

Notes for TE/2 Version 1.23
April 15, 1993

Oberon Software
518 Blue Earth St.
Mankato, MN 56001-2142

Voice: 507/388-7001
BBS: 507/388-1154

Copyright 1991-93, Oberon Software, Mankato, MN - All Rights Reserved

The top portion of this document outlines the changes applied between TE/2 Version 1.20 and 1.23. The second portion of this document outlines changes made between TE/2 Version 1.10C and 1.20.

=====
Changes which effect both the Registered and Shareware Versions of TE/2
=====

New INI File Settings

Variable	Setting and Notes
SetVTDefAttr	TRUE or FALSE, default is FALSE. If set to TRUE, then the value of your TermAttr will be given to the VT100 handler to use as the default attribute. This replaces and overrides any setting for the "V" flag in the COMMPAK2FLAGS environment variable.
NoHPFSSalvage	TRUE or FALSE, default is FALSE. If set to TRUE, TE/2 will NOT use the HPFS-type file renaming scheme when a filename collision occurs on a download.
NoExtSalvage	TRUE or FALSE, default is FALSE. If set to TRUE TE/2 will use only one digit from 1 to 9 as the final character when attempting to resolve a file name collision on a FAT drive (or when NoHPFSSalvage is TRUE).
SalvageBrkCh	any number from 1 to 255, default is 59. Set this to the ASCII value of the character you wish to use as a separator character when resolving file name collisions on an HPFS drive. By default this is a semi-colon and renamed files have the form: filename;nn (where nn is a number from 1 to 999). Suggested settings for this would be 46 (a period), 32 (a space), 45 (a hyphen), or 95 (an underscore). Do NOT use any character which is not valid in an HPFS file name such as +, /, \, :, * or ?
CursorTop	These are the initial scan line setting for the cursor.

CursorBottom If they are left un-set, the cursor will not be changed. If one is set, they must both be set. They can be queried from the script language but not set, use the new SetCursorSz() function instead.

ExtendedFIFO This is not a new setting but it was only partially enabled in TE/2 1.20. The choices for this setting are now: TRUE, FALSE, or AUTO. The "fourth choice" is to leave reference to it OUT of TE2.INI. If it omitted from TE2.INI then the ambient state of the 16550 is preserved and used. If set to anything at all, the device is checked first for Extended Hardware Buffering support. If supported, then if ExtendedFIFO is set to FALSE, buffering is disabled; if set to TRUE, buffering is enabled with a Receive Trigger level of 8 and a Transmit Load Buffer Count of 16; if set to AUTO, then Auto-Buffering is enabled.

Problem Resolution and Other Enhancements

You may now specify the name of a Named Pipe, either local or remote, in the "Device" field in TE2.INI. If a pipe name is used then the various COM-specific functions, like setting the baud rate, parity, etc., are silently ignored.

An add-on product is available for using TE/2 with TCP/IP sockets and/or NETBIOS sessions. Please contact Oberon Software if interested.

The TE/2 intro screens are no longer garbaged if you start up in 132 column mode.

Added command line parameter "-?" which will display a list of TE/2's command line parameters, version information, and then exits.

```
=====
Changes which effect only the Registered Version of TE/2
=====
```

New Script Functions

```
integer CRC16(string strg, integer len)
integer CRC32(string strg, integer len)
```

These compute and return 16-bit and 32-bit CRCs for the the first 'len' bytes of the supplied string. If you set len to -1, then the entire string length is used.

```
integer SetFNKeyFile(string filename)
```

Allows a script to load a new function key definition file. Returns 0 on success, 1 if the file was not found or could not be opened for reading, and 2 if an error was encountered reading the file.

If the filename parameter is "" then the user will be queried for a file name. If the filename parameter contains path

information and/or a drive letter, then the name is accepted as is, otherwise the standard search algorithm for TE/2 support files is used (current directory, directory containing TE2.EXE, and then each directory in the OS/2 PATH statement).

string ReadScr(integer row, integer col, integer length)

Reads 'length' characters from the terminal screen at row 'row' and column 'col'. 'row' and 'col' are 0 based (upper left corner of screen is [0,0]). 'length' must be greater than zero and less than 256. If 'length' is zero or less, the empty string is returned. If 'length' is greater than 256, 256 is used. No error checking on 'row' and 'col' for validity is performed yet.

integer StrnStr(string s1, string s2, integer n)

Finds the n-th occurrence of s2 in s1. The call StrnStr(s1, s2, 1) would be identical to StrStr(s1, s2).

string UniqueFName(string template)

