

=====
Elo Multiple Touchscreen Linux Driver - Single Touch (ST) USB
Intel i686 (32 bit) or AMD64/Intel (64 bit) or ARMv7l (32 bit)
Installation/Calibration/Uninstallation Instructions
=====

Version 5.0.0
February 22, 2018
Elo Touch Solutions

=====
Elo Linux Multiple Touchscreen ST USB Driver package contains userspace Linux drivers designed for Linux kernels 4.x, 3.x and 2.6.x, video alignment utility and control panel utilities for Elo touchmonitors. This driver requires the presence of libusb-1.0 shared library on the target system for its operation.

The driver requires unique touch controller serial numbers to work with multiple touchscreens. Elo Intellitouch Pro PCAP controllers that do not contain a valid serial number will only work in single touchscreen mode.

This readme file is organized as follows:

1. Supported Touchmonitors and Elo Touchscreen Controllers
2. System Requirements
3. Installing the Elo Touchscreen USB Driver
4. USB Driver Commandline Options and Usage
5. Setting Active Touch Area
6. Calibrating the Touchscreen
7. Retrieving Calibration Values from NVRAM / Pre-Calibration (Optional)
8. Accessing the Control Panel
9. Uninstalling the Elo Touchscreen USB Driver
10. Troubleshooting
11. Contacting Elo Touch Solutions

=====
1. Supported Touchmonitors and Elo Touchscreen Controllers
=====

- Elo Multi Touch(MT) USB Controllers [only primary touch is processed]
TouchPro PCAP controllers,
IntelliTouch Plus/iTouch Plus 2515-07(non HID), 2521, 2515-00, 3200XX,
Multi Touch IR controllers
- Elo Single Touch(ST) USB Controllers
IntelliTouch(R) 2701, 2700, 2600, 2500U,
CarrollTouch(R) 4501, 4500U, 4000U,
Accutouch(R) 2218, 2216, 3000U,
Surface Capacitive 5020, 5010, 5000,
Accoustic Pulse Recognition(APR) Smartset 7010,
and other Elo Smartset ST USB controllers

=====
2. System Requirements
=====

Visit the Linux downloads section at www.elotouch.com to download the driver package for your 32 bit Intel, 64 bit AMD/Intel or 32 bit ARM v7l Linux.

- 32 bit Intel i686 (x86) platform (or)
64 bit AMD/Intel x86_64 platform (or)
32 bit ARM v7l platform

- Kernels supported:
 - Kernel version 4.x.x
 - Kernel version 3.x.x
 - Kernel version 2.6.x (GCC version 4.0.0 and later)
- Xorg Xwindows version supported:
 - Xorg version 6.8.2 - 7.2
 - Xorg Xserver version 1.3 and newer
- Motif versions supported:
 - Motif version 3.0 (libXm.so.3)
- libusb versions supported:
 - libusb version 1.0

=====
 3. Installing the Elo Touchscreen USB Driver
 =====

Important:
 =====

- a.) Must have root or administrator access rights on the Linux machine to install the Elo Touchscreen USB Driver.
- b.) Ensure all earlier Elo drivers are uninstalled from the system. Follow the uninstallation steps from the old driver's readme.txt file to remove the old driver completely.
- c.) The Elo Touchscreen driver components require libusb-1.0 library support (older libusb-0.1 library will not work). Most Linux distributions have started shipping this library (update to the popular libusb-0.1 library) as a part of their standard release. Customers can also download and compile the libusb-1.0 library from source (requires gcc v4.0.0 or later) available at libusb website.
- d.) Do not extract the downloaded binary package on a Windows system.
- e.) Motif 3.0 (libXm.so.3) library is required to use the Graphic User Interface (GUI) based control panel (/etc/opt/elo-usb/cpl). Openmotif or lesstif installation packages provide the required libXm.so.3 library.

Step I:

Copy the elo driver files from the binary folder to the default elo folder. Change the permissions for all the elo driver files. These broad permissions are provided to suit most systems. Please change them to tailor it to your access control policy and for specific groups or users.

