

**NAME**

sane-cardscan – SANE backend for Corex CardScan usb scanners

**DESCRIPTION**

The **sane-cardscan** library implements a SANE (Scanner Access Now Easy) backend which provides access to the Corex CardScan 800c & 600c small-format scanners.

The backend supports only grayscale and color modes and media of (theoretically) infinite length.

This backend may support other scanners. The best way to determine level of support is to get a trace of the windows driver in action, and send it to the author.

**OPTIONS**

The cardscan backend supports the following options:

mode m

Selects the mode for the scan. Options are "Gray" and "Color".

**CONFIGURATION FILE**

The configuration file "cardscan.conf" is used to tell the backend how to look for scanners, and provide options controlling the operation of the backend. This file is read each time the frontend asks the backend for a list of scanners, generally only when the frontend starts. If the configuration file is missing, the backend will use a set of compiled defaults, which are identical to the default configuration file shipped with SANE.

Scanners can be specified in the configuration file in 2 ways:

"usb 0x04c5 0x1042" (or other vendor/product ids)

Requests backend to search all usb buses in the system for a device which uses that vendor and product id. The device will then be queried to determine if it is a cardscan scanner.

"usb /dev/usb/scanner0" (or other device file)

Some systems use a kernel driver to access usb scanners. This method is untested.

Additionally, there are two configuration options that control the protocol used by the backend:

"lines\_per\_block 16" (or other number from 1 to 32)

Controls the number of lines of image data which will be acquired in each pass. Older scanners will require this number set lower, often 1.

"has\_cal\_buffer 1" (1 or 0)

Causes the backend to get calibration data from scanner during initialization. Older scanners do not support this request, and must be set to 0.

**ENVIRONMENT**

The backend uses a single environment variable, SANE\_DEBUG\_CARDSCAN, which enables debugging output to stderr. Valid values are:

5 Errors  
10 Function trace  
15 Function detail  
20 Option commands  
25 SCSI/USB trace  
30 SCSI/USB detail  
35 Useless noise

**KNOWN ISSUES**

The scanner does not seem to have much control possible, so the backend cannot set x/y coordinate values, resolutions, etc. These things could be simulated in the backend, but there are plenty of command line tools.

The backend also does not send all the commands that the windows driver does, so it may not function the same.

The backend does not have the calibration or ejection options of the windows driver.

**CREDITS**

The hardware to build this driver was provided to the author by:

Jeff Kowalczyk <jtk a t yahoo d o t com>

**SEE ALSO**

sane(7), sane-usb(5)

**AUTHOR**

m. allan noah: <kitno455 a t gmail d o t com>