NAME
         sane−mustek_pp – SANE backend for Mustek parallel port flatbed scanners

DESCRIPTION
         The **sane−mustek_pp** library implements a SANE (Scanner Access Now Easy) backend that provides access to Mustek parallel port flatbed scanners and OEM versions.

         There are 2 classes of Mustek parallel port scanners: regular **CCD** (cold cathode device) scanners and **CIS** (contact image sensor) scanners.

         The current version of this backend supports both CCD type scanners and CIS type scanners.

         The following scanners might work with this backend:

CCD scanners

Model:	ASIC ID:	CCD Type:	works:
SE 6000 P	1013	00	yes
SM 4800 P	1013/1015	04/01	yes
SE 1200 ED Plus	1015	01	no
SM 1200 ED Plus	1015	01	no
SE 12000 P	1505	05	no
600 III EP Plus	1013/1015	00/01	yes
SE 600 SEP	1013	??	yes
600 II EP	????	??	no
MD9848	1015	00	yes
Gallery 4800	????	??	yes
Viviscan Compact II	1013	00	yes

CIS scanners

Model:	ASIC ID:	works:
Mustek 600 CP & 96 CP	1015	yes (*)
Mustek 1200 CP	1015	yes
Mustek 1200 CP+	1015	yes

OEM versions	Original	works
Medion/LifeTec/Tevion		
MD/LT 9350/9351	1200 CP	yes
MD/LT 9850/9851	1200 CP	maybe (**)
MD/LT 9858	1200 CP	probably
MD/LT 9890/9891	1200 CP	yes
Targa		
Funline TS12EP	1200 CP	yes
Funline TS6EP	600 CP	yes
Trust		
Easy Connect 9600+	600 CP	yes
Cybercom		
9352	1200 CP	yes (***)

         (*)   Calibration problems existed with earlier version of this driver. They seem to be solved now.

(**)   Problems have been reported in the past for the MD/LT9850 type (striped scans, head moving in wrong direction at some resolutions). It is not known whether the current version of the driver still has these problems.

IF YOU HEAR LOUD CLICKING NOISES, IMMEDIATELY UNPLUG THE SCANNER ! (This holds for any type of scanner).

(***) Possibly, the engine_delay parameter has to be set to 1 ms for accurate engine movements.

Please note that this backend is still under construction. Certain models are currently not supported and some may never be because the communication protocol is still unknown (eg., SE 12000 P).

Some scanners work faster when **EPP/ECP** is enabled in the BIOS. EPP mode however may lead to hard-locks on some Linux systems. If that is the case for you, you can either disable ECP/EPP in your BIOS or disable it in the backend itself (see GLOBAL OPTIONS).

Note that the backend needs to run as root or has to have appropriate access rights to */dev/parport** if libieee1284 support is compiled in. To allow user access to the scanner run the backend through the network interface (See **saned**(8) and **sane−net**(5)). Note also that the backend *does not* support *parport sharing*, i.e. if you try printing while scanning, your computer may crash. To enable parport sharing, you have to enable libieee1284 at compile time. This backend also conflicts with the **sane−musteka4s2**(5) backend. You can only enable one of them in your *dll.conf*. However, you have to enable the backend explicitly in your *dll.conf*, just remove the hash mark in the line "mustek_pp".

DEVICE DEFINITION

This backend allows multiple devices being defined and configured via the *mustek_pp.conf* file (even simultaneously, provided that they are connected to different parallel ports). Please make sure to edit this file **before** you use the backend.

A device can be defined as follows:

> *scanner <name> <port name> <driver>*

where

<name> is an arbitrary name for the device, optionally enclosed by double quotes, for instance "LifeTec 9350".

<port name> is the name of the parallel port to which the device is connected. In case libieee1284 is used for communication with the port *(default setup)*, valid port names are **parport0**, **parport1**, and **parport2**.

In case the backend is configured for raw IO *(old setup)*, port addresses have to be used instead of port names: **0x378**, **0x278**, or **0x3BC**. The mapping of parallel ports (lp0, lp1, and lp2) to these addresses can be different for different Linux kernel versions. For instance, if you are using a Kernel 2.2.x or better and you have only one parallel port, this port is named lp0 regardless of the base address. However, this backend requires the base address of your port. If you are not sure which port your scanner is connected to, have a look at your */etc/conf.modules*, */etc/modules.conf* and/or */proc/ioports*.

