TECO-64 – Command-Line Options

Overview

This document describes the command-line options that can be used to start TECO-64. Generally, only one file can be edited at a time, although it is possible to read one file and write another. Also, as noted in other documentation, once TECO has started, it is possible to switch between primary and secondary input and output streams.

As TECO processes the options it finds, it builds a command string that is processed when initialization has completed. However, this command string is not created until all options have been processed, and it therefore does not matter in what order they occur.

In addition to the options described below, there are also several optional environment variables that may be used in processing these options, or which affect TECO's behavior.

TECO_INIT	Defines the name of an initialization file which is immediately executed. This is often used to set desired features, such as enabling windowing mode on startup.
TECO_MEMORY	This is used by TECO to determine the name of the last file edited. With some operating systems, this environment variable will contain that name directly; with others (such as Linux, Windows, and MacOS), it contains the name of a file that in turn specifies the last file edited.
TECO_PROMPT	This contains a string used for TECO's prompt (instead of the standard asterisk prompt).

Basic Invocation

`teco foo`

Open *foo* for input and output (using *EB* command), then read in first page.

`teco`

Open last file edited, by reading its name from the file pointed to by the *TECO_MEMORY* environment variable. If that variable is not defined, TECO will start without opening any file.

Options

`-Ann, --argument=nn`

Store *nn* in Q-register A before starting TECO (using *UA* command). Normally used in conjunction with the *–execute* option.

`-B`*`string`*`, --buffer=`*`string`*

    Insert *string* into text buffer before TECO starts (using *I* command). Normally used in conjunction with the *–execute* option.

`-C, --create` (default)

    If *foo* does not exist, then create it (using *EW* command). This is effectively a *make* command with other versions of TECO.

`-c, --nocreate`

    If *foo* does not exist, then issue an error and exit.

`-E`*`mungfile`*`, --execute=`*`mungfile`*

    Execute *mungfile* as an indirect command file (using *EI* command). This is effectively a *mung* command with other versions of TECO. If file does not exist, TECO will try opening *mungfile.tec*.

    If *mungfile* is in single or double quotes, then that is treated as a command string instead of a file name.

`-I`*`initfile`*`, --initialize=`*`initfile`*

    Open *initfile* as an indirect command file (using *EI* command). If file does not exist, TECO will try opening *initfile.tec.*

    If *initfile* is in single or double quotes, then that is treated as a command string instead of a file name.

`-I, --initialize (default)`

    Use the *TECO_INIT* environment variable to find the name of the initialization file. If *TECO_INIT* is not defined, then no initialization occurs before TECO starts.

    If the value of *TECO_INIT* is in single or double quotes, then that is treated as a command string instead of a file name.

`-i, --noinitialize`

    Do not use any initialization file, and do not check *TECO_INIT.*

`-L`*`logfile`*`, --log=`*`logfile`*

    Open *logfile* as a log file for TECO input and/or output (a flag variable controls whether either or both are logged).

`-M, --memory` (default)

    If no file is specified, then use the *TECO_MEMORY* environment to find the name of a file which contains the name of the last file edited.

`-m, --nomemory`

> Do not use *TECO_MEMORY* to find the name of the last edited file.

`-O`*baz*`, --output=`*baz*

> Open *foo* for input (using *ER* command), and open *baz* for output (using *EW* command), then read in first page.

`-o, --nooutput` (default)

> Use the same name for the output file as the input file (unless –readonly is specified).

`-R, --readonly`

> Open *foo* for inspection (using *ER* command), and read in first page.

`-r, --noreadonly` (default)

> Open *foo* for input and output (using *EB* command).

`-S`*nn*`, --scroll=`*nn*

> Start TECO in split-screen window mode, with *nn* lines for a scrolling region (using *1W* and *nn,7:W* commands).

`-W, --window`

> Start TECO in window mode (using *1W* command).

`-X, --exit`

> Used with -E to exit from TECO once the indirect command file has been processed.

`-Z, --zero`

> Start TECO with strict enforcement of command syntax (using *-1EW* command).