide a bit-mapped display, like that of the HP 9000 Series 300 computers, on your PC monitor. Graphics performance is greatly increased, using the full area of the screen with no clipping of the graphics image. (Refer to chapter 6 in *Installing and Using HP BASIC in the MS-DOS Environment* for a description of the bit-mapped display.)

- A new mass storage binary — the DFS (DOS File System) binary — has been added to provide direct access from HP BASIC to the MS-DOS hierarchical file structure. This provides much faster disk access, and also provides file compatibility with MS-DOS applications. Note, however, that Release II provides complete compatibility with the LIF and HPW (virtual-LIF) file systems used in Release I. (Refer to chapter 7 in *Installing and Using HP BASIC in the MS-DOS Environment* for further information about mass storage.)

- Background mode has been modified to provide much greater capability when an HP BASIC program is running in background. The program in background can access DFS mass storage, and channels of communication can be set up between language processors in background and a foreground MS-DOS application. (Refer to chapter 4 in *Installing and Using HP BASIC in the MS-DOS Environment* for further information about background mode.)

- An MS-DOS device driver is installed that can control up to three HP BASIC Language Processors in your PC. (Refer to chapter 11 in *Installing and Using HP BASIC in the MS-DOS Environment* for information about multiple language processor operations.)

What Release Do I Have?

The software that this manual came with is the HP BASIC Language Processor Release II software. (The version number on the disk label is C.00.00 or higher.) The HP 82300C BASIC Language Processor includes the Release II software. You can update an HP 82300A/B BASIC Language Processor to Release II with the HP 82323A HP BASIC Language Processor Upgrade Kit.

There have been several previous versions of the language processor software, but for the purposes of this manual we will refer to all of them as "Release I". (The HP 82300A and HP 82300B BASIC Language Processors both included versions of the Release I software.)

The following table identifies the version numbers and product numbers associated with Release I and Release II of the language processor software:

	Product Number	Software Version Number	
		(Disk Label)	(Boot Message)*
Release I	HP 82300A	A.00.00	A.0
		A.00.01	A.0
Release I	HP 82300B	A.01.00	A.6
		A.02.00	A.02
		A.02.01	A.02.01
		A.02.02	A.02.02
Release II	HP 82300C	C.00.00 or higher	C.00.00 or higher
* The boot message appears in your display during the boot process.			

Note

The keyboard mapping of Release II is the same as that of the later versions of Release I. However, if you have been using versions A.00.00 or A.00.01 (HP 82300A) of the Release I software, you should note that the keyboard interface has changed, starting with version A.01.00. Refer to chapter 5 in *Installing and Using HP BASIC in the MS-DOS Environment* for information about the keyboard.

Differences from Release I

This section covers some Release II differences from Release I that have porting implications.

Hardware Requirements for Release II

You may install the HP BASIC Language Processor in any desk-top or desk-side Vectra PC. (All models of the Vectra CS, ES, QS, and RS are supported, but the Vectra LS is not.) You may also install the language processor in an IBM PC-AT computer and in many AT-compatible computers. The hardware requirements are given in detail in chapter 1 of *Installing and Using HP BASIC in the MS-DOS Environment*. However, the following are some differences that you should note if you are porting from Release I to Release II.

- *To run the Release II software, your PC must have at least 640 KB (kilobytes) of internal RAM (user RAM).*

- *Release II cannot be installed in a computer that has only one disk drive.* You need at least one flexible disk drive and one hard disk drive, or else two flexible disk drives, one of which must be a *high-density* (3.5-inch, 1.44 MB or 5.25-inch, 1.2 MB) drive. If you have only one disk drive in your computer, you will need to at least add a second flexible disk drive in order to install Release II.

- *CGA (Color Graphics Adapter) display systems are not supported by Release II. Also, the EGA (Enhanced Graphics Adapter) memory requirement is increased.* Release I required only 64 KB of EGA RAM, but Release II requires 256 KB of EGA RAM. All other display systems that were supported by Release I are supported by Release II.

- *Your HP BASIC Language Processor must be equipped with at least 1 MB of on-board RAM in order to run Release II.* (With Release I you could run a "stripped-down" system with only 0.5 MB of on-board RAM.)

- *Your language processor should be set to one of the following hardware configurations.* (Refer to chapter 1 in *Installing and Using HP BASIC in the MS-DOS Environment* for further information.)

  - Address 250h (hexadecimal), interrupt level IRQ7 (the default).
  - Address 280h, interrupt level IRQ9.
  - Address 330h, interrupt level IRQ5.

- *In order to obtain the increased boot speed of Release II, your language processor needs to have the new Release II boot ROMs.* All HP 82300C BASIC Language Processors have the new boot ROMs. Also, a set of Release II boot ROMs is included with the HP 82323A Language Processor Upgrade Kit. *You can run Release II with the old boot ROMs, but you won't get the increased boot speed.* However, once BASIC is running you will get Release II performance.

Other Differences from Release I

Here are some other differences that you should be aware of:

- *Release II does not support BASIC in ROM.* No HP BASIC ROM Board is available for Release II. Thus, Release II is always loaded and run in the language processor's on-board RAM. *If you have been using a ROM-based HP BASIC system, you most likely will need to add an additional 0.5 MB of RAM to your language processor.* The reason is that HP BASIC Release II will reside in RAM and will occupy space formerly available for your HP BASIC application programs.

- *The Series 200 display model is no longer supported.* Release II does not emulate Series 200 display hardware, but provides a bit-mapped, combined alpha/graphics display like that of Series 300. Clipping is no longer a problem, but you will need to modify your Release I programs to "undo" the clipping corrections that were recommended for Release I. Refer to chapter 2 in this manual for specific instructions.

- *Blinking characters are no longer supported.*

- *The Release II alpha display has 25 to 34 lines of alpha text, depending on the display used.* (Refer to chapter 2 for specific instructions.)

- *With Release II, each language processor running in background requires about 100 KB of PC RAM for an MS-DOS memory resident control program.* The memory resident program for background with Release I occupies only 8 KB of PC RAM, but provides much less functionality. This difference is significant if you are running a large MS-DOS application in foreground with one or more language processors in background.

- *The new DFS mass storage binary provides syntax differences that may require changes to your programs.* You will need to change mass storage specifiers in MASS STORAGE IS statements, and in other file specifiers, to make use of DFS. Refer to chapter 2 or 3 in this manual for further information.

- *The CONF.EXE utility has been modified.* The following changes have been made. (Refer to chapter 8 in *Installing and Using HP BASIC in the MS-DOS Environment* for detailed information.)

  - The name of the configuration file has been changed from "HPW.CON" to "BLP.CON".

  - The language processor address and interrupt level are no longer configured by CONF.EXE. (This is done by the driver.)

  - The "Machine Type", "VGA Enable", and "Background Mode" fields have been eliminated because the new display system doesn't need them.

  - The default BLP.CON file does not configure any internal disk drives for LIF or HPW use. You will need to run CONF.EXE if you want to access LIF or HPW volumes.

  - The default *BASIC* (68000) interrupt levels for PC serial ports and PC plug-in HP-IB cards have been changed. (However, the PC interrupt levels are the same as before.)

- The "DGCLEAR", "DRECTANGLE", and "DEL_GRAPH" CSUBs, provided with Release I, are not needed for Release II, and are not supported.

Differences from Series 200/300

The following comments are true for both Release I and Release II of the HP BASIC Language Processor software. They may affect you if you are porting a program from the Series 200/300 environment:

- CSUBs (Compiled Subprograms) that access the Series 200/300 hardware directly won't run with the language processor. Refer to appendix C in this manual for further information.

- Serial TRANSFERs are not supported. Use the ENTER and OUTPUT statements, not TRANSFER, with the PC serial ports COM1 and COM2.

- DMA (Direct Memory Access) is not supported by the language processor.

- The HP-HIL interface is supported only for HP Vectra PCs, not for non-HP computers.

