TEXAS v1.0a

Shih-Ping Chan
Department of Mathematics
National University of Singapore

March 1, 1992

Contents

1 Overview

`TEXAS` is a 32-bit protected mode version of big TEX 3.14 compiled for MS-DOS 386/486 computers. It uses a royalty-free distribution version of Rational Systems Inc. DOS Extender which is XMS, VCPI, and DPMI compatible. `TEXAS.ZIP` is an executable distribution only. It is intended to complement any existing 16-bit TEX setup or as an alternative to 32-bit TEXs like PCTEX/386 or emTEX/386.

`TEXAS` is completely free and you are welcome to use it in any way you see fit (producing beautiful documents of course).

2 Manifest

1. `README` — stop press information
2. `TEXAS.EXE` — the TEX executable
3. `TEX.POO` — the pool file used by `TEXAS.EXE`

4. `DOS4GW.EXE` — the royalty-free distribution version of RSI's DOS Extender
5. `RMINFO.EXE` — two programs from RSI to test protected mode
6. `PMINFO.EXE` — operation (distribution permitted by RSI)
7. `TEXAS.DVI` — `TEXAS` documentation
8. `TEXAS.PS` — `TEXAS` documentation in PostScript format

3 Changes from v1.0

(a) Minor optimisations have been made so that `TEXAS 1.0a` runs 5–10% faster than `TEXAS 1.0`.

(b) `TEXAS` now responds to `<ctrl>-C` and `<ctrl>-<break>`.

(c) If the `TEXINPUTS` environment variable is set, then `TEXAS 1.0` does not search for files in the current directory and files given by relative path names (other than files starting with "`./`" and "`../`") unless the current directory is explicitly included in `TEXINPUTS`. This has now been changed. `TEXAS 1.0a` now searches the current directory for files (including relative path names) before trying the paths listed in the `TEXINPUTS` variable. This also applies to `TEXPOOL`, `TEXFONTS`, and `TEXFORMATS`.

4 Getting Started

(a) Copy `TEXAS.EXE` and `DOS4GW.EXE` to a directory which is on your DOS path. `TEXAS.EXE` uses `DOS4GW.EXE` and searches for it using the `PATH` variable.

(b) Create a new directory to put `TEX.POO` and the format files. Remember that `*.FMT` files and `TEX.POO` are not interchangeable between different implementations of TeX.

(c) Set the following environment variables so that `TEXAS` can find all its auxiliary files. The value of the variables can be a list of directories separated by semi-colons, exactly like the DOS `PATH` variable.

```
set TEXPOOL=<directory list to find TEX.POO>
set TEXFORMATS=<directory list for *.FMT files>
set TEXFONTS=<directory list for *.TFM files>
set TEXINPUTS=<directory list for input files>
```

E.g.,

```
set TEXPOOL=c:\tex\formats
set TEXFORMATS=c:\tex\formats
set TEXFONTS=c:\tex\fonts
set TEXINPUTS=c:\tex\inputs;c:\tex\myinputs
```

(d) `TEXAS` needs about 5 MB of memory (conventional and extended). If you do not have enough memory installed `TEXAS` will use your hard disk as virtual memory but you will first have to set additional environment. Refer to section 5 if you do not have 5 MB of RAM.

(e) Create your format files and put them into the `TEXFORMATS` directory. `TEXAS -i` starts `TEXAS.EXE` up in `INITEX` mode which is used for creating format files.

E.g., the command `TEXAS -i plain` will create the file `PLAIN.FMT` (after typing "`\dump`" at the * prompt). You should now move this file to the `TEXFORMATS` directory.

NOTES

- It is not necessary to include the current directory "`.`" in the environment variables. When looking for file `myfile` (which may be a relative path name) `TEXAS` will always try `./myfile` before using the directories listed in the appropriate environment variable.

5 Virtual Memory

This information is for users of `TEXAS` who have less than 4 MB of extended memory available. It documents environment variable settings which enable the DOS Extender to control virtual memory. This information is taken from the manual 'DOS Programming Guide for 32-bit Applications' by G. Coschl and J. B. Schueler of WATCOM. (I'm assuming they won't mind me reproducing the relevant information here — after all the DOS Extender is meant for distribution and it is not much use without this information.)

`DOS4GW.EXE` is a 32-bit Rational Systems' DOS Extender that is provided with WATCOM C/386. It is a subset of Rational Systems' DOS/4G product and only works with WATCOM C/386 programs. The virtual memory manager is restricted to 32 MB of memory.

5.1 How to use virtual memory

To enable the virtual memory manager (VMM) just set the DOS4GVM environment variable.

Syntax: `set DOS4GVM= [option[#value]] [option[#value]]` ...