Creates a temporary file name based on 'template' which is a basename plus wildcard characters '?'. For example: C:\TEMP\FILE?????. The function does NOT open the file. Returns the empty string if no new names could be generated. Note: this function works exactly like the REXX/2 utility function SysTempFileName except that you cannot specify an alternate filler character.

integer CancelAllWatch()

Cancel all active watches. Returns the number of watchfor's which were canceled.

integer SetCursorSz(integer top, integer bottom)

Sets the size of the cursor. 'top' and 'bottom' are the top and bottom scan lines for the cursor, respectively. Returns TRUE on success and FALSE on failure. If successful, system variables 'CursorTop' and 'CursorBottom' are updated.

integer SetMark(string tag, special)

"tag" is a dialing directory tag, "special" is TRUE, FALSE, or TOGGLE. This function sets the mark for the queue dialer on the specified dialing directory entry. Returns the OLD state of the mark for that entry (TRUE or FALSE) or -1 if "tag" was not found.

integer TermWindow(integer Row, integer Col, integer nRows, integer nCols)

Resets the terminal window (a subset of the physical screen) to the size specified. Returns TRUE on success, FALSE on error. It's recommended that you perform a cls() before and after this call. Take great care with this one, it's a direct back door into the terminal emulation code.

New 'system', read-only script variables:

integer Emulation

Reports the current emulation mode:

- 0 -> TTY
- 1 -> ANSI-BBS
- 2 -> ANSI-TE/2
- 3 -> VT100
- 4 -> IBM 3101

integer Fullscreen

Set to TRUE (1) if TE/2 executing in a fullscreen session or FALSE (0) if in a window

integer ScreenRows

integer ScreenCols

These report the actual number of rows and columns on the physical screen.

string StartScr

Contains the name of the script specified on the TE/2 command line via "-m<scriptfile>". If none was specified, StartScr contains "".

integer REXXRC

Contains the return code from the last executed REXX script (0 if none have been executed). See notes on REXX scripting below.

Changes and corrections to Script Parser

Change to the way REXX script return codes are handled.

Previously you had to end your REXX script with "exit 1" (or other non-zero value) if you needed to return execution to a TE/2 language script. A return value of 0 would cause script execution to end. TE/2 will now continue executing the current script regardless of the REXX return code. A new, readonly script variable "REXXRC" is set to the return value of the last executed REXX script.

Example:

```
program
  message("This is a TE/2 script^M^J")
  run("rexxscr.cmd")
  message("REXX return code is %d^M^J", REXXRC)
end
```

Default ScriptPath

If the ScriptPath setting in TE2.INI was left NULL, it was erroneously defaulting the search path to the root directory of the current disk. TE/2 now defaults to the CURRENT directory as the documentation suggests.

Reentrancy and Watchfor()

Previously the script parser was not reentrant and because of

this Watchfor() statements could only be executed between the execution of individual lines of the script file. This severely limited the use of Watchfor() in conjunction with actions that should occur immediately (i.e., transmitting a response to a trigger string). This limitation has now been removed. Some of the new example scripts use this feature extensively, please refer to them for examples.

Download()

Because of changes in the TE/2 and Commpak/2 handling of input data streams in the 1.20 release, Download() stopped working correctly with respect to ZModem. This has been corrected.

Watchfor()

The number of available watchfor handles has been increased from 10 to 32

Watchfor(), OpenDialog(), and SaveScr()

These functions may now be called 'directly' from REXX scripts. Be sure to retrieve the returned handles immediately from the REXX 'RC' variable after calling one of these.

Transmit()

The transmit() function was erroneously limiting output to the first 64 characters of any given string and discarding the remainder. This has been corrected.

The transmit() function would behave unpredictably on strings containing one or more modem delay characters if local echo mode was not ON. This has been corrected.

Notes for TE/2 Version 1.20
June 15, 1992

Oberon Software
518 Blue Earth St.
Mankato, MN 56001-2142

Voice: 507/388-7001
BBS: 507/388-1154

=====
Changes which effect both the Registered and Shareware Versions of TE/2
=====

* New INI File variables

DirFile <filename> -- Names the default dialing directory file which TE/2 will load at startup.

FnkFile <filename> -- Names the default function key definition file which TE/2 will load at startup.

XexFile <filename> -- Names the default external programs file which TE/2 will load at startup.

VT100Prn <devname> -- Names the "printer device" to be used in conjunction with the new printer support in the VT100 emulation (described elsewhere in this document)

VT100Backspace -- Set to True or False, default is False. If "true", the VT100 emulation will interpret incoming backspace characters as "destructive" (That is, a backspace will be equivalent to the sequence backspace-space-backspace).