- The HP 82335A and HP 82990A HP-IB interface cards are supported by the language processor through select codes 24 and 25. HP-IB (IEEE-488) cards from other manufacturers are not supported.

- HFS file access is supported only if the HP 82313A Hierarchical File System, available separately, is installed.

- The following Series 200/300 binaries are *not* supported by the language processor: "KNB2_0", "BCD", "BUBBLE", "EPROM", "FHPIB", and "HP9885".

Refer to appendix B for a summary of keyword differences from Series 200/300.

2

Porting from Release I to Release II

This chapter will help you port your programs written for Release I of the HP BASIC Language Processor to run optimally in the Release II environment. We say "optimally" because most programs written for Release I will run in the Release II environment with no changes. However, you may need to make some changes in order to make use of Release II's increased performance, and to obtain graphics with the proper aspect ratio.

> **Note**
>
> This manual assumes that you have installed Release II in your computer following the instructions in chapter 2 of *Installing and Using HP BASIC in the MS-DOS Environment*.

There are two primary areas of interest in porting a program from Release I to Release II: *mass storage* and the *display*. Let's look at mass storage first.

Mass Storage Issues

The HP BASIC Language Processor Release II software provides three mass storage *file systems*, which are described in detail in chapter 7 of *Installing and Using HP BASIC in the MS-DOS Environment*. These are the *LIF* and *HPW* file systems, which provide compatibility with the Series 200/300 and Release I environments, and the new DOS File System, or *DFS*, which provides increased access speed and direct access to the MS-DOS hierarchical file structure. *It is recommended that you use DFS file access for optimum performance and to allow direct sharing of files with MS-DOS applications.* (Release II also supports the HP 82313A Hierarchical File System, or *HFS*, which is available separately.)

When you write new programs, you can use DFS file access exclusively. Just follow the examples given in *Installing and Using HP BASIC in the MS-DOS Environment*. When you port a program from Release I, you need to do two things: 1) change the mass storage references and 2) copy existing files into a DFS directory. Let's look at the program modifications first.

Modifying Programs for DFS Mass Storage

The DFS mass storage file system is covered in detail in chapter 7 of *Installing and Using HP BASIC in the MS-DOS Environment*. In general, any statement in your Release I program that contains a MSVS (Mass Storage Volume Specifier) will need to be modified to access a DFS volume. A typical MSVS for an HPW volume would be ":,1500,0" (to access drive A:), or ":,1500,2" (to access drive C:).

Note
You can continue using HPW volumes with Release II, but you must *enable* the disk drives that you plan to access using the utility "CONF.EXE". (Refer to chapter 8 in *Installing and Using HP BASIC in the MS-DOS Environment* for further information.) *Note, however, that the new DFS mass storage system provides significantly faster mass storage access.*

You can access a DFS volume with an MSVS such as ":DOS,A" (to access drive A:) or ":DOS,C" to access drive C:. However, normally you will want to subdivide your hard disk into *directories* such as "C:\BLP" and "C:\BLP\DATA", for example. You can specify a full path in a mass storage specifier. For example, to specify the directory "C:\BLP\DATA" as the current mass storage, use the following statement.

```
MASS STORAGE IS "\BLP\DATA:DOS,C"
```

or

```
MSI "\BLP\DATA:DOS,C"
```

In addition to changing the mass storage specifiers, you may need to change the file name references in your program. DFS file names are limited to eight characters plus a "dot extension" of three characters. All alpha characters are "case-folded" into uppercase characters. Refer to "File Name Differences," later in this chapter, for further information.

An HPW Example. Let's port a short program segment. In the following program segment the MASS STORAGE IS statement declares ":,1500,2" (an HPW volume on drive C: in this case) to be the current mass storage:

```
340    ! Create file and store data from array "Trace_a"
350    !
360    MASS STORAGE IS ":,1500,2"
370    CREATE BDAT "DATAFILE01",13
380    !
390    ASSIGN @File TO "DATAFILE01"
400    !
410    OUTPUT @File;Trace_a(*)
420    ASSIGN @File TO *
430    !
440    END
```

This program segment creates a BDAT file named "DATAFILE01" in HPW volume ":,1500,2" and then outputs data from the array "Trace_a" to that file.

Let's modify the program segment so that it creates a file in a DFS directory. Let's further modify the program so that it creates a data file of the *DOS* file type, rather than BDAT: *

```
340    ! Create file and store data from array "Trace_a"
350    !
360    MASS STORAGE IS "\BLP\DATA:DOS,C"
370    CREATE "DATAFILE.01",1
380    !
390    ASSIGN @File TO "DATAFILE.01"
400    !
410    OUTPUT @File;Trace_a(*)
420    ASSIGN @File TO *
430    !
440    END
```

Note that line 370 creates a DOS type file with only one record. Don't worry, though. DOS files are *extensible* — they "grow" to the size needed. The file name has been modified to "DATAFILE.01", which is a legal file name for DFS.

You can create and use BDAT files in a DFS directory. However, *the DOS file type is preferred for DFS file access* for two reasons: *higher performance* and *better compatibility* with MS-DOS applications. A DOS-type file is really a standard MS-DOS file — it has no "header." Thus an MS-DOS

---

* The "CREATE" statement, without the "BDAT" or "ASCII" secondary keyword, creates an "HP-UX" file in a LIF or HPW volume, but creates a "DOS" file in a DFS volume.

application can read the file directly. Refer to chapter 10 in *Installing and Using HP BASIC in the MS-DOS Environment* for further information.

A LIF Example. Now let's port a program written to access LIF volumes on an external HP-1B disk drive. A typical MSVS for a LIF volume on an external HP-IB disk drive would be ":,700,0" (to access drive 0 at select code 7, address 00). The following program does not specify the current mass storage device, but specifies the MSVS each time the file name is specified:

```
340    ! Create file and store data from array "Trace_a"
350    !
360    !
370    CREATE BDAT "DATAFILE:,700,0",13
380    !
390    ASSIGN @File TO "DATAFILE:,700,0"
400    !
410    OUTPUT @File;Trace_a(*)
420    ASSIGN @File TO *
430    !
440    END
```

In the following program, each LIF MSVS is replaced with the DFS MSVS ":DOS,A" to access MS-DOS disk drive "A:". Again, line 370 is modified to create a *DOS* type file.

```
340    ! Create file and store data from array "Trace_a"
350    !
360    !
370    CREATE "DATAFILE:DOS,A",1
380    !
390    ASSIGN @File TO "DATAFILE:DOS,A"
400    !
410    OUTPUT @File;Trace_a(*)
420    ASSIGN @File TO *
430    !
440    END
```

If you want to continue using external LIF mass storage with Release II, you don't need to change the program at all. Also, as with Release I, no software configuration is required for external LIF mass storage. Just use the appropriate MSVS for your HP-IB disk drive.

Converting Existing Files to DFS

If you have existing data or program files in LIF, HPW, SRM, or HFS volumes, you can convert the files to DFS by copying them to a DFS directory. You can use the HP BASIC "COPY" statement, which performs the conversion automatically, but only copies one file at a time. If you want to copy files from a LIF or HPW volume, you can use the "CATCOPY" utility, which can copy an entire LIF or HPW volume at once.

The CATCOPY Utility. Obviously, if you have a large collection of files in a LIF or HPW volume, it would be inconvenient to copy these files one-at-a-time with the COPY statement. But, you don't have to do it that way. An HP BASIC utility program called "CATCOPY" is provided with Release II. You can use "CATCOPY" to copy all of the files, or those files that you choose, from a LIF or HPW volume *on an internal disk drive* to a DFS directory. The "CATCOPY" utility automatically creates a legal DFS file name for each file in the destination directory. This utility is described in appendix A of this manual.

