Driver Software

dreaMTouch

Linux
Installation and Usage Guide

READ CAREFULLY BEFORE FIRST INSTALLATION!

Revision

Rev.	Description of Changes	Name	Datum
01	initial version	TTHA	2009-08-04
02	Integrated HID device installation added Description of embedded Web Server added Several minor changes	TTHA	2010-05-19
03	Migration tool added Registry contents description added Detailed CSS description added New document layout Trouble Shooting Guide added	TTHA	2010-07-19
04	Renamed to “Installation and Usage Guide” New command line options Extended trouble shooting	TTHA	2010-09-24
05	Chapter 3 changed completely Minor changes to trouble shooting guide	TTHA	2010-09-28
06	New TUIO protocol option New Error Display Registry options description added Added control center dialog description Several chapter rearranged an rewritten	TTHA	2010-10-26
07	New Mouse emulation options: “Enter Mode” and “Left/Right Mode”	TTHA	2011-05-05
08	Complete rework for new driver release Citmuto03-1.3.25	TTHA	2012-02-23
09	Standard/Advanced user interface mode added	TTHA	2012-03-01
10	Adapted Manual for Linux	KBRO	2012-05-15
11	Added qt4 package installation as prerequisite aligned with citmuto03 Windows manual : added description of new GUI elements for detection tweaking added description of loadable touch configuration files	KBRO	2012-07-31
12	Added note about pinch-zoom gesture in touch preview, misc. spelling and typo corrections	KBRO	2012-08-10
13	configuration key “extraTuio” added (for *citmuto03 1.3.27-4* or newer)	KBRO	2013-02-21
14	Aligned with Windows manual	KBRO	2013-07-22
15	Added instructions on using pkg repositories	KBRO	2013-10-22

Exclusion of liability

The contents of this manual serve for information purposes only. Citron GmbH reserves the right to change the contents of this manual without prior notice. While reasonable efforts have been made in the preparation of this manual to assure its accuracy, errors may occur. Therefore, Citron GmbH assumes no liability resulting from errors or omissions in this manual or from the use of the information contained herein.
Citron GmbH appreciates suggestions with regard to improvements or corrections.

Document information

Filename: \\ntserv1\dokument\MANUALS\dreaMTouch\citmuto03\dreaMTouch-Linux_Installation_and_Usage_Guide-en-R15.odt

Table of Contents

1 Preface

The Citron „*dreaMTouch*“ is a new kind of infrared multi-touch with unique features.

The driver software suite for the dreaMTouch is named “citmuto03”. This document describes installation and usage of the driver software on GNU/Linux OS based on the Debian 6.0 distribution.

2 System Requirements

In order to be able to use dreaMTouch and citmuto03 the following minimum hardware requirements have to be satisfied:

- CPU: Intel Core2Duo 2 GHz (or faster)
- at least 1 GB RAM
- 25 MByte free space on hard disk
- One free USB 1.1 (full-speed) port

The dreaMTouch driver software demands following software prerequisites to be met :

- GNU/Linux >= 2.6.32 (with Multitouch Support)
- Qt >= 4.6
- libtcmalloc_minimal0

- additionally/optional an multi-touch capable version of X.org XServer (1.12), xf86-input-evdev (2.7.0) and a graphics toolkit (Gtk/Qt) with enabled Xinput 2.2 protocol support for full native Linux multitouch experience.

citmuto03 comes packaged as binary for Debian 6.0 and Ubuntu 12.04LTS by default, if you need packages for other distributions, please contact us (support@citron.de).

Up to 8 dreaMTouch devices are supported by the dreaMTouch driver. Although, due to overall system performance, it is not recommended to connect so many devices to a single PC.

3 Software Installation

The driver files are distributed in tarballs containing distribution packages.

Before attaching the dreaMTouch hardware to the computer the appropriate driver software should be installed.