If you are unsure which port to use, you can use the magic value ***** to probe for your scanner.

<driver> is the driver to use for this device. Currently available drivers are:

> **cis600**   : for 600 CP, 96 CP & OEM versions
> **cis1200** : for 1200 CP & OEM versions
> **cis1200+** : for 1200 CP+ & OEM versions
> **ccd300**   : for 600 IIIE P & OEM version
>
> **Choosing the wrong driver can damage your scanner!**
> Especially, using the 1200CP settings on a 600CP can be harmful. If the scanner starts making a loud noise, turn it off immediately !!!

Using the cis600 driver on a 1200CP or a 1200CP+ is probably not dangerous. The cis1200+ driver also works for the 1200CP, and using the cis1200 driver on a 1200CP+ will typically result in scans that cover

only half of the width of the scan area (also not dangerous).

If unsure about the exact model of your OEM version, check the optical resolution in the manual or on the box: the 600CP has a maximum optical resolution of 300x600 DPI, whereas the 1200CP and 1200CP+ have a maximum optical resolution of 600x1200 DPI.

Examples:

        scanner "LifeTec 9350" 0x378 cis1200

        scanner Mustek_600CP 0x378 cis600

        scanner Mustek_600IIIEP * ccd300

If in doubt which port you have to use, or whether your scanner is detected at all, you can use *sane−find−scanner −p* to probe all configured ports.

CONFIGURATION

The contents of the *mustek_pp.conf* file is a list of device definitions and device options that correspond to Mustek scanners. Empty lines and lines starting with a hash mark (#) are ignored. Options have the following format:

        option <name> [<value>]

Depending on the nature of the option, a value may or may not be present. Options always apply to the scanner definition that precedes them. There are no global options. Options are also driver-specific: not all drivers support all possible options.

Common options
bw <value>

        Black/white discrimination value to be used during lineart scanning. Pixel values below this value are assumed to be black, values above are assumed to be white.
        Default value: 127
        Minimum: 0
        Maximum: 255

        Example: option bw 150

CIS driver options
top_adjust <value>

        Vertical adjustment of the origin, expressed in millimeter (floating point). This option can be used to calibrate the position of the origin, within certain limits. Note that CIS scanners are probably temperature sensitive, and that a certain inaccuracy may be hard to avoid. Differences in offset between runs in the order of 1 to 2 mm are not unusual.
        Default value: 0.0
        Minimum: −5.0
        Maximum: 5.0

        Example: option top_adjust −2.5

slow_skip

        Turns fast skipping to the start of the scan region off. When the region to scan does not start at the origin, the driver will try to move the scanhead to the start of the scan area at the fastest possible speed. On some models, this may not work, resulting in large inaccuracies (up to centimeters). By setting this option, the driver is forced to use normal speed during skipping, which can circumvent the accuracy problems. Currently, there are no models for which these inaccuracy problems are known to occur.

By default, fast skipping is used.

Example: option slow_skip

engine_delay <value>

Under normal circumstances, it is sufficient for the driver to wait for the scanner signaling that the engine is stable, before a new engine command can be transmitted. In rare cases, certain scanners and/or parallel port chipsets appear to prevent reliable detection of the engine state. As a result, engine commands are transmitted too soon and the movement of the scanner head becomes unreliable. Inaccuracies ranging up to 10 cm over the whole vertical scan range have been reported. To work around this problem, the engine_delay option can be set. If it is set, the driver waits an additional amount of time after every engine command, equal to the engine_delay parameter, expressed in milliseconds. It practice an engine_delay of 1 ms is usually sufficient. The maximum delay is 100 ms.

Note that every additional ms of delay can add up to 14 seconds to the total scanning time (highest resolution), so an as small as possible value is preferred.

Default value: 0
Minimum: 0
Maximum: 100

Example: option engine_delay 1

CCD driver options

top <value>

Number of scanlines to skip to the start of the scan area. The number can be any positive integer. Values known to me are 47 and 56.

