

PERQ Pascal Extensions  
Addendum

Miles Barel

Three Rivers Computer Corporation

November 21, 1980

Three Rivers Computer Corporation  
720 Gross Street  
Pittsburgh, PA 15224  
(412) 621-6250

This document is an addendum to the PERQ Pascal Extensions manual released on October 27, 1980.

The information in this document is subject to change without notice and should not be construed as a commitment by Three Rivers Computer Corporation. Three Rivers Computer Corporation assumes no responsibility for any errors that may appear in this document.

## Table of Contents

- 1. Identifiers
- 2. Files
- 3. Procedure/Function Parameters
  - 3.1 Parameter List
  - 3.2 Function Result Type
- 4. Command Line and Compiler Switches
  - 4.1 Command Line
  - 4.2 Compiler Switches
    - 4.2.1 File Inclusion
    - 4.2.2 Automatic RESET/REWRITE
- 5. Quirks and Other Oddities

## PERQ Pascal Extensions Addendum

### 1. Identifiers

PERQ Pascal permits the inclusion of the underscore character "" as a significant character in identifiers.

## 2. Files

PERQ Pascal permits the use of files as the type for arrays, pointers and record fields.

### 3. Procedure/Function Parameters

The following are additions to Section 9 of the Pascal Extensions Manual.

#### 3.1 Parameter List

PERQ Pascal permits, but does not require, the parameter list of procedures and functions which have been forward declared to be repeated at the site of the actual declaration. If given, the parameter list must match the previous declaration or a compilation error will occur. For redeclaration of function parameters, both the parameter list and the function type must be repeated.

#### 3.2 Function Result Type

PERQ Pascal functions may return any type, with the exception of type FILE.

#### 4. Command Line and Compiler Switches

The following are additions and corrections to Section 15 of the Pascal Extensions Manual.

##### 4.1 Command Line

The syntax for the compiler command line is:

```
[<OutFile>=<InFile>{/SW}
```

<InFile> is the source file to be compiled. If the file <InFile> does not exist, ".PAS" will automatically be concatenated onto <InFile> and a second attempt will be made to find <InFile>.

<OutFile> is the name to be given to the output of the compiler. ".SEG" will be concatenated onto the end of <OutFile> if it is not already present. If no <OutFile> is given, <InFile> will be used, replacing ".PAS", if present, with ".SEG".

##### 4.2 Compiler Switches

###### 4.2.1 File Inclusion

The PERQ Pascal compiler may be directed to include the contents of secondary source files in the compilation. The effect of using the file inclusion mechanism is identical to having the text of the secondary file(s) present in the primary source file (the primary source file is that file which the compiler was told to compile).

To include a secondary file, the following syntax is used:

```
{ $I FILENAME }
```

The characters between the "\$I" and the "}" are taken as the name of the file to be included (leading spaces and tabs are ignored). The comment must terminate at the end of the filename, hence no other options can follow the filename.

If the file FILENAME does not exist, ".PAS" will be concatenated onto the end of FILENAME, and a second attempt will be made to find the file.

The file inclusion mechanism may be used anywhere in a program or module, and the results will be as if the entire contents of the include file were contained in the primary source file (the file containing the include directive).

Note: There is no form of this switch for the command line, it may only be used in comment form within a program.

#### 4.2.6 Automatic RESET/REWRITE

The PERQ Pascal compiler will cause a RESET(INPUT) and REWRITE(OUTPUT) to be performed automatically. This may be disabled if desired with the use of the AUTO switch. The format for this switch is:

Default value: Automatic initialization enabled

\$A+ or /AUTO enables automatic initialization

\$A- or /NOAUTO disables automatic initialization

If "\$A" is not followed by a "+" or "-" then "+" is assumed.

If the comment form of this switch is used, it must precede the BEGIN of the main body of the program.

## 5. Quirks and Other Oddities

The following are descriptions of known quirks and problems with the PERQ Pascal compiler. It is our intension that these problems will be corrected in the near future.

FOR loops with an upper bound of 32,767 will never terminate.

The construct

```
WRITE(File,Character)
```

does not work when File is a VAR parameter. In this case, use

```
WRITE(File,Character:1)
```

The last line of any PROGRAM or MODULE must end with a carriage return, or an "Unexpected End of Input" error will occur.

NEW does not return nil if the space allocation requested cannot be performed.



## INDEX

( )ries entirely in upper case are reserved words or predeclared identifiers)

Automatic RESET/REWRITE . . . . .	5
Command Line . . . . .	4
Compiler Switches . . . . .	4
File Inclusion . . . . .	4
Files . . . . .	2
Function Result Type . . . . .	3
Functions . . . . .	3
Identifiers . . . . .	1
Include . . . . .	4
Parameter Lists . . . . .	3
Procedures . . . . .	3
Quirks . . . . .	6
Switches . . . . .	4