File Name Differences. Both the LIF and HPW file systems allow file names of up to 10 characters, with no file name extensions. LIF and HPW file names are "case-sensitive." That is, both uppercase and lowercase characters may be used to make different file names. The following are valid LIF or HPW file names (they are all different and may exist in the same volume):

```
DATAFILE
Datafile
DATAFILE01
DATAFILE02
```

The DFS file system uses the same file name conventions as MS-DOS. That is, DFS file names may include up to eight alphanumeric characters followed by a "dot-extension" of up to three alphanumeric characters. All alpha characters are "case-folded" into uppercase characters. The dot extension, if included, must follow a period. However, the dot extension (and the period) are optional. (The same rules are used in naming a DFS directory. Refer to chapter 7 in *Installing and Using HP BASIC in the MS-DOS Environment* for a complete discussion of this topic.)

Thus, all four of the above file names will map to the same DFS file name, which can cause a problem when you copy LIF or HPW files to the DFS file system. You may need to modify file names to keep each file unique. For example, the above four file names could be converted to the following for DFS:

```
DATAFILE
DATAFILE.A
DATAFILE.01
DATAFILE.02
```

SRM file names may have up to 16 characters and HFS file names may have up to 14 characters. When you copy such a file to a DFS directory, you may have to shorten the file name.

The COPY Statement. You can use the HP BASIC "COPY" statement to copy files from one file system to another. The necessary conversions are performed automatically. For example, specify the file name and MSVS of a LIF or HPW file as the "source" and a file name and DFS directory path as the "destination":

```
COPY "File1:,1500,0" TO "\BLP\FILE1:DOS,C" Enter
```

👆 You can use the HP BASIC "COPY" statement to copy *individual* files only. You *cannot* use "COPY" to copy an entire volume or directory. If you want to copy all **Note** files from an HPW or LIF volume to a DFS directory, use the "CATCOPY" utility.

Remember that DFS file names are case folded into uppercase, and that they have a maximum of eight characters plus a three character extension. If you execute the following statement:

```
COPY "Filename01:,1500,0" TO "\BLP\Filename01:DOS,C" Enter
```

the destination file name will be truncated and case folded to "FILENAME". *If a file with the name "FILENAME" already exists, you will get a duplicate file name error.* You can avoid this problem by specifying a legal DFS file name for the destination file:

```
COPY "Filename01:,1500,0" TO "\BLP\FILENAME.01:DOS,C" Enter
```

You may find it more convenient to make the destination directory the current mass storage, and then copy one or more files. For example:

```
MSI "\BLP:DOS,C" Enter
COPY "Filename01:,1500,0" TO "FILENAME.01" Enter
COPY "Filename02:,1500,0" TO "FILENAME.02" Enter
```

Here are some SRM and HFS examples:

```
COPY "/DATA/DATAFILE01:REMOTE" TO "\BLP\DATAFILE.01:DOS,C" Enter
COPY "/PROJECTS/DATAFILE_A:,700" TO "\BLP\DATAFILE.A:DOS,C" Enter
```

You can use the COPY statement to copy any file *type*: PROG, BDAT, ASCII, or HP-UX. If you copy an HP-UX file from a LIF or HPW volume (or an SRM or HFS directory), the destination file in the DFS directory will be a DOS file. Otherwise, the file type of the DFS destination file will be the same as the source file.

Display Issues

The Release II software no longer emulates an HP 9000 Series 200 display system (for example, the 9836C) using PC hardware. Instead, a Series 300 style bit-mapped display is used, which uses the full area of the screen with no clipping of graphics images. As with Series 300, the default display mode is *combined* (merged) alpha and graphics. This display system is described in detail in chapter 6 of *Installing and Using HP BASIC in the MS-DOS Environment*. We'll cover some specific porting issues here.

No More Clipping Corrections

With bit-mapped graphics, you no longer need to correct for clipping of the graphics image as you did for Release I (with VIEWPORT and WINDOW statements). This means, however, that you should "undo" such corrections in any graphics programs you wrote for Release I.

If a Release I program queried the BASIC system for display specific information, it likely used statements like the following:

```
OUTPUT 19; "HPWSTATUS"
ENTER 19; Sw_rev$,Disp_size$,Card_rev,Machine$,Dot_ratio
```

The reason is that with Release I the standard system functions, such as SYSTEM$, RATIO, and GESCAPE, would return the values corresponding to the *emulated hardware* (the HP 9816, HP 9836A, or HP 9836C) instead of the actual values for the PC display system.

For example, the following Release I program segment eliminates clipping and defines a proper aspect ratio for an EGA display being used for HP 9836C emulation:

```
100    ! This segment corrects for clipping for Release I.
110    !
120    OUTPUT 19;"HPWSTATUS"
130    ENTER 19;Sw_rev$,Disp_size$,Card_rev,Machine$,Dot_ratio
140    Ega_ratio=(4/3)/(350/390)*Dot_ratio
150    GINIT
160    PLOTTER IS CRT,"INTERNAL"
170    VIEWPORT 0,100*(4/3),0,100*(350/390)
180    WINDOW 0,100*Ega_ratio,0,100
```

This correction is no longer necessary for Release II. You can use the normal RATIO statement just as you would with Series 300. The above Release I segment can be replaced with the following for Release II.

```
100    ! No clipping correction needed for Release II
110    !
120    GINIT
130    PLOTTER IS CRT,"INTERNAL"
140    VIEWPORT 0,100*RATIO,0,100
```

Combined Alpha and Graphics

When you reboot the language processor with Release II software, the default display mode is *Series 300 combined alpha and graphics*. The alpha and graphics planes are combined, or *merged*, into one display plane. The "ALPHA ON/OFF" and "GRAPHICS ON/OFF" statements (and the corresponding keyboard functions) have no effect. If you have a VGA or EGA display system, you can go into a Series 200 style separate alpha/graphics mode by executing the following HP BASIC statement. (If you execute this statement with a non-VGA monochrome display system, you will receive an error message.)

```
SEPARATE ALPHA FROM GRAPHICS
```

To return to Series 300 style combined alpha and graphics, execute:

```
MERGE ALPHA WITH GRAPHICS
```

In porting your Release I programs to Release II, you may want to include these statements to achieve the same effect as before. Or you may want to remove the "ALPHA ON/OFF" and "GRAPHICS ON/OFF" statements from your program and use combined alpha and graphics.

The alignment of the alpha and graphics displays will be different than that of Release I. (This is true for both the combined and separate alpha/graphics modes.) This is because Release II uses the entire width of the screen for graphics, while Release I did not.

Alpha Height

There is one other display porting issue. With Release II, the alpha screen has a total height (including the softkey labels) of 25 to 34 lines, depending on the type of display. For example, the VGA display has 34 lines, the EGA display has 25 lines, and the HP Multimode Display has 28 lines. To ensure that the amount of alphanumeric data displayed is not display dependent, you can include the following statement in your program:

```
ALPHA HEIGHT 25
```

Porting from Series 200/300 to Release II

This chapter will help you port your programs written for the HP 9000 Series 200/300 (BASIC 5.0/5.1) environment to run optimally with Release II of the HP BASIC Language Processor. We say "optimally" because most programs written for Series 200/300 (BASIC 5.0/5.1) will run in the Release II environment with no changes. However, you may need to make some changes in order to make use of Release II's increased performance.

Note This manual assumes that you have installed Release II in your computer following the instructions in chapter 2 of *Installing and Using HP BASIC in the MS-DOS Environment*. It is also assumed that you are porting programs written for BASIC 5.0/5.1. If you are porting from Series 200/300 BASIC 3.0 or 4.0, you will need to port to BASIC 5.0/5.1 first, then to the language processor. Refer to *BASIC 5.0/5.1 Programming Techniques, Volume II: Porting Information* for further information.

There are two primary areas of interest in porting a program from Series 200/300 to Release II: *mass storage* and the *display*. Let's look at mass storage first.