- a.) Copy the driver files to /etc/opt/elo-usb folder location.

```
# cp -r ./bin-usb/ /etc/opt/elo-usb
```

- b.) Use the chmod command to set full permissions for all the users for the /etc/opt/elo-usb folder (read/write/execute). These broad permissions are provided to suit most systems. Please change them to tailor it to your access control policy and for specific groups or users.

```
# cd /etc/opt/elo-usb
# chmod 777 *
# chmod 444 *.txt
```

- c.) Copy the udev rules file to /etc/udev/rules.d/ folder location. Please edit touchscreen device permissions to tailor it to your access control policy and for specific groups or users.

```
# cp /etc/opt/elo-usb/99-elotouch.rules /etc/udev/rules.d
```

Step II: [Linux distributions with systemd init system]

Install a script to invoke Elo service through systemd init at system startup. Check if systemd init is being used in your Linux distribution and then proceed with this installation step. If systemd init is not active, proceed with Step III of the installation.

Check for active systemd init process.

```
# ps -eaf | grep [s]ystemd
# ps -eaf | grep init
# ls -l /sbin/init
```

If systemd init system is active, copy and enable the elo.service systemd script to load the elo driver at startup. Proceed to Step IV of the installation.

```
# cp /etc/opt/elo-usb/elo.service /etc/systemd/system/
# systemctl enable elo.service
# systemctl status elo.service
```

Step III: [Linux distributions with sysvinit or Upstart or older init system]

Install a script to invoke Elo service on older init systems (non systemd) at system startup.

Redhat, Fedora, Mandrake, Slackware, Mint, Debian and Ubuntu systems:

Add the following line at the end of daemon configuration script in "/etc/rc.local" file.

[rc.local file might also be at location /etc/rc.d/rc.local. Use the "# find /etc -name rc.local" command to locate the rc.local file.]

```
/etc/opt/elo-usb/loadEloTouchUSB.sh
```

SUSE Systems:

Add the following line at the end of the configuration script in "/etc/init.d/boot.local" file.

```
/etc/opt/elo-usb/loadEloTouchUSB.sh
```

Step IV:

Plug in the USB touchscreen and reboot the system to complete the driver installation process.

```
# shutdown -r now
```

The USB (elousbd) driver commandline options are listed below. If required, modify the /etc/opt/elo-usb/loadEloTouchUSB.sh script file to add commandline options to the elousbd driver startup.

- help
Print usage information and available options
- version
Display USB touchscreen driver version information
- displaycoordinates
Display the touch data, corresponding to each touch, on a terminal. Touch data consists of touch state (Touch/ Stream/ Untouch) along with X and Y coordinates. This option is used for testing the touchscreen, hence touch data is only displayed and not sent to the Xserver to move the mouse pointer.
- ignoreserialnumber
Ignore Touch controller serial number in device config file
- xwarppointer
Use XWarpPointer call to send touch events to X window system
- activetoucharea <OriginX,OriginY,Width,Height>
Set the active touch area parameters : OriginX, OriginY, Width and Height. When these parameters are set, the mouse pointer will respond to touch input within the specified area only.

Usage Examples:

```

elousbd --help
  Print usage information and available options

elousbd --version
  Display USB touchscreen driver version information

elousbd --displaycoordinates
  Enable printing of touch data to a terminal for testing the touchscreen

elousbd --ignoreserialnumber
  Ignore Touch controller serial number in device config file

elousbd --xwarppointer
  Use XWarpPointer call to send touch events to X window system

elousbd --activetoucharea 150,200,1000,700
  Set the active touch area: Origin=(150,200), Width=1000, Height=700

```

=====
5. Setting Active Touch Area
=====

Important:
=====

The "--activetoucharea" driver commandline option will render some areas of the Linux desktop inaccessible to touch input. This option currently works with one videodeskop and its behaviour is unknown in a multiple videodeskop setup. If Elo touchscreen is the only input device for the Linux system, please pay close attention to the active touch area parameters.

Step I:

Skip this step, if you already know the values for the "--activetoucharea" option parameters. Open a terminal, check if the USB touchscreen is connected.

Then unload the "elousbd" driver, if it is active. Launch the elousbd driver with "--displaycoordinates" option to determine and compute the values for "--activetoucharea" option parameters (OriginX, OriginY, Width and Height). Use the sudo command to execute the following commands, if you do not have administrator access.

```
# lsusb
    List all USB devices connected to the Linux system

# killall elousbd
    Unload the elousbd driver, if it was loaded

# cd /etc/opt/elo-usb
# ./elousbd --displaycoordinates
    Load the elousbd driver to display coordinates on the terminal
```

Now the driver will report the video coordinates corresponding to the area being touched. Use this information to determine and compute the origin coordinates (top left corner), width and height of the active touch area.