The software consists of several parts as laid out in the table below

Name	Installed Files	Description
Citmuto03drv	/etc/init.d/citmuto03 /etc/udev/rules.d/99-mtir.rules /etc/xdg/Citron\ GmbH/dreaMTouch.conf /usr/sbin/citmuto03drv /usr/share/doc/citmuto03drv/changelog.Debian.gz /usr/share/doc/citmuto03drv/changelog.gz /var/www/ciwebs/dreaMTouch/default.css /var/www/ciwebs/dreaMTouch/favicon.ico /var/www/ciwebs/dreaMTouch/index.html /var/www/ciwebs/dreaMTouch/logo.jpg	User-space driver service for Citron dreaMTouch multitouch screens
Citmuto03gui	/usr/bin/citmuto03gui /usr/lib<,64>/libQtSolutions_SingleApplication-head.so* /usr/share/applications/citmuto03gui-root.desktop /usr/share/applications/citmuto03gui.desktop /usr/share/doc/citmuto03gui/changelog.Debian.gz /usr/share/doc/citmuto03gui/changelog.gz /usr/share/pixmaps/citmuto03gui.png	User interface for citmuto03drv service
libmtirctl1	/usr/lib<,64>/libmtirctl.so.* /usr/share/doc/citmuto03/LICENSE.gz /usr/share/doc/libmtirctl1/changelog.Debian.gz /usr/share/doc/libmtirctl1/changelog.gz	Foundation communication library for Citron dreaMTouch multitouch screens
libciwebs1	/usr/lib<,64>/llibciwebs.so* /usr/share/doc/libciwebs1/changelog.Debian.gz /usr/share/doc/libciwebs1/changelog.gz	Citron embedded web server library, needed by citmuto03drv
Libftd2xx1	/etc/modprobe.d/blacklist-mtir.conf /usr/lib<,64>/libftd2xx.so* /usr/share/doc/libftd2xx1/changelog.Debian.gz /usr/share/doc/libftd2xx1/changelog.gz	Third party communication library for FTDI ICs, needed by libmtirctl1, included for user convenience

3.1 Step by Step Installation (on Debian / Ubuntu)

You can install the citmuto03 dreaMTouch/xtreMTouch driver suite from different sources.

3.1.1 From distribution tarball

Manual installation as described below, gives you the most possible control over what packages are installed. On the down-side the automatic package dependency resolution mechanism of the package manager is circumvented, so you'll have to install prerequisite packages and perform package updates yourself.

1) **Install prerequisite packages (provided by OS Vendor)**

   Within a Terminal (you might need the administrator password for this) enter

   ```
   sudo apt-get install libtcmalloc-minimal0 libqtcore4 libqtgui4 libqt4-network
   ```

2) **Unpack the provided driver tarball**

   ```
   tar xf citmuto03-<version>-<release>-<distro>-<arch>.tar.bz2
   ```

   installing citmuto03 1.3.25 on Debian 6, 32 Bit this translates to

   ```
   tar xf citmuto03-1.3.25-1-debian6-i386.tar.bz2
   ```

3) **Change into decompressed directory**

   ```
   cd citmuto03-<version>-<release>-<distro>-<arch>
   ```

   installing citmuto03 1.3.25 release 1 on Debian 6, 32 Bit this translates to

   ```
   cd citmuto03-1.3.25-1-debian6-i386
   ```

4) **Install driver packages**

   The following command-line is to be input as one line without “\” and reformatted here for better readability

   ```
   sudo dpkg -i libftd2xx1_<version>-<release>_<arch>.deb \
           libciwebs1_<version>-<release>_<arch>.deb \
           libmtirctl1_<version>-<release>_<arch>.deb \
           citmuto03drv_<version>-<release>_<arch>.deb \
           citmuto03gui_<version>-<release>_<arch>.deb
   ```

   installing citmuto03 1.3.25 release 1 on Debian 6, 32 Bit this translates to

   ```
   sudo dpkg -i libftd2xx1_1.3.25-1_i386.deb \
           libciwebs1_1.3.25-1_i386.deb \
           libmtirctl1_1.3.25-1_i386.deb \
           citmuto03drv_1.3.25-1_i386.deb \
           citmuto03gui_1.3.25-1_i386.deb
   ```

5) **Configure driver package**

   ```
   dpkg-reconfigure citmuto03drv
   ```

   follow the onscreen text dialog

3.1.2 From Citron Driver CD-ROM repository

Installation of the citmuto03 dreaMTouch/xtreMTouch driver suite from the Citron Driver CD-ROM, packaged with the touchscreen device, is recommended where a stable working version of the driver is needed and the host computer has no access to the internet.