Mass Storage Issues

The HP BASIC Language Processor Release II software provides three mass storage *file systems*, which are described in detail in chapter 7 of *Installing and Using HP BASIC in the MS-DOS Environment*. These are the *LIF* and *HPW* file systems, which provide compatibility with the Series 200/300 and Release I environments, and the new DOS File System, or *DFS*, which provides increased access speed and direct access to the MS-DOS hierarchical file structure. *It is recommended that you use DFS file access for optimum performance and to allow direct sharing of files with MS-DOS applications.* (Release II also supports the HP 82313A Hierarchical File System, or *HFS*, which is available separately.)

When you write new programs, you can use DFS file access exclusively. Just follow the examples given in *Installing and Using HP BASIC in the MS-DOS Environment*. When you port a program from Series 200/300, you need to do two things: 1) change the mass storage references and 2) copy existing files into a DFS directory. Let's look at the program modifications first.

Modifying Programs for DFS Mass Storage

The DFS mass storage file system is covered in detail in chapter 7 of *Installing and Using HP BASIC in the MS-DOS Environment*. In general, any statement in your Series 200/300 program that contains a MSVS (Mass Storage Volume Specifier) will need to be modified to access a DFS volume. A typical MSVS for a LIF volume would be ":,700,0" (to access drive 0 on an external HP-IB disk drive at select code 7, address 00).

Note

You can access LIF volumes on an external HP-IB disk drive with Release II. You won't need to modify your Series 200/300 programs unless the address or drive number changes. No software configuration is necessary for an external disk drive.

You can also access LIF flexible disk volumes using an internal flexible disk drive, but you must first configure the drive using the utility "CONF.EXE" as described in chapter 8 of *Installing and Using HP BASIC in the MS-DOS Environment*. *Note, however, that the new DFS mass storage system provides significantly faster mass storage access.*

You can access a DFS volume with an MSVS such as ":DOS,A" (to access drive A:) or ":DOS,C" to access drive C:. However, normally you will want to subdivide your hard disk into *directories* such as "C:\BLP" and "C:\BLP\DATA", for example. You can specify a full path in a mass storage specifier. For example, to specify the directory "C:\BLP\DATA" as the current mass storage, use the following statement.

```
MASS STORAGE IS "\BLP\DATA:DOS,C"
```

or

```
MSI "\BLP\DATA:DOS,C"
```

In addition to changing the mass storage specifiers, you may need to change the file name references in your program. DFS file names are limited to eight characters plus a "dot extension" of three characters. All alpha characters are "case-folded" into uppercase characters. Refer to "File Name Differences," later in this chapter, for further information.

A LIF Example. Let's port a short program segment. In the following program segment the MASS STORAGE IS statement declares ":,700,0" to be the current mass storage:

```
340    ! Create file and store data from array "Trace_a"
350    !
360    MASS STORAGE IS ":,700,0"
370    CREATE BDAT "DATAFILE01",13
380    !
390    ASSIGN @File TO "DATAFILE01"
400    !
410    OUTPUT @File;Trace_a(*)
420    ASSIGN @File TO *
430    !
440    END
```

This program segment creates a BDAT file named "DATAFILE01" in LIF volume ":,700,0" and then outputs data from the array "Trace_a" to that file.

Let's modify the program segment so that it creates a file in a DFS directory. Let's further modify the program so that it creates a data file of the *DOS* file type, rather than BDAT: [*]

```
340    ! Create file and store data from array "Trace_a"
350    !
360    MASS STORAGE IS "\BLP\DATA:DOS,C"
370    CREATE "DATAFILE.01",1
380    !
390    ASSIGN @File TO "DATAFILE.01"
400    !
410    OUTPUT @File;Trace_a(*)
420    ASSIGN @File TO *
430    !
440    END
```

Note that line 370 creates a DOS type file with only one record. Don't worry, though. DOS files are *extensible* — they "grow" to the size needed. The file name has been modified to "DATAFILE.01", which is a legal file name for DFS.

---

[*]   The "CREATE" statement, without the "BDAT" or "ASCII" secondary keyword, creates an "HP-UX" file in a LIF (or HFS) volume, but creates a "DOS" file in a DFS volume.

You can create and use BDAT files in a DFS directory. However, *the DOS file type is preferred for DFS file access* for two reasons: *higher performance* and *better compatibility* with MS-DOS applications. A DOS-type file is really a standard MS-DOS file — it has no "header." Thus an MS-DOS application can read the file directly. Refer to chapter 10 in *Installing and Using HP BASIC in the MS-DOS Environment* for further information.

An HFS Example. You may have noticed that DFS is very similar to HFS (Hierarchical File System) for Series 200/300. Let's port a program written for HFS file access to use DFS file access:

```
340    ! Create file and store data from array "Trace_a"
350    !
360    !
370    CREATE BDAT "/PROJECTS/DATAFILE:,700,0",13
380    !
390    ASSIGN @File TO "/PROJECTS/DATAFILE:,700,0"
400    !
410    OUTPUT @File;Trace_a(*)
420    ASSIGN @File TO *
430    !
440    END
```

In the following program, each HFS path and MSVS is replaced with one for a DFS directory. Note that HFS paths always use the forward slash (/), but DFS paths may use either the back slash (\) or the forward slash (/).

```
340    ! Create file and store data from array "Trace_a"
350    !
360    !
370    CREATE "\BLP\DATA\DATAFILE:DOS,C",1
380    !
390    ASSIGN @File TO "\BLP\DATA\DATAFILE:DOS,C"
400    !
410    OUTPUT @File;Trace_a(*)
420    ASSIGN @File TO *
430    !
440    END
```

Converting Existing Files to DFS

If you have existing data or program files in LIF, SRM, or HFS volumes, you can convert the files to DFS by copying them to a DFS directory. You can use the HP BASIC "COPY" statement, which performs the conversion automatically, but only copies one file at a time. If you want to copy files from a LIF volume, you can use the "CATCOPY" utility, which can copy an entire LIF volume at once.

The CATCOPY Utility. Obviously, if you have a large collection of files in a LIF volume, it would be inconvenient to copy these files one-at-a-time with the COPY statement. But, you don't have to do it that way. An HP BASIC utility program called "CATCOPY" is provided with Release II. You can use "CATCOPY" to copy all of the files, or those files that you choose, from a LIF volume *on an internal disk drive* to a DFS directory. The "CATCOPY" utility automatically creates a legal DFS file name for each file in the destination directory. This utility is described in appendix A of this manual.

File Name Differences. The LIF file system allows file names of up to 10 characters, with no file name extensions. LIF file names are "case-sensitive." That is, both uppercase and lowercase characters may be used to make different file names. The following are valid LIF file names (they are all different and may exist in the same volume):

```
DATAFILE
Datafile
DATAFILE01
DATAFILE02
```

The DFS file system uses the same file name conventions as MS-DOS. That is, DFS file names may include up to eight alphanumeric characters followed by a "dot-extension" of up to three alphanumeric characters. All alpha characters are "case-folded" into uppercase characters. The dot extension, if included, must follow a period. However, the dot extension (and the period) are optional.* (The same rules are used in naming a DFS directory. Refer to chapter 7 in *Installing and Using HP BASIC in the MS-DOS Environment* for a complete discussion of this topic.)

Thus, all four of the above file names will map to the same DFS file name, which can cause a problem when you copy LIF files to the DFS file system. You may need to modify file names to keep each file unique. For example, the above four file names could be converted to the following for DFS.

---

* MS-DOS *reserves* certain file names and extensions, which have a special meaning. Refer to your MS-DOS *User's Reference* manual for further information.

```
DATAFILE
DATAFILE.A
DATAFILE.01
DATAFILE.02
```

SRM file names may have up to 16 characters and HFS file names may have up to 14 characters. When you copy such a file to a DFS directory, you may have to shorten the file name.