Step II:

Use the values for "--activetoucharea" option parameters (OriginX, OriginY, Width and Height) and load the elousbd driver manually.

```
# killall elousbd
    Unload the elousbd driver, if it was loaded

# elousbd --activetoucharea 150,200,1000,700
    Set the active touch area: Origin=(150,200), Width=1000, Height=700
```

To load these parameters every time the driver loads, modify the /etc/opt/elo-usb/loadEloTouchUSB.sh script file and add these commandline options to the elousbd driver startup.

Note:
=====

When the active touch area option is selected in the driver, all touch events that are generated outside the active area are discarded by the driver and not reported to Xwindows. Event reporting resumes when touch events are generated within the active area.

=====
6. Calibrating the Touchscreen
=====

Important:
=====

Users must have read and write access to "/dev/elo-usb" and "/etc/opt/elo-usb" directories to perform the touchscreen calibration. All long command line options in elova calibration utility use the "--" format. (example: "--help")

Type "# /etc/opt/elo-usb/elova --help" for available command line parameters and usage.

Step I:

The calibration utility, by default, will perform an onboard calibration for supported controllers like 2701, 2218, etc. Onboard calibration process applies the computed calibration parameters directly to the touch controller and aligns the touch coordinates output from the controller. Hence, calibration retrieval process in Step 7 is not required, if the onboard calibration process has been completed successfully.

Run the calibration utility with root privileges from a command window in X Windows from the /etc/opt/elo-usb directory for a single or multiple video setup (supports Xorg Xinerama, Xorg non-Xinerama and Nvidia Twinview options).

```
# cd /etc/opt/elo-usb
# sudo ./elova
```

If the previous process based on the --nvramp option is used to store the calibration data, then the calibration retrieval process in Step 7 is required.

```
# cd /etc/opt/elo-usb
# sudo ./elova --nvramp
```

The '--nvramp' option writes the calibration data to the Non Volatile RAM (if present) on the touchmonitor and the configuration file on the hard disk. To perform the calibration and update only the configuration file on the hard disk, for all touch controllers which do not support onboard calibration, use the command shown below.

```
# cd /etc/opt/elo-usb
# sudo ./elova
```

In a multiple video setup, the calibration target(s) will be shown on the first video screen and switch to the next video screen after a 30 second default timeout for each target or screen. Once the touchscreen is calibrated the data is stored in a configuration file on the hard disk. To display the calibration targets on just one specific video screen(example:videoscreen[1]) use the command shown below.

```
# cd /etc/opt/elo-usb
# sudo ./elova --videoscreen=1
```

To change or disable the default calibration timeout for each target or screen, use the command shown below. [Timeout Range: Min=0 (no timeout), Max=300 secs, Default=30 secs]

```
# cd /etc/opt/elo-usb
# sudo ./elova --caltargettimeout=0
    Disable the calibration timeout for all targets and videoscreens

# sudo ./elova --caltargettimeout=45
    Modify the calibration timeout to 45 seconds
```

To view a list of video and USB touch devices available for calibration, use the command shown below.

```
# cd /etc/opt/elo-usb
# ./elova --viewdevices
```

To view all the available options and specific usage for elova calibration program, use the command shown below.

```
# cd /etc/opt/elo-usb
# ./elova --help
```

Step II:

Touch the target(s) from a position of normal use. The calibration data is written to the driver at the end of calibration.

=====
7. Retrieving Calibration Values from NVRAM / Pre-Calibration (Optional)
=====

Important:

=====
This calibration retrieval step is not required if the default onboard calibration is performed on supported touch controllers like 2701, 2218, etc. Onboard calibration process applies the calibration parameters directly to the touch controller and aligns the touch coordinates output from the controller.

You should implement this calibration value retrieval step, only if the --nvram option was used with the elova calibration utility to store valid calibration data in the controller NVRAM (Non Volatile Random Access Memory) , for the Elo driver to use later.

A valid calibration must exist in the touchmonitor NVRAM (Non Volatile Random Access Memory) to use this function. Users must first perform a touchscreen calibration using elova and write the calibration values to the monitor NVRAM. The existing values in the NVRAM will be lost as only one set of calibration values can be stored in the NVRAM. Hence ensure that the current NVRAM calibration values can be overwritten before performing a new calibration and writing to the NVRAM. Note that not all touchmonitors provide this NVRAM storage feature.