Option	**Purpose [default value in K]**
`MINMEM`	Minimum amount of RAM managed by VMM [512]
`MAXMEM`	Maximum amount of RAM managed by VMM [4096]
`SWAPNAME`	Swap file name [`\DOS4GVM.SWP`]
`DELETESWAP`	Delete the swap file on program exit
`VIRTUALSIZE`	Size of virtual memory space [16384]

E.g.,

```
set DOS4GVM=deleteswap virtualsize#4096
```

NOTES

- `VIRTUALSIZE` = Swap file + allocated memory
- `TEXAS` only needs a virtual size of about 4 MB

5.2 Alternative method

Create a file with extension "`.VMC`" e.g. `MYFILE.VMC` and use the file name (preceded by `@`) as an option to DOS4GVM.

```
set DOS4GVM=@MYFILE.VMC
```

Put your settings in `MYFILE.VMC` as follows (reproduced from the WATCOM manual):

```
=========A Sample VMC file=========================================

!Sample .VMC file
!This file shows the default parameter values
minmem = 512 At least 512 bytes of RAM is required
maxmem = 4096 Uses no more than 4MB of RAM
virtualsize = 16384 Swap file plus allocated memory is 16MB
!To delete the swap file automatically when the program exits, add
!deleteswap
!To store the swap file in a directory called swapfile add
!swapname = c:\swapfile\dos4gvm.swp

=========End of Sample VMC file====================================
```

NOTES

- Comments are permitted. Comments on lines by themselves are preceded by an exclamation point (!). Comments that follow option settings are preceded by white space.
- Do not insert blank lines; processing stops at the first blank line.

5.3 Sample Setting

The following is a sample setting which is appropriate for `TEXAS`,

```
set DOS4GVM=swapname#c:\tmp\dos4gvm.swp virtualsize#4096
```

but you may need to fiddle with `virtualsize`.

6 Command line options

Usage: `TEXAS [-iv?]  [&format_file] [tex_command_line]`

`-i` starts `TEXAS` in `INITEX` mode which is used for creating format files. To use another format file besides the default `PLAIN.FMT` use the & method.

`-v` prints the current version of `TEXAS.EXE`.

`-?` will print a help message.

E.g., `TEXAS &lplain myfile` starts `TEXAS` with `LPLAIN.FMT` instead of the default `PLAIN.FMT`.

7 Technical Notes

(a) `TEXAS` is an implementation of big TEX 3.14. Main memory size is 256K, font memory size is 100000, and the pool size is 72000. `TEXAS` also incorporates the changes in the UNIX TEX C distribution which allows large hyphenation patterns of European languages; hence the banner line reads "`...C Version 3.14t3`". In fact `TEXAS` was compiled from the UNIX C distribution. Modifications to the source code were made to combine `INITEX` and `VIRTEX` into a single program and to account for the differences between UNIX and DOS file handling. `TEXAS` was compiled using WATCOM C/386.

(b) `TEXAS` has passed the `TRIP` test for TEX 3.0. The `TRIPIN.LOG, TRIP.FOT`, and `TRIP.LOG` files are available on request. Source code and diffs from the UNIX TEX C distribution are also available. The source code was also compiled successfully with the `DJGPP` port of the GNU C compiler.

(c) The DOS Extender is both VCPI and DPMI compliant. `TEXAS` has been tested in DESQview and Windows 3.0 (386 Enhanced mode) DOS boxes. I'd be interested to find out if it works in the OS/2 v2.0 DOS box.

(d) `PMINFO.EXE` measures the performance of protected/real-mode switching and extended memory (quoted from the WATCOM C/386 manual).

`RMINFO.EXE` supplies configuration information and the basis for real/ protected-mode switching in your machine (quoted from the WATCOM C/386 manual).

8 Bugs

(a) `TEXAS` is a TEX 3.0 so it should be able to handle 8-bit input. However it opens all input files in text mode so it might well choke if your input files contain `^Z` or ASCII 26. An example of such a file is `TEXINFO.TEX` that comes with some of the GNU manuals. A work around is to replace all occurrences of `^Z` with the string "`^^Z`".

(b) Full path names should follow the UNIX convention of having '`/`' as the directory separator. However both '`/`' and '`\`' can be used in the setting of environment variables. E.g.,

```
\input c:/tex/inputs/plain.tex
```

`TEXAS` will try, however, to display file names using UNIX conventions. This is a change over `TEXAS 1.0`.

(c) `TEXAS` has problems finding files under PC-DOS 4.01. There is no workaround yet.

9 Acknowledgments

(a) `DOS4GW.EXE`, `RMINFO.EXE`, and `PMINFO.EXE` are solely products of Rational Systems Inc.. Their distribution is permitted and encouraged by WATCOM and RSI. I have had nothing whatsoever to do with these programs.

(b) The UNIX TeX C distribution (without which `TEXAS` would have been impossible) is the work of Tim Morgan, Tomas Rokicki, Pierre Mackay, and Karl Berry (and I've probably left out a few names here). It is available for anonymous ftp on `byron.u.washington.edu`.

(c) Thanks to Helmer Aslaksen, James McKinnon, and Eberhard Mattes for bug reports.

10 Change History

1. 1.0a — internal changes; `-v?` switches added; documentation available in TeX form; removed bug in interrupt handling; search current directory/relative path names first.

2. 1.0 — initial release

11 Authorship

`TEXAS` was compiled by

Shih-Ping Chan ("I do not speak for, or represent, my employer")
Department of Mathematics
National University of Singapore
10 Kent Ridge Crescent
Singapore 0511
Republic of Singapore

All bug reports and comments for improvement are welcome and may be sent to `matcsp@nuscc.nus.sg`.