The COPY Statement. You can use the HP BASIC "COPY" statement to copy files from one file system to another. The necessary conversions are performed automatically. For example, specify the file name and MSVS of a LIF file as the "source" and a file name and DFS directory path as the "destination". (In this example, the source is a file on an HP-IB LIF disk drive.)

```
COPY "File1:,700,0" TO "\BLP\FILE1:DOS,C" [Enter]
```

Note You can use the HP BASIC "COPY" statement to copy *individual* files only. You *cannot* use "COPY" to copy an entire volume or directory. If you want to copy all files from a LIF volume to a DFS directory, use the "CATCOPY" utility.

Remember that DFS file names are case folded into uppercase, and that they have a maximum of eight characters plus a three character extension. If you execute the following statement:

```
COPY "Filename01:,700,0" TO "\BLP\Filename01:DOS,C" [Enter]
```

the destination file name will be truncated and case folded to "FILENAME". *If a file with the name "FILENAME" already exists, you will get a duplicate file name error.* You can avoid this problem by specifying a legal DFS file name for the destination file:

```
COPY "Filename01:,700,0" TO "\BLP\FILENAME.01:DOS,C" [Enter]
```

You may find it more convenient to make the destination directory the current mass storage, and then copy one or more files. For example:

```
MSI "\BLP:DOS,C" [Enter]
COPY "Filename01:,700,0" TO "FILENAME.01" [Enter]
COPY "Filename02:,700,0" TO "FILENAME.02" [Enter]
```

Here are some SRM and HFS examples:

```
COPY "/DATA/DATAFILE01:REMOTE" TO "\BLP\DATAFILE.01:DOS,C" [Enter]
COPY "/PROJECTS/DATAFILE_A:,700" TO "\BLP\DATAFILE.A:DOS,C" [Enter]
```

You can use the COPY statement to copy any file *type*: PROG, BDAT, ASCII, or HP-UX. If you copy an HP-UX file from a LIF volume (or an SRM or HFS directory), the destination file in the DFS directory will be a DOS file. Otherwise, the file type of the DFS destination file will be the same as the source file.

Display Issues

When porting to Release II, the graphics issues depend on whether you are porting from Series 300 or from Series 200. There is one display issue that is applicable to both Series 300 and Series 200, however. That is the issue of alpha height.

Alpha Height

With Release II, the alpha screen has a total height (including the softkey labels) of 25 to 34 lines, depending on the type of display. For example, the VGA display has 34 lines, the EGA display has 25 lines, and the HP Multimode Display has 28 lines. To ensure that the amount of alphanumeric data displayed is not display dependent, you can include the following statement in your program:

ALPHA HEIGHT 25

Series 300 Graphics Issues

Release II uses a Series 300 style bit-mapped display, which uses the full area of the screen with no clipping of graphics images. As with Series 300, the default display mode is *combined* (merged) alpha and graphics. Thus, you won't need to modify your graphics programs in most cases. You may want to refer to chapter 6 of *Installing and Using HP BASIC in the MS-DOS Environment* for a detailed description of the Release II bit-mapped display.

There are very few porting issues for Series 300 graphics, *provided you are using a VGA monitor.* For other monitors, there are some aspect ratio issues, which are described in chapter 6 of *Installing and Using HP BASIC in the MS-DOS Environment*. The VGA color display provides good compatibility with the color-mapped-mode pen colors produced by Series 300 (pens 0 through 15). However, if you are using an EGA display, pen colors 8 through 15 are not a good match. (Refer to chapter 10 in *Programming with HP BASIC.*) *For optimum compatibility with Series 300 color graphics, a VGA color display system is highly recommended.*

There is another issue that may affect some programs. The alignment of the alpha and graphics displays is dependent on the display hardware in use. This is true for both the language processor and for Series 300. Thus, if you have written a program that mixes alpha text with graphic images, the positioning of the alpha and graphics elements will be different than for Series 300. You will

have to experiment to achieve the desired effect. (This is true for both the combined and separate alpha/graphics modes.)

Series 200 Graphics Issues

When you are porting a program written for a Series 200 computer to Release II, you will have to confront the same issue that arises when you port from Series 200 to Series 300: *combined* alpha and graphics planes.

When you reboot the language processor with Release II software, the default display mode is *Series 300 combined alpha and graphics*. The alpha and graphics planes are combined, or *merged*, into one display plane. The "ALPHA ON/OFF" and "GRAPHICS ON/OFF" statements (and the corresponding keyboard functions) have no effect. If you have a VGA or EGA display system, you can go into a Series 200 style separate alpha/graphics mode by executing the following HP BASIC statement. (If you execute this statement with a non-VGA monochrome display system, you will receive an error message.)

SEPARATE ALPHA FROM GRAPHICS

To return to Series 300 style combined alpha and graphics, execute:

MERGE ALPHA WITH GRAPHICS

In porting your Series 200 programs to Release II, you may want to include these statements to achieve the same effect as before. Or you may want to remove the "ALPHA ON/OFF" and "GRAPHICS ON/OFF" statements from your program and use Series 300 style combined alpha and graphics.

There is another issue that may affect some programs. The alignment of the alpha and graphics displays is dependent on the display hardware in use. Thus, if you have written a program that mixes alpha text with graphic images, the positioning of the alpha and graphics elements will be different than for Series 200. You will have to experiment to achieve the desired effect.

Other Porting Issues

Appendix B gives a complete summary of the keyword differences for Release II compared with Series 200/300. The table includes enhancements as well as functions not implemented. In particular, you should note the following (these statements are true for both Release I and Release II):

- Serial TRANSFERs are not supported. (Use ENTER and OUTPUT for the serial interface.)

- The SOUND keyword is not supported. (Use BEEP instead.)

- The WAIT statement returns in 80 percent of the specified time. For example, use "WAIT 5" to wait four seconds.

- The READIO and WRITEIO statements access hardware registers and memory locations directly. Thus, a detailed knowledge of the hardware of the language processor is necessary. If you have written a Series 200/300 program using READIO and WRITEIO, it is likely that the statements won't work due to hardware differences. It is recommended that you use the STATUS and CONTROL statements instead.

Caution

It is possible to damage your language processor system by writing to certain memory locations with WRITEIO. Hewlett-Packard cannot be held liable for any damages caused by improper use of this feature.

A

Using the CATCOPY Utility to Convert Files

The "CATCOPY" utility is an interactive HP BASIC program that allows you to copy files from an HPW (virtual-LIF) or LIF disk volume to a DFS directory. With CATCOPY you can copy all of the files in an HPW or LIF volume to a DFS directory in a single batch process. Or you can use the "interactive" mode and copy only those files that you choose. In either case, the *source* can be either an HPW volume or a LIF volume in one of the *internal* PC disk drives (select code 15). *
The *destination* can be any DFS directory that you specify.

Note

Before you run the CATCOPY program, you need to configure the internal disk drive or drives that you want to access through select code 15. Use the "CONF.EXE" utility to do this, as described in chapter 8 of *Installing and Using HP BASIC in the MS-DOS Environment*.

If you have Release I installed in a directory such as "C:\HPW", it is recommended that you configure that directory, copy all of your program and data files to a DFS directory using CATCOPY, and then *un-configure* C:\HPW. This will protect against accidental confusion between DFS and HPW directories. Refer to chapter 7 in *Installing and Using HP BASIC in the MS-DOS Environment* for further information.

---

* You cannot use CATCOPY to copy files from an external disk drive such as an HP-IB disk drive at select code 7. If you want to copy files from a LIF flexible disk, configure one of the internal PC flexible disk drives with the "CONF.EXE" utility and use that drive as the source drive. You cannot use CATCOPY to copy files from an SRM or HFS directory. (Use the COPY statement instead.)

Running CATCOPY

Once HP BASIC is running you can start the CATCOPY program as follows. Insert the "Manual Examples and Selected CSUBs" disk in drive A: and type:

LOAD "CATCOPY:DOS,A" Enter

and then press Run (F3).

The first two screens are informational — they describe what the program does. Read these screens if you wish, and press the Continue softkey (F2) each time to go to the next screen.