Option I: [Manual Option]

To retrieve the calibration values from the NVRAM immediately, run the 'eloautocalib' utility from a command window in X Windows from the /etc/opt/elo-usb directory. The command line option '--renew' enables the reading of the calibration values from monitor NVRAM and overwriting the current values in the configuration file on the hard disk.

```
# cd /etc/opt/elo-usb
# ./eloautocalib --renew
```

To view all the available options and specific usage for eloautocalib utility, use the command shown below.

```
# cd /etc/opt/elo-usb
# ./eloautocalib --help
```

Option II: [Automatic Option]

Copy the xEloInit.sh script file present in the /etc/opt/elo-usb directory to the /etc/X11/xinit/xinitrc.d/ directory. If the destination "xinitrc.d" does not exist, edit the /etc/X11/xinit/xinitrc script file and add a line to invoke the /etc/opt/elo-usb/xEloInit.sh script file.

```
# cp /etc/opt/elo-usb/xEloInit.sh /etc/X11/xinit/xinitrc.d/
```

To retrieve the calibration values from the NVRAM automatically on system startup, enable the 'eloautocalib' entry in the 'xEloInit.sh' script file located in the '/etc/X11/xinit/xinitrc.d/' directory. The eloautocalib entry is commented out by default and does not load the calibration values from monitor NVRAM. Uncomment the entry '/etc/opt/elo-usb/eloautocalib --renew' to enable reading the calibration values from monitor NVRAM and overwriting the current values in the configuration file on the hard disk during system startup.

Default: '# /etc/opt/elo-usb/eloautocalib --renew'
Does not load calibration values from NVRAM

Modified: '/etc/opt/elo-usb/eloautocalib --renew'
Loads calibration values from NVRAM

=====
8. Accessing the Control Panel
=====

The control panel application allows the user to easily set the available driver

configuration options. After the driver package is installed, change to the /etc/opt/elo-usb directory and run the control panel application(cpl or cplcmd).

Important:
=====

Users must have read and write access to "/dev/elo-usb" folder to run the control panel applications.

Step I:

Run the control panel utility with root privileges from a command window in X Windows from the /etc/opt/elo-usb directory. Motif version 3.0 (libXm.so.3) is required to use the GUI based control panel (/etc/opt/elo-usb/cpl). If Motif or GUI control panel(cpl) is not present, use the command line version of the application(cplcmd) in Step III.

```
# cd /etc/opt/elo-usb
# sudo ./cpl
```

Step II:

Navigate through the various tabs by clicking on them. Here is an overview of information related to each tab.

General	- Perform touchscreen calibration
Mode	- Change the touchscreen mode
Sound	- Change Beep on Touch Parameters (Enable/Disable Beep, Beep Tone, Beep Duration)
Touchscreen-0	- Display data related to the USB touchscreen 0.
Touchscreen-1	- Display data related to the USB touchscreen 1.
About	- Information about the package. Click on the Readme button to open this Readme file.

Step III:

If Motif is not installed or GUI control panel(cpl) is not present, use the command line version of the application(cplcmd) to access the control panel. Run the command line application from a command window in X Windows from the /etc/opt/elo-usb directory.

```
# cd /etc/opt/elo-usb
# ./cplcmd
```

=====
9. Uninstalling the Elo Touchscreen USB Driver
=====

Important:
=====

Must have root or administrator access rights on the Linux machine to uninstall the Elo Touchscreen USB Driver.

Step I:

Delete the script or commands that invoke Elo service at startup.