SwapBS4Del -- Set to True or False, default is False. If "true", TE/2 will transmit a DEL character (ASCII 127) from the backspace key instead of the default backspace character (ASCII 8). Note that this effects ONLY the backspace key itself and not CONTROL+H.

DialerSendInit -- Set to True or False, default is False. If "true" the TE/2 dialer will transmit the modem initialization string ("ModemInitStrg") to the modem at the beginning of each dialing session.

* New INI file keyword

INCLUDE <filename> -- allows you to segregate portions of your INI file settings into separate files. This feature is used in the default INI file for the new modem definition setup described elsewhere in this document.

* OnLine Colors Setup!

Atl-Y from within TE/2 will now execute TE2COLOR.EXE if possible.

TE2COLOR.EXE must be either

- 1) in the current directory,
- 2) in the same directory as TE2.EXE, or
- 3) somewhere on the PATH.

Your INI file must be located likewise (if you've changed directories since startup, this may cause a problem). Note that online color setup WILL NOT WORK if you have your color attributes in a file which has been "included" into your .INI file. If this is the case, you still can run TE2COLOR as a standalone, the command line syntax is "TE2COLOR filename".

* Modem setup files

A file named MODEMS.ZIP is included with this release of TE/2. It contains over 60 modem configuration files designed to be "included" into your TE2.INI for various modem types. To use these files, refer to the file MODEMS.DOC which is contained in MODEMS.ZIP.

* Alternate Keyboards

Alt-= brings up a menu of alternate keyboards for you to choose from. Support is provided for:

OS2You Mode
Doorway Mode (Standard)
Doorway Mode (Enhanced)
"Pure" Binary Mode for function keys

See TE2.DOC, the section on "Alternate Keyboards" for a description of each of these and further information.

* PASSWORD Field in Dialing Directory entries

You can assign a string to be associated with each entry into the dialing directory. The "Toggle" function in the dialing directory screen will allow you to view this setting. In the full featured version of TE/2, you may access this value via the "PASSWORD()" function (described elsewhere in this document).

* "Access Codes" in the dialing directory

This was previously a "registered version only" feature, it has now been included into the shareware release of TE/2. This feature allows you to enter up to ten strings for each dialing directory that may be substituted as "macros" into the phone numbers of directory entries.

* New "Filename Mangling" logic on download filename collisions

The old filename resolution logic, involving dollar sign characters, has been replaced. TE/2 will resolve filename collisions differently depending on whether the target disk is formatted with HPFS or not:

On HPFS, ";xx", where "xx" is a number starting at 1, will be appended to the existing file's name.

On FAT, the last two characters of the existing file's name will be replaced with a two digit number beginning at 01.

* COMMPAK2FLAGS Environment Variable

This environment variable has been instituted as a means of passing certain information into Commpak2.Dll to alter its behavior. At this time, there is one valid setting for COMMPAK2FLAGS:

"set COMMPAK2FLAGS=X:Y;" or "set COMMPAK2FLAGS=X:N;"

These settings will turn the 16550 Extended Hardware Buffering check ON and OFF respectively. The default is on. Although the com device driver should only return an error in response to a check for the 16550 chip, on some machines it causes a fatal trap! If you experience this problem just as a file transfer begins, you should set COMMPAK2FLAGS=X:N; in your CONFIG.SYS.

* New INI Setting: ExtendedFIFO True/False

TE/2 will not alter the BUFFERS setting for the com port upon startup unless this setting appears in the INI file (that is,

TE/2 will use whatever is default or has been set via the OS/2 MODE command). If ExtendedFIFO is set to "TRUE", TE/2 will attempt to set AUTO buffering, if it is set to "FALSE" it will attempt turn off the 16550 FIFO buffer. Do not use this setting unless you deem it absolutely necessary. Note also that the file transfer protocols will, by default, attempt to set AUTO buffering during file transfer. If this causes a problem on your machine, you can turn this off via the COMMPAK2FLAGS environment variable described elsewhere in this document.

* VT100 Printing

If the new INI file setting "VT100Prn" is set to a valid device or file name before entering VT100 mode, certain VT220 printing features will be enabled. This will allow the host program to direct the VT100 to perform various print functions:

Auto Print Mode ON and OFF (^[[?5i and ^[[?4i)

All following display lines print when you move the cursor off the line using a LF, FF, VT, or auto wrap. The printed line ends with a CR and the character which moved the cursor off the previous line. Auto line wrap line "ends" with a LF.

Printer Controller Mode ON and OFF (^[[5i and ^[[4i)

The terminal sends received characters to the printer without displaying them on the screen. All characters and character sequences, except NULL, XON, XOFF, ^[[5i, and ^[[4i, are sent to the printer. Printer controller mode has a higher priority than Auto print mode. It can be selected during auto print mode.

