

NAME

DVIasm – a TeX utility program for editing DVI files directly

SYNOPSIS

dviasm [*<options>*] *<dvi_file>*|*<dvi_dump_file>*

DESCRIPTION

DVIasm is a TeX utility program which is designed for editing DeVice-Independent (DVI) files directly. It consists of a single Python script, `dviasm.py`, in a human readable text format. It runs on any platform in which Python 3 is installed.

Features of **DVIasm** includes:

- 1) Disassemble a DVI file (or XeTeX XDV file) into a human-readable text file, the contents of which are easy to modify.
- 2) Assembles the output text file back to the binary format.

There are several alternatives for dumping and editing DVI files:

- 1) **DVItyp**e developed by Donald E. Knuth supports one-way conversion from DVI to a text format. There are also some derivatives: **pDVItyp**e for pTeX, **upDVItyp**e for upTeX, **ODVItyp**e for Omega.
- 2) The programs **dv2dt** and **dt2dv** can be used in pairs to allow two-way conversion between DVI and the DVI Text Language (DTL).
- 3) The program **dvispc** (part of DVIOUT previewer) also has an ability to convert between DVI and text.

Among those, **DVIasm** is designed to allow users to edit DVI files easily, for example by unifying “right1”–“right4” to a single command “right” which can be used regardless of the amount of move.

OPTIONS

- version** Show program’s version number and exit.
- h, --help** Show this help message and exit.
- u <STR>, --unit=<STR>**
Set unit [default: ‘pt’]. Allowed values are: ‘sp’, ‘pt’, ‘bp’, ‘mm’, ‘cm’ and ‘in’.
- o <FILE>, --output=<FILE>**
Set filename for output instead of stdout.
- e <STR>, --encoding=<STR>**
Set encoding for input/output of dumped text [default: ‘utf8’]. Allowed values are: ‘ascii’, ‘latin1’, ‘utf8’, ‘sjis’ and ‘eucjp’. When used with the option **-p** (or **--ptex**), allowed values are only ‘utf8’, ‘sjis’ and ‘eucjp’.
NOTE: This feature does not support dumping through stdout.
- x <STR>, --xxx-encoding=<STR>**
Set encoding for interpreting ‘xxx:’ strings [default: ‘none’]. Allowed values are: ‘none’, ‘utf8’, ‘sjis’ and ‘eucjp’.
This option allows users to specify the correct encoding for dumping/compiling the contents of \special.

Typical usages are as follows:

- 1) By default, the contents of `\special` are dumped/compiled as byte-to-byte escape sequences in ‘`\x..`’ format. This would suffice when you don’t need to read/edit the contents of `\special`.
- 3) For XeTeX and upTeX users, ‘`-x utf8`’ would be useful since these engines use UTF-8 to encode strings inside `\special` commands.
- 2) For pTeX users with ISO-2022-JP-encoded DVI, it would be helpful to select the appropriate option ‘`-x eucjp`’ (for Unix) or ‘`-x sjis`’ (for Windows) to be consistent with the internal Kanji encoding used by pTeX engine.

-t *<INT>*, **--tabsize=***<INT>*

Set tab size for push/pop [default: 2].

-p, **--ptex** Dump and compile ISO-2022-JP-encoded DVI for Japanese pTeX. This option is required to decode or encode Japanese characters being typeset (using ‘`set2`’ or ‘`put2`’) properly.

-s *<STR>*, **--subfont=***<STR>*

Specify the list of fonts with UCS2 subfont scheme in the comma-separated format. If *<STR>* is empty, the pre-defined internal subfont list is disabled. This option would be useful for example when using CJK package.

EXAMPLE

Documentation of **DVIasm** by the original author Jin-Hwan Cho is published as:

<http://tug.org/TUGboat/Articles/tb28-2/tb89cho.pdf> (TUGboat)

<http://ajt.ktug.kr/assets/2008/5/1/0201cho.pdf> (Asian Journal of TeX)

AUTHORS

The first version was written in Python 2 by Jin-Hwan Cho in 2007. Khaled Hosny started to support dumping/compiling XeTeX XDV format in 2014. Arthur Reutenauer ported the script to Python 3 in 2019. Current maintainer of **DVIasm** is Hironobu Yamashita, since 2020.

BUG REPORT

Please visit the GitHub repository <https://github.com/aminophen/dviasm>.

SEE ALSO

dvitype(1), **dv2dt(1)**, **dt2dv(1)**, **dvispc(1)**.