Default value: 47
Minimum: 0
Maximum: none

Example: option top 56

waitbank <value>

The number of usecs to wait for a bank change. You should not touch this value actually. May be any positive integer

Default value: 700
Minimum: 0
Maximum: none

Example: option waitbank 700

A sample configuration file is shown below:

```
#
# LifeTec/Medion 9350 on port 0x378
#
scanner "LifeTec 9350" 0x378 cis1200

# Some calibration options (examples!).
option bw 127
option top_skip -0.8
```

```
#
# A Mustek 600CP on port 0x3BC
#
scanner "Mustek 600CP" 0x3BC cis600

# Some calibration options (examples!).
option bw 120
option top_skip 1.2

#
# A Mustek 1200CP+ on port 0x278
#
scanner "Mustek 1200CP plus" 0x278 cis1200+

# Some calibration options (examples!).
option bw 130
option top_skip 0.2

#
# A Mustek 600 III EPP on port parport0
#
scanner "Mustek 600 III EPP" parport0 ccd300

# Some calibration options (examples!).
option bw 130
option top 56
```

GLOBAL OPTIONS

You can control the overall behaviour of the **sane-stek_pp** backend by global options which precede any scanner definition in the *mustek_pp.conf* file.

Currently, there is only one global option:

Global options
no_epp

> Disable parallel port mode EPP: works around a known bug in the Linux parport code. Enable this option, if the backend hangs when trying to access the parallel port in EPP mode.

> Default value: use EPP

> Example: option no_epp

FILES

/etc/sane.d/mustek_pp.conf
> The backend configuration file (see also description of **SANE_CONFIG_DIR** below).

/usr/lib/x86_64-linux-gnu/sane/libsane−mustek_pp.a
> The static library implementing this backend.

/usr/lib/x86_64-linux-gnu/sane/libsane−mustek_pp.so
> The shared library implementing this backend (present on systems that support dynamic loading).

ENVIRONMENT
SANE_CONFIG_DIR
This environment variable specifies the list of directories that may contain the configuration file. On *NIX systems, the directories are separated by a colon (':'), under OS/2, they are separated by a semi-colon (';'). If this variable is not set, the configuration file is searched in two default directories: first, the current working directory (".") and then in */etc/sane.d*. If the value of the environment variable ends with the directory separator character, then the default directories are searched after the explicitly specified directories. For example, setting **SANE_CONFIG_DIR** to "/tmp/config:" would result in directories *tmp/config*, ., and */etc/sane.d* being searched (in this order).

SANE_DEBUG_MUSTEK_PP
If the library was compiled with debug support enabled, this environment variable controls the debug level for this backend. E.g., a value of 128 requests all debug output to be printed. Smaller levels reduce verbosity.

```
level    debug output
-------------------------------------
 0       nothing
 1       errors
 2       warnings & minor errors
 3       additional information
 4       debug information
 5       code flow (not supported yet)
 6       special debug information
```

SANE_DEBUG_SANEI_PA4S2
This variable sets the debug level for the SANE interface for the Mustek chipset A4S2. Note that enabling this will spam your terminal with some million lines of debug output.

```
level    debug output
---------------------------
 0       nothing
 1       errors
 2       warnings
 3       things nice to know
 4       code flow
 5       detailed code flow
 6       everything
```

SEE ALSO
sane(7), **sane−mustek**(5), **sane−net**(5), **saned**(8), **sane−find−scanner**(1), **scanimage**(1)

For latest bug fixes and information see
http://www.penguin−breeder.org/sane/mustek_pp/

For additional information on the CIS driver, see
http://home.scarlet.be/eddy_de_greef/

AUTHORS
Jochen Eisinger
<jochen at penguin−breeder dot org>

Eddy De Greef
<eddy_de_greef at scarlet dot be>

BUGS

Too many... please send bug reports to *sane−devel@alioth-lists.debian.net* (note that you have to subscribe first to the list before you can send emails... see *http://www.sane−project.org/mailing−lists.html*).

BUG REPORTS

If something doesn't work, please contact us (Jochen for the CCD scanners, Eddy for the CIS scanners). But we need some information about your scanner to be able to help you...

SANE version
Run *scanimage −V* to determine this.

the backend version and your scanner hardware
Run *SANE_DEBUG_MUSTEK_PP=128 scanimage −L* as root. If you don't get any output from the **sane−mustek_pp** backend, make sure a line "mustek_pp" is included into your */etc/sane.d/dll.conf* . If your scanner isn't detected, make sure you've defined the right port address in your *mustek_pp.conf* .

the name of your scanner/vendor also a worthy information. Please also include the
optical resolution and lamp type of your scanner, both can be found in the manual of your scanner.

any further comments
if you have comments about the documentation (what could be done better), or you think I should know something, please include it.