The third screen prompts you to identify the *source* disk drive or volume. In the following example, two select code 15 disk drives have been configured:

```
                        CATCOPY DISK UTILITY

Following disks are configured as PC internal mass storage.

                    :,1500,0
                    :,1500,1









Please enter a source drive number from 0 to 1

__
```

Let's assume that you have configured the PC drives as follows: ":,1500,0" is drive A: and ":,1500,1" is directory C:\HPW on drive C:. If you want to copy files from the latter, type:

1 Enter

and the following screen will appear:

A-2   Using the CATCOPY Utility to Convert Files

```
Do you want to proceed with copy or do you want to do CAT of :,1500,1?

Please press a softkey.


                                                  User 1    Caps   Running
          Continue              CAT                                 Done
                                                                      ■
```

You are now asked whether you want to list a CAT of ":,1500,1". To do so, press CAT ([F5]). The following is an example of a typical listing:

```
                         CATCOPY DISK UTILITY

                          Cat of :,1500,1
:,1500,1
VOLUME LABEL: HPW_C
FILE NAME PRO TYPE  REC/FILE BYTE/REC   ADDRESS    DATE       TIME

REVID            ASCII       2     256        26   7-Mar-89 16:34
SYSTEM_BA5       SYSTM    2370     256        28   7-Mar-89 16:38
TEST_PROG        PROG        1     256      2398  16-Mar-89 11:01
DATAFILE01       BDAT        1     256      2399  16-Mar-89 13:34
DATAFILE02       BDAT        1     256      2400  16-Mar-89 13:43
Datafile01       HP-UX     256       1      2401   2-Apr-89 10:11



Do you want to proceed with copy or do you want to do CAT of :,1500,1?

Please press a softkey.


                                                  User 1    Caps   Running
          Continue              CAT                                 Done
                                                                      ■
```

The CAT listing shows all of the files in C:\HPW (":,1500,1"). The "REVID" file identifies the revision of the language processor software in "C:\HPW". The "SYSTEM_BA5" file is the HP BASIC 5.0 system file in "C:\HPW". The remaining files are program and data files — it is these files that you will want to copy to your new system directory. Press Continue to proceed.

```
                        CATCOPY DISK UTILITY

   Please enter the complete destination DOS directory path name,
   for example C:\BLP\PROGS

   If the destination DOS directory does not exist, it can be created
   at your option.




   Destination DOS directory name?

   _
```

Enter the name of the DFS *destination* directory to which you want to copy your files. (If the directory doesn't already exist, you will be given the option of having CATCOPY create the directory.) For example, type:

C:\BLP\DATA [Enter]

Choosing the "Auto" or "Manual" Copy Method

Once you have entered a destination directory, the following screen appears:

```
                          CATCOPY DISK UTILITY

    You can copy all files automatically or you can copy
    interactively, one by one, only those files you choose.












    Please select Auto for automatic or Manual for interactive copy.


                                              User 1    Caps    Running
     ┌───────┐┌────────┐┌──────┐┌──────┐   ┌──────┐┌──────┐┌──────┐┌──────┐
     │ Auto  ││ Manual ││      ││      │   │      ││      ││      ││      │
     └───────┘└────────┘└──────┘└──────┘   └──────┘└──────┘└──────┘└──────┘  ▪
```

You can choose either Auto or Manual. The "Auto" method copies all files (except "SYS-
TEM" files) from the source to the destination directory. Files may include both program and data
files. The "Manual" method prompts you to confirm each file before it is copied.

Let's look at the "Auto" procedure first.

The "Auto" Copy Method

To select the "Auto" copy method, press Auto ([F1]) in the previous screen. Now you are prompted to select a method for naming the destination file if the source file name is longer than eight characters. (Remember that DFS file names can consist of up to eight characters, plus a "dot extension" of up to three characters.) You can choose either Truncate ([F1]) or Extension ([F2]):

```
Please choose a method.



                                               User 1    Caps    Running
   ┌────────┬────────┬──────┬──────┐    ┌──────┬──────┬──────┬──────┐
   │Truncate│Extensio│      │      │    │      │      │      │      │
   │        │n       │      │      │    │      │      │      │      │   ■
   └────────┴────────┴──────┴──────┘    └──────┴──────┴──────┴──────┘
```

The "Truncate" Option. If you choose Truncate, each file name will simply be truncated to eight characters. (The file name will also be case folded into uppercase characters.) If truncation or case folding results in a file name that already exists, a single-character, alphabetic name extension will be appended. (You won't get an error message, as you would with the COPY statement.) Press Truncate ([F1]) and the following screen appears:

```
                    CATCOPY DISK UTILITY

COPYING files from drive: :,1500,1
To DOS directory: C:\BLP\DATA

All source file will be copied automatically without your input.
The old system files will not be copied automatically because they
cannot be used with the new language processor software.
If the source LIF file name is longer than 8 characters, the excess
characters will be truncated in the destination file name.
DOS file names are all uppercase. If case folding or truncation of the
source file names result in duplicate file names in the destination
directory, a letter from A..Z is appended to make the file name unique.
```

Press `Continue` to start the copying process:

```
┌────────────────────────────────────────────────────────────────────────┐
│                         CATCOPY DISK UTILITY                             │
│                                                                          │
│  COPYING: TEST_PROG:,1500,1 TO: \BLP\DATA\TEST_PRO:DOS,C                 │
│                                                                          │
│  COPYING: DATAFILE01:,1500,1 TO: \BLP\DATA\DATAFILE:DOS,C                │
│                                                                          │
│  COPYING: DATAFILE02:,1500,1 TO: \BLP\DATA\DATAFILE:DOS,C                │
│  DOS file \BLP\DATA\DATAFILE:DOS,C exists. Making a unique name.         │
│                                                                          │
│  COPYING: DATAFILE02:,1500,1 TO: \BLP\DATA\DATAFILE.A:DOS,C              │
│                                                                          │
│  COPYING: Datafile01:,1500,1 TO: \BLP\DATA\DATAFILE:DOS,C                │
│  DOS file \BLP\DATA\DATAFILE:DOS,C exists. Making a unique name.         │
│                                                                          │
│  COPYING: Datafile01:,1500,1 TO: \BLP\DATA\DATAFILE.A:DOS,C              │
│  DOS file \BLP\DATA\DATAFILE.A:DOS,C exists. Making a unique name.       │
│                                                                          │
│  COPYING: Datafile01:,1500,1 TO: \BLP\DATA\DATAFILE.B:DOS,C              │
│  Done with copying. Do you want to copy more files?                      │
│                                                                          │
│  Please press a softkey.                                                 │
│                                                                          │
│                                          User 1   Caps   Running         │
│   ┌─────┐┌────────┐┌─────┐┌─────┐   ┌─────┐┌─────┐┌─────┐┌──────┐        │
│   │     ││Restart ││     ││     │   │     ││     ││     ││ Done │        │
│   └─────┘└────────┘└─────┘└─────┘   └─────┘└─────┘└─────┘└──────┘        │
└────────────────────────────────────────────────────────────────────────┘
```

Note that the "REVID" and "SYSTEM_BA5" files are *not* copied by the "Auto" procedure. These files are part of the old BASIC system, and are not compatible with Release II.

The "Extension" Option. If you choose `Extension` instead of `Truncate`, the ninth and tenth characters of the LIF or HPW file name become an extension of the file name. Again, the file name is case folded. If case folding results in a file name that already exists, a single alphabetic character will be added as the third character in the file name extension. The procedure is essentially the same, but the final screen will appear as follows.

```
                        CATCOPY DISK UTILITY


  COPYING: TEST_PROG:,1500,1 TO: \BLP\DATA\TEST_PRO.G:DOS,C

  COPYING: DATAFILE01:,1500,1 TO: \BLP\DATA\DATAFILE.01:DOS,C

  COPYING: DATAFILE02:,1500,1 TO: \BLP\DATA\DATAFILE.02:DOS,C

  COPYING: Datafile01:,1500,1 TO: \BLP\DATA\DATAFILE.01:DOS,C
  DOS file \BLP\DATA\DATAFILE.01:DOS,C exists. Making a unique name.

  COPYING: Datafile01:,1500,1 TO: \BLP\DATA\DATAFILE.01A:DOS,C
  Done with copying. Do you want to copy more files?
```

As you can see, the "Extension" option fits this particular situation better than the "Truncate" option.

