

PolyPascal Supplementary Documentation  
for PolyPascal-86 V3.10 (CP/M-86)

Copyright (C) 1985  
PolyData MicroCenter A/S

Thank you for purchasing the PolyPascal package. We are confident that you will find PolyPascal to be a high quality product. Before moving on, please read the License Agreement enclosed to the package, and then fill in and return the accompanying registration form.

1. Distribution Disk Overview

---

You will find the following files on your distribution disk:

- |            |   |
|------------|---|
| README.DOC | The file you are currently reading.   |
| PPAS.CMD   | This file is the actual PolyPascal system. It contains the editor, the compiler and the run-time library.   |
| PPAS.HLP   | The help message file. This file contains the help messages displayed by the HELP command.  |
| PPAS.ERM   | The error message file. This file contains the error messages displayed by the compiler when it locates an error.   |
| PPBS.CMD   | The Business version of PolyPascal. The Business version uses 10-byte BCD reals with 18 significant digits and furthermore provides an advanced formatting function for numbers and strings.  |
| PP87.CMD   | The 8087 version of PolyPascal. If your computer is equipped with the 8087 NDP (numeric data processor) you will probably want to use this version of PolyPascal instead of the standard version, since it offers significant execution speed improvements on floating point operations (from 10 to 100 times faster), as well as greater precision and exponent range (15 significant digits, 1E-300 to 1E+300). The 8087 version is fully source code compatible with the standard version. |

INSTALL.CMD      The INSTALL program. If you have bought an "unconfigured" copy of PolyPascal, i.e. a copy which is not set up for any specific terminal or computer system, then before you can use it, you must install it using this program. If you try to run an unconfigured copy, you will simply get an error message. The INSTALL program is self-explaining, but to answer the questions asked by it you will probably have to consult your system documentation. Note that the Business version and the 8087 version are always shipped as unconfigured copies, and thus they cannot be used before you have installed them. If you wish, you may also use the INSTALL program to customize an already installed copy to suit your individual needs.

INSTALL.TRM  
INSTALL.DAT      Datafiles used by INSTALL. These files contain configuration data for up to 40 different terminals and computers. Note that INSTALL will not run if these files are missing.

INSTALL.DOC      This file contains additional documentation on how to install PolyPascal.

INSTALL.PAS      INSTALL program source text. INSTALL is written entirely in PolyPascal.

CALC.PAS          Demonstration program source text. This program acts as a calculator. It allows you to enter numeric expressions and have them evaluated.

HEXDUMP.PAS      Demonstration program source text. HEXDUMP will output a hex listing of any file to any other file or the printer.

PRIMES.PAS        Demonstration program source text. A short program which computes all prime numbers between 1 and 30000.

QSORT.PAS        Demonstration program source text. QSORT generates a list of random numbers and then sorts them using the quicksort algorithm.

LIST.PAS          Demonstration program source text. LIST is a very useful all-purpose list program. It can output listings of all sorts of text files, with headers, line numbers, and full control of page length, line length, left margin, and bottom margin.

CROSSREF.PAS Demonstration program source text. CROSSREF outputs a listing and a cross-reference map of any PolyPascal source text.

DATMAN.PAS These files constitute the PolyFile package. Poly-  
 NPFKEY.PAS File is a complete set of utility routines which  
 ADDKEY.PAS allow you to use indexed files with your PolyPascal  
 DELKEY.PAS programs. The PolyFile routines are written entire-  
 ly in PolyPascal. For further details on PolyFile,  
 please refer to the PolyFile Reference Manual which  
 is included in your PolyPascal documentation.

DATABASE.PAS Demonstration program source text. DATABASE uses  
 the PolyFile routines to maintain a simple customer  
 data base.

GSX.DOC GSX.PAS contains a number of routines to interface  
 GSX.PAS the GSX graphics operating system. If GSX is sup-  
 ported by your computer, you may use these routines  
 to generate high-resolution graphics from your  
 PolyPascal programs. For further details, please  
 refer to the GSX.DOC file.

Before you start using PolyPascal, we strongly recommend that you make a backup copy of this disk. To generate a PolyPascal work disk, copy PPAS.COM, PPAS.HLP and PPAS.ERM onto a blank disk along with any other system programs you may require (e.g. PIP.COM and STAT.COM). If you are short of disk space, you may omit PPAS.HLP and PPAS.ERM, but then the HELP command will not work and the compiler will only output error numbers.

## 2. PolyPascal compared to COMPAS Pascal

---

PolyPascal was formerly called COMPAS Pascal, and PolyPascal V3.10 is the successor of COMPAS Pascal V3.07. PolyPascal is fully source code compatible with COMPAS Pascal, except for the following minor differences (all section references address the PolyPascal Programming Manual):

- o The TEXT identifier is a reserved word in PolyPascal.
- o PolyPascal handles logical I/O devices (CON:, LST:, etc.) differently than COMPAS Pascal. Specifically, the "eof" and "eoln" functions operate differently when applied to logical devices. In contrast to COMPAS Pascal, there is no difference between the way PolyPascal handles a disk file and a logical device. For further details, please refer to section 13.3.