Print Screen (^[[i or ^[[0i)

Print the screen or the current scrolling region (depending on the DECPEX setting). Each line ends with a CR+LF pair and the operation is completed with a FF if DECPFF mode is selected.

DSR response to ^[[?15n

The VT100 will respond to the hosts query for terminal printer information with ^[[?10n if a device has been named via VT100Prn (implying the printer is "ready") or with ^[[?13n if no device is named (printer NOT "ready").

Print Mode Selection Supported

^[[?18h -- DECPFF. Print FF after print screen.
^[[?18l -- DECPFF. Don't print FF after print screen.
^[[?19h -- DECPEX. ^[[i prints full screen.
^[[?19l -- DECPEX. ^[[i prints active scroll region.

Print cursor line NOT supported

^[[1i, "Print current cursor line" is NOT supported by the VT100 emulation as it would not be possible to produce reasonable printer output from this function given the intervention of the OS/2 spooler.

* Character translation of "[" characters filtered

For certain European character setups, the "[" needs to be

translated but it still needs to be accessed "as is" when it appears in an ANSI or VT100 control sequence! When applying character translation from the XLat table, TE/2 now checks for the special case of '[' immediately following an ESCape character and does not translate it in this situation. Any arbitrary number of control characters may appear between the ESC and the '[' and it will still be considered "immediately after".

- * High-bit characters allowed in input fields.
European users can now enter their names properly in the dialing directory and other input fields!
- * TE/2 now no longer reports negative numbers for baud rates of 38400 or greater.
- * The default baud rate for a new entry into the TE/2 dialing directory is now set to whatever was specified in your "Baud" statement in TE2.INI.
- * TRAP 0008 problem for TE/2 with shared modem over LAN addressed.
If you have experienced this problem, place the line:

NoOvIO TRUE

anywhere in your TE2.INI. This should correct the problem at the expense of a small amount of throughput. If the problem persists, please contact Oberon Software.

- * Cursor problem fixed in input fields when screen in 25 line mode. Previously the cursor would not display correctly in input fields unless the screen was in 43 or 50 line mode.
- * Long "CONNECT" strings no longer overwrite the borders of the dialer dialog box.

=====
Changes which effect only the Registered Version of TE/2
=====

- * REXX/2 Interface for script writers!
You now have the option of writing your TE/2 scripts in either the TE/2 "Original Syntax" Script Language or in the REXX/2 Procedure language! See the notes in SCRIPT.DOC regarding REXX scripts for further information.
- * New Script Language Functions
NOTE: In each of the following functions which use "tag" as an argument, the string "tag" may be either a value to search the directory file's "Tag" field for or it may be "" in which case it is interpreted as "the most recently dialed entry".

Password(tag) Returns the "Password" field for the

	entry or "" on error.
DirName(tag)	Returns the "Name" field for the entry or "" on error.
DirPhone(tag)	Returns the "Number" field for the entry or "" on error.
DirScript(tag)	Returns the "Script" field for the entry or "" on error.
DirTag(tag)	Returns the "Tag" field for the entry or "" on error.
SaveDirFile(filename)	Saves the current directory as filename, if filename is "" saves the current directory without changing the name. Returns 0 on success, non zero otherwise.
SetTag(tag, tag2)	Sets the "Tag" field for the entry, returns 1 on success, 0 on failure.
SetName(tag, name)	Sets the "Name" field for the entry, returns 1 on success, 0 on failure.
SetPassword(tag, password)	Sets the "Password" field for the entry, returns 1 on success, 0 on failure.
SetPhone(tag, phone)	Sets the "Number" field for the entry, returns 1 on success, 0 on failure.
SetScript(tag, script)	Sets the "Script" field for the entry, returns 1 on success, 0 on failure.
StrEval(strg)	Returns the value of its argument. This function appears as an aid to the REXX language script writer.
SetXlatTable(fname, ioflag)	Reads a new character translation table from the indicated file. If ioflag is non-zero, it will initialize the OUTPUT translation, if it is zero it will initialize INPUT translation. If fname is "" the corresponding character translation is turned OFF.
SetVT100BkSp(flag)	Sets the VT100 backspace mode (see INI variable VT100Backspace) to ON or OFF depending on the Boolean "flag". Returns the previous state of the VT100 backspace mode.

* File Transfer Progress Display revised and cleaned up for CIS B+ The somewhat confusing "Bytes Transferred" and "Time Remaining" statistics for CIS B+ transfers have been reworked and cleaned up. TE/2 will now "ring the bell" after a CIS B+ file transfer if you have alarms enabled also!