Linux with Systemd init system:

Disable and remove the elo.service startup script registered with systemd init

enable and remove the elo.service startup script registered with systemd init system in Step II of Installation section.

```
# systemctl status elo.service
# systemctl stop elo.service
# systemctl disable elo.service
# systemctl status elo.service
# rm -rf /etc/systemd/system/elo.service
```

Linux with sysvinit or Upstart or older init system:

SUSE systems:

Remove the following entry created in Step III of Installation section from the configuration script in "/etc/init.d/boot.local" file.

```
/etc/opt/elo-usb/loadEloTouchUSB.sh
```

Redhat, Fedora, Mandrake, Slackware, Mint, Debian and Ubuntu systems:

Remove the following entry created in Step III of Installation section from the configuration script in "/etc/rc.local" file. (or "/etc/rc.d/rc.local" file)

```
/etc/opt/elo-usb/loadEloTouchUSB.sh
```

Step II:

Delete all the elo driver files from the system.

a.) Delete the main elo driver folder.

```
# rm -rf /etc/opt/elo-usb
```

b.) Delete the elo related device folder and files.

```
# rm -rf /dev/elo-usb
# rm -rf /etc/udev/rules.d/99-elotouch.rules
```

Step III:

Reboot the system to complete the driver uninstallation process.

```
# shutdown -r now
```

10. Troubleshooting

A. Make sure libusb-1.0 library is installed on the target Linux system. The driver will NOT work with the older libusb-0.1 library. Most Linux distributions ship with the newer libusb-1.0 library installed by default. It can also be installed by downloading and compiling the library source (requires gcc v4.0.0 or later) from the libusb-1.0 website.

B. If touch is not working, check if the elousbd driver is loaded and currently available in memory. Some Xorg Xserver versions terminate the touchscreen driver upon user logout. The current workaround in this situation is to startup the driver from Xwindows startup script or reboot the system.

```
# ps -e |grep elo
```

Check the driver log file for any errors that have been reported.

```
# gedit /var/log/elo-usb/EloUsbErrorLog.txt
```

If the driver is not present then load the driver again. Root access is needed to load the driver manually. Normal users will have to restart the system so that the elousbd daemon is loaded again during system startup. Normal users may be able to load the driver manually depending on access control and file permissions that are setup.

```
# sudo /etc/opt/elo-usb/elousbd
```

- C. If starting the Elo touchscreen driver from the normal startup locations like rc.local or boot.local does not work, first test if the touchscreen is working by manually launching the driver from a terminal window within XWindows graphics desktop session.

```
# sudo /etc/opt/elo-usb/loadEloTouchUSB.sh
```

If the touchscreen works when the driver is launched manually, try to add the touchscreen driver startup line to the end of one of the XWindows startup scripts. The Xwindows startup scripts are located usually in the following path /etc/X11/xinit/xinitrc.d/. Running the touchscreen driver from the Xwindows startup script will provide touch input ONLY after the user has logged in successfully at the GUI Login screen.

- D. Beep-on-touch feature does not work in the GUI control panel sound tab (Beep Test button) or if the driver is loaded manually from a non-root user context. The driver has to be loaded from a system startup script or root user account for beep-on-touch to function properly. The beep on touch feature also depends on the pcspkr(PC Speaker) kernel module. Ensure that the pcspkr kernel module is loaded and active in memory using the lsmod command.

- E. While trying to load the driver manually, if you get an error "Error opening USB_ERROR_LOG_FILE", check the file permissions for the /var/log/elo-usb/EloUsbErrorLog.txt file. The user needs to be the root user or have read and write access to this log file to launch the driver. Try using the sudo command to launch the driver manually, if its a non root user.

- F. If the target Linux platform has multiple video screens configured in separate X video screen mode or Xinerama mode, the cursor may not always respond to touch on the proper video screen after calibration. The default call the driver uses to send touch events, XTestFakeMotionEvent has a bug that prevents the switching of cursor across video screens in separate X video screen mode or Xinerama mode[Xorg v7.4 or later]. In this case, launch the driver with "--xwarppointer" commandline parameter to use XWarpPointer call instead of the XTestFakeMotionEvent call to send touch events to X windows system.

- G. Newer Linux distributions are starting to switch to the new systemd init system startup mechanism. If the addition of the Elo startup script loadEloTouchUSB.sh to rc.local or boot.local scripts does not load the Elo driver on reboot, check if systemd init system is active. If systemd init is active then register and enable the elo.service systemd startup script as per instructions in Step II of the Installation section.

=====
11. Contacting Elo Touch Solutions
=====

Website: <http://www.elotouch.com>

E-mail: customerservice@elotouch.com

Mailing Address:

Elo Touch Solutions Inc,
670 N. McCarthy Blvd,
Milpitas, CA 95035
USA

Phone: (800) 557-1458
(408) 597-8000

=====

Copyright (c) 2018 Elo Touch Solutions

All rights reserved.

=====