- o The "blockread" and "blockwrite" standard procedures require four parameteres instead of three. The fourth parameter returns the actual number of records transferred. For further details, please refer to section 13.4.
- o In PolyPascal, overlay files are opened and closed everytime they are accessed. Therefore, there is never a need to specifically close an overlay file. The Y compiler directive is no longer supported. Instead, the "ovdrive" standard procedure may be used to specify the drive on which overlay files reside. For further details, please refer to section 15.9.
- o The "chain" and "execute" standard procedures no longer sets a flag in location \$80 in the data segment. For further details, please refer to section 19.
- o In PolyPascal, a constant identifier used in a CODE statement does not always generate two bytes of code. For further details, please refer to section 20.
- o The internal data format of file variables has been changed. For further details, please refer to section 23.1.5.

### 3. Additional Standard Functions

---

PolyPascal-86 V3.10 features two new standard functions for accessing command line parameters. They are:

`argcnt` Returns the argument count of the command line, i.e. the number of parameters that followed the program name. The result is of type integer.

`argstr(n)` Returns a string containing the n'th command line argument. n must be an integer expression, and 1 denotes the first argument. If n is zero or greater than `argcnt`, `argstr` returns an empty string.

If, for instance, a program file called "DUMP.COMD" is executed using the command line:

```
DUMP CROSSREF.PAS CON:
```

Then `argcnt` returns 2, `argstr(1)` returns "CROSSREF.PAS", and `argstr(2)` returns "CON:". Note that only blanks and TABS are considered as parameter separators.

Although it is not stated in the manual, the RUN command allows you to specify a command line when you run a program in memory. All characters that follow the RUN command word are copied into the program's command line buffer.

#### 4. Overlay File Handling

---

PolyPascal V3.10 never leaves overlay files "open". If the code of an overlay procedure or function is not in memory when it is called, the associated overlay file is opened, the code is read into memory, and the file is closed. Thus, overlay files need never be closed explicitly, and therefore the "ovclose" standard procedure mentioned in the manual is not available.

#### 5. User Written Error Handlers

---

In PolyPascal-86 V3.10 you may write your own error handler, which is called in case of an I/O or execution error. The procedure must have the following header:

```
PROCEDURE error(errno,errofs: integer);
```

The names of the procedure and its parameters are unimportant, as long as it is a procedure with two value parameters of type integer.

The value passed in "errno" is the error type and number. The most significant byte, i.e. "hi(errno)", contains the error type, and the least significant byte, i.e. "lo(errno)", contains the error number (see Appendix F or G in the PolyPascal Programming Manual). The following error types are defined:

- 0 User interrupt (Ctrl-C).
- 1 I/O error.
- 2 Execution error.

In case of a user interrupt (Ctrl-C), the low byte of "errno" is always 1. "errofs" contains the offset address of the error.

To activate an error handler, assign its offset address to the standard variable "ehofs", i.e. "ehofs:=ofs(error)".

There are no limits to what an error handler may do. Typically it will close all open files, output an error message, and call the "halt" standard procedure to terminate the program. If an error handler returns, i.e. if it does not call "halt", or if an error occurs within an error handler, PolyPascal will itself output the error message and terminate the program.

#### 6. User Interrupts During Console I/O

---

In addition to the C compiler directive, PolyPascal-86 V3.10 now implements a predefined boolean variable called "cbreak", to which you may assign the values true or false. When "cbreak" is true, PolyPascal will check for Ctrl-S and Ctrl-C during console I/O. When "cbreak" is false, no checks are made. The C compiler directive controls the initial value of "cbreak" when a program is run. "C+" means true (on), and "C-" means false (off). The default setting is "C+".

#### 7. Concurrent CP/M notes

---

The CP/M-86 version of PolyPascal-86 runs under Concurrent CP/M with no changes.

When running under Concurrent CP/M, PolyPascal-86 will automatically detach the printer when a program terminates following a RUN command. Likewise, when the ↑K↑P editor command completes printing a block, it automatically detaches the printer.

PolyPascal-86 V3.10 provides a standard integer variable called "sysres". Whenever the run-time library performs a file oriented BDOS call (such as open file, close file, read random, etc.), the exit code returned in the AX register is stored in "sysres". This may at times be useful in determining the actual cause of an I/O error.

#### 8. Comments and Suggestions

---

If you have any comments to the PolyPascal package and/or its documentation, e.g. improvements you would like to see, or bugs you may have spotted, please don't hesitate to contact us. Our address is:

PolyData MicroCenter A/S  
Aaboulevarden 13