The "Manual" Copy Method

The "Manual" copy method works essentially like the "Auto" method of the previous section, but you will be prompted whether to copy each file. The following is a typical screen:

```
  Copy REVID:,1500,1 ?


                                              User 1    Caps    Running
        No Skip  Restart                   Yes Copy               Done
                                                                     ■
```

If you use the "Manual" copy method, you should skip "REVID" and any old "SYSTEM" files such as "SYSTEM_BA5".

Exiting CATCOPY

Whatever procedure you follow, when all files are copied you will be prompted to press the Restart softkey to begin again (perhaps with a new directory) or Done to quit.

A-8   Using the CATCOPY Utility to Convert Files

HP BASIC Keyword Differences

For the most part, Release II of the HP BASIC Language Processor software provides the same HP BASIC keyword functionality as does an HP 9000 Series 200/300 BASIC Workstation, as described in the keyword dictionary in the HP BASIC *Language Reference* manual. However, there are a few keywords that function differently for the language processor. Also, Release II provides additional functionality for several keywords, principally to support DFS mass storage.

Release II, like Release I, supports the HP 82313A Hierarchical File System (HFS). The keyword syntax for HFS is supported, but only if you have installed HFS on your language processor.

The keyword differences and additional functionality for Release II are summarized as follows:

BASIC Keyword	Series 200/300 BASIC Workstation	HP BASIC Language Processor Release II
ALPHA ON ALPHA OFF	Turn the alpha display on and off. For Series 300, they work only after SEPARATE ALPHA FROM GRAPHICS has been executed.	For VGA or EGA display, same as Series 300. For other displays, SEPARATE ALPHA FROM GRAPHICS is not supported and, thus, neither are ALPHA ON/OFF.
AREA	Specifies an area fill color.	Works the same as for Series 200/300 for a VGA color display. For an EGA display, the fill color selection is limited.
CAT	Lists the contents of a mass storage directory or provides information on a specified PROG file.	For LIF, SRM, and HFS directories, same as Series 200/300. For DFS directories the first four lines are: DIRECTORY: <directory path> LABEL: <drive label or empty> FORMAT: DOS AVAILABLE SPACE: <number of 256-bytes blocks>

BASIC Keyword	Series 200/300 BASIC Workstation	HP BASIC Language Processor Release II
CONTROL <I/O path>,9	Not applicable.	Used to select buffered or unbuffered mode of I/O path output to a DFS file. For example, the statement "CONTROL @File,9;0" selects unbuffered output for the I/O path "@File". The statement "CONTROL @File,9;1" restores buffered mode, the default.
CONTROL <I/O path>,10	Not applicable.	For buffered output to a DFS file, writing a "1" to this register causes all pending data in the buffer to be written to the disk file. For example, "CONTROL @File,10;1" causes all data to be written to the file assigned to the I/O path "@File".
CONTROL 32	This pseudo select code is used to turn on and off parity checking, system (memory) cache, and processor (instruction) cache.	Not supported.
CONTROL CRT,14	Sets display replacement rule.	Not supported.
CONTROL CRT,18	Sets alpha write-enable mask to a bit pattern.	Not supported. Use the SET ALPHA MASK statement instead.
CONTROL CRT,21	Switches between Series 200 CRT compatibility mode and native bit-mapped mode.	Not supported.
COPY	Copies a file from one volume (LIF, HFS, SRM) to another. Also can be used to copy an entire LIF or HFS volume, provided the destination volume is large enough.	Works same as for Series 200/300 except that copying an entire volume is not supported. Also, you can copy files to and from DFS and HPW volumes in the internal PC disk drives. If you want to copy an entire DFS directory, use the MS-DOS "COPY" command.

BASIC Keyword	Series 200/300 BASIC Workstation	HP BASIC Language Processor Release II
CREATE CREATE ASCII CREATE BDAT CREATE DIR	The first three statements create, respectively, an HP-UX file, an ASCII file, and a BDAT file. In each case, space is pre-allocated on the disk for the file. The files are extensible for SRM or HFS volumes, but not for LIF volumes. The CREATE DIR statement creates a new SRM or HFS directory.	These statements work the same as for Series 200/300 for LIF, SRM, and HFS volumes. In addition, HP-UX, ASCII, and BDAT files may be created in HPW volumes. For DFS, the CREATE statement creates a DOS file, while the other statements create ASCII files, BDAT files, and new directories. All file types are extensible for DFS.
ENTER 19	Not applicable.	Select code 19 is the language processor MS-DOS communication port. ENTER 19 is used by the language processor to obtain information from that port.
GRAPHICS ON GRAPHICS OFF	Turn the graphics display on and off. For Series 300, they work only after SEPARATE ALPHA FROM GRAPHICS has been executed.	For VGA or EGA display, same as Series 300. For other displays, SEPARATE ALPHA FROM GRAPHICS is not supported and, thus, neither are GRAPHICS ON/OFF.
INITIALIZE	Initializes (formats) mass storage media for use as LIF volumes. Also used to initialize RAM volumes.	Works the same as for Series 200/300 for LIF media in external HP-IB disk drives only. Cannot be used to initialize LIF media in internal PC drives (use the LIFINIT utility). In the internal drives, INITIALIZE formats an HPW (virtual-LIF) disk through select code 15. INITIALIZE cannot be used to format a DFS disk (use the MS-DOS "FORMAT" command).

BASIC Keyword	Series 200/300 BASIC Workstation	HP BASIC Language Processor Release II
KNOBX KNOBY	Returns the net number of horizontal knob pulses (KNOBX) or vertical knob pulses (KNOBY).	Works the same as for Series 200/300, except BASIC 2.0/2.1 compatibility (the KNB2_0 binary) is not supported.
MASS STORAGE IS MSI	Specifies the MSVS of the current mass storage volume. For HFS or SRM, can also specify the directory path name.	Works the same as for Series 200/300, but also adds functionality. For HPW volumes, access is through select code 15. For DFS mass storage, the MSI statement specifies the PC drive and directory path name (for example, "\BLP\DATA:DOS,C") of the current mass storage.*
ON/OFF CYCLE ON/OFF DELAY ON/OFF TIME ON/OFF TIMEOUT	These statements enable/cancel event-initiated branches.	Works the same as for Series 200/300. However (due to hardware differences) the delay time is only 80 percent of that requested. So use 1.25 to get a delay of 1 second.
OUTPUT 19	Not applicable.	Select code 19 is the language processor MS-DOS communication port. OUTPUT 19 is used by the language processor to send information to that port.
PASS CONTROL	Passes Active Controller capability to a specified HP-IB device.	On the language processor's "built-in" HP-IB, PASS CONTROL works just as with Series 200/300. However, it is not supported for PC plug-in HP-IB cards.

* All mass storage statements that specify a file (for example, LOAD, STORE, RE-STORE, GET, SAVE and RE-SAVE), may include a DFS volume and directory specification.

BASIC Keyword	Series 200/300 BASIC Workstation	HP BASIC Language Processor Release II
PERMIT	Modifies the owner, group, or public access permissions of an HFS file or directory.	Works the same as for Series 200/300 for HFS (if present). However, PERMIT is not implemented for DFS. To change the permission fields for DFS, use the MS-DOS "ATTRIB" statement.
PRINT LABEL	Gives a name to a mass storage volume.	Not supported on DFS.
PURGE	The PURGE statement deletes individual files from a volume or directory. For HFS and SRM volumes, PURGE can be used to delete an empty directory from its superior directory.	Works the same as for Series 200/300. In addition, PURGE can be used for DFS mass storage to delete files and empty directories. Note that PURGE cannot delete a DFS file that has its MS-DOS read-only attribute set.
READIO		This function reads the contents of hardware registers or memory locations directly. Use of READIO and WRITEIO requires specific knowledge of the hardware. It is recommended that you use the STATUS and CONTROL statements instead.
READ LABEL	Reads a volume label into a string variable.	Not supported on DFS.
RENAME	Changes a file name or (for HFS or SRM) a directory name.	Works the same as for Series 200/300, but also renames DFS files and directories.
SCRATCH BIN	Performs all functions of SCRATCH A, but in addition, clears all binaries except the CRT driver for the CRT in use.	Works the same as with Series 200/300. However, note that SCRATCH BIN deletes the DFS binary, thus LOAD BIN cannot subsequently load binaries from the DFS disk files.

BASIC Keyword	Series 200/300 BASIC Workstation	HP BASIC Language Processor Release II
SEPARATE ALPHA FROM GRAPHICS	Puts Series 300 computer into Series 200-style separate alpha and graphics display mode (for multiple-plane displays only).	For VGA or EGA display, same as Series 300. Not supported for other displays.
SET CHR	Redefines the bit-pattern used for characters in the current font (on bit-mapped alpha/graphics displays only).	Works the same as Series 300, except that all bits in the bit pattern for a character must be of the same color.
SET PEN	Defines the color for one or more entries in the color map.	With a VGA color display, works same as for Series 300. However, with an EGA display, the color selections are limited.
SET TIME	Resets the time-of-day given by the real-time clock.	With MS-DOS 3.1 and 3.2, SET TIME affects only the "local" BASIC time and MS-DOS time. With MS-DOS 3.3 and above, SET TIME also sets the CMOS, battery-backed clock (the real time clock on the PC).
SET TIMEDATE	Resets the absolute seconds (time and date) given by the real-time clock.	With MS-DOS 3.1 and 3.2, SET TIMEDATE affects only the "local" BASIC time and MS-DOS time. With MS-DOS 3.3 and above, SET TIMEDATE also sets the CMOS, battery-backed clock (the real time clock on the PC).
SOUND	Generates a single tone or multiple tones on the sound generator of an HP-HIL interface.	Not supported for the language processor. (Use BEEP instead.)

BASIC Keyword	Series 200/300 BASIC Workstation	HP BASIC Language Processor Release II
STATUS <I/O path>,9	Not applicable	Used to determine the mode of output to a DFS file through an I/O path. A returned value of "1" indicates buffered mode and a value of "0" indicates unbuffered mode.
SYSTEM$ ("MASS MEMORY")	Returns X000YZ0000000000 where: X = Number of internal disk drives. Y = Number of initialized EPROM cards. Z = Number of bubble memory cards. (If Y or Z exceeds 9, an asterisk appears.)	Always returns 0000000000000000
SYSTEM$ ("MASS STORAGE IS")	Returns the mass storage volume specifier (MSVS) of the current MASS STORAGE IS device as it appears in the CAT heading.	For DFS, returns "<dir_path>:DOS,<dr_letter>". Otherwise, same as Series 200/300.
SYSTEM$ ("SERIAL NUMBER")	Returns one of the following: Bytes 4-14 of an ID PROM (if present). An encoded serial number from an HP-HIL ID module (if present). Otherwise, returns null string.	Returns: An encoded serial number from an HP-HIL ID module (if present). Otherwise, returns the string "111111111".

BASIC Keyword	Series 200/300 BASIC Workstation	HP BASIC Language Processor Release II
TRANSFER	Initiates unformatted I/O transfers, which can take place concurrently with program execution.	Generally works the same as for Series 200/300. However, overlapped transfers on DFS, HPW, or emulated LIF disks are not supported. Also, serial transfers are not supported. Refer to chapter 15 in *Programming with HP BASIC* for further information.
WAIT	Causes the computer to wait approximately the specified number of seconds before executing the next statement.	Works the same as for Series 200/300, however the wait period is only 80 percent of the specified time. So use "WAIT 1.25" to wait 1 second.
WRITEIO		This function writes directly to hardware registers or memory locations. Use of READIO and WRITEIO requires specific knowledge of the hardware. It is recommended that you use the STATUS and CONTROL statements instead.

Caution

It is possible to damage your language processor system by writing to certain memory locations with WRITEIO. Hewlett-Packard cannot be held liable for any damages caused by improper use of this feature.

Porting Programs That Use CSUBs

The HP BASIC Language Processor Release II software is compatible with most CSUBs (Compiled Subprograms) written for HP 9000 Series 200/300 BASIC Workstations or earlier language processor versions. However, there are exceptions. *If a CSUB attempts to access the hardware directly (for example, graphics memory), it will have to be modified for Release II.*

CSUB Errors

If your program uses a CSUB that is not compatible with the language processor, you will receive an HP BASIC error message in the range 369 through 398. Errors in this range indicate a CSUB Pascal error (probably because of a hardware dependency), for example:

```
ERROR 389 IN 4    CSUB Pascal error -11
```

You can obtain information about Pascal errors in the *Pascal Workstation System* manual.

CSUBs Provided with Release II

The graphics CSUBs "BPLOT", "GDUMP_C", and "GDUMP_R" access the hardware directly. Thus, special versions of these CSUBs are provided with the HP BASIC Language Processor Release II software. They are found on the "Manual Examples and Selected CSUBs" disk (file names "BPLOT", "GDUMP_C", and "GDUMP_R"). You must use the correct versions of these CSUBs in your programs. The Series 200/300 versions won't work with the Release II software — neither will the versions supplied with earlier versions of the language processor.

If you are porting a program that uses one of the graphics CSUBs from Series 200/300, or from an earlier version of the language processor, you will have to replace the CSUBs with the Release II versions. To see how to do this, let's look at an example. Suppose the following program was created, and the "BPLOT" CSUBs "Bload" and "Bstore" were loaded, on an HP 9000 Series 300 BASIC Workstation.

```
10     ! Bstore with VGA (Pixel_ratio = 1.0)
20
30     GINIT
40     OPTION BASE 0
50     INTEGER Ge(7)
60     INTEGER Size,A(100)
70     GESCAPE CRT,3;Ge(*)
80     WINDOW 0,Ge(2),Ge(3),0
90     MOVE 0,0
100    Size=10
110    RECTANGLE Size,Size,FILL
120    Bstore(A(*),Size+1,Size+1)
130    END
140    CSUB Bload(INTEGER A(*),Xpixels,Ypixels,OPTIONAL INTEGER Rule,REAL Xstart,Ystart)
150    CSUB Bstore(INTEGER A(*),Xpixels,Ypixels,OPTIONAL INTEGER Rule,REAL Xstart,Ystart)
```

The same program will run on the language processor once you have done the following:

1. Boot HP BASIC on the language processor.
2. LOAD the program (from a LIF flexible disk, for example).
3. Remove the Series 300 CSUBs by executing the following statement from the HP BASIC command line:

    DELSUB Bload,Bstore Enter

4. Insert the "Manual Examples and Selected CSUBs" disk in drive A:.
5. Load the Release II versions of the same CSUBs by executing:

    LOADSUB ALL FROM "BPLOT:DOS,A" Enter

The procedure is the same for programs using "GDUMP_C" or "GDUMP_R", except the HP BASIC statements are:

DELSUB Gdump_colored Enter

and then:

LOADSUB ALL FROM "GDUMP_C:DOS,A" Enter

or

DELSUB Gdump_rotated Enter

and then:

LOADSUB ALL FROM "GDUMP_R:DOS,A" Enter

C-2   Porting Programs That Use CSUBs

HEWLETT
PACKARD