1) **Mount the Citron Driver CD-ROM**

   ```
   sudo mount /dev/sr0 /media/cdrom
   ```

2) **Add the Citron GPG key to the trusted keys database**

   ```
   sudo apt-key add /media/cdrom/key/citmuto.gpg.key
   ```

3) **Add the CD-ROM to pkg sources**

   ```
   sudo apt-cdrom add
   ```

4) **Update apt package cache**

   ```
   sudo apt-get update
   ```

5) **Install packages**

   ```
   sudo apt-get install citmuto03gui
   ```

6) **Optional: Remove CD-ROM from pkg sources**

   Remove all lines beginning with "*deb cdrom:[Citmuto CDROM driver repository*" from */etc/apt/sources.list*

7) **Configure driver package**

   ```
   dpkg-reconfigure citmuto03drv
   ```

   follow the onscreen text dialog

3.1.3 From Citron webserver repository

Installation of the citmuto03 dreaMTouch/xtreMTouch driver suite from the Citron webserver repository is recommended as it provides the most up-to-date version of the packages and resolution of package dependencies is automatically performed by the package manager. On the down-side the host computer must have access to the internet at least during installation or update.

1) **Add the Citron GPG key to the trusted keys database**

Download http://www.citron.de/fileadmin/downloads/repos/key/citmuto.gpg.key

sudo apt-key add citmuto.gpg.key

2) **Add the Citron webserver repository to your pkg sources**

Add an entry in your /etc/apt/sources.list as follows

If you are using Debian:

deb http://www.citron.de/fileadmin/downloads/repos/apt/debian <distro> non-free

If you are using Ubuntu:

deb http://www.citron.de/fileadmin/downloads/repos/apt/ubuntu <distro> non-free

where <distro> is either one of the following (matching your installed Debian/Ubuntu distribution):

<distro>	distribution
squeeze	Debian 6 “Squeeze”
wheezy	Debian 7 “Wheezy”
jessie	Debian 8 “Jessie”
precise	Ubuntu 12.04 LTS “Precise Pangolin”
quantal	Ubuntu 12.10 “Quantal Quetzal”
raring	Ubuntu 13.04 “Raring Ringtail”

3) **Update the apt package cache**

sudo apt-get update

4) **Install packages**

sudo apt-get install citmuto03gui

5) **Configure driver package**

dpkg-reconfigure citmuto03drv

follow the onscreen text dialog

3.2 Removing the driver package

In a Terminal (you might need the administrator password for this) enter

```
sudo apt-get purge citmuto03gui citmuto03drv citmuto03-doc libftd2xx1 libmtirctl1 libciwebs1
```

3.3 First Steps

All settings for a dreaMTouch device are set to carefully selected defaults after installation. So there is usually no requirement to make any changes.

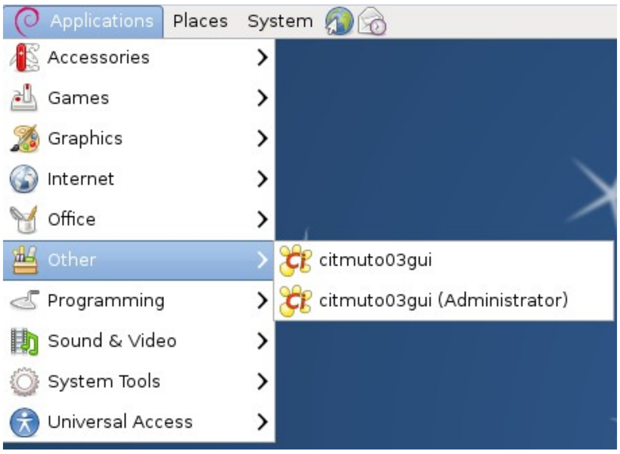
⚠ Nevertheless it is higly recommended that after installation, or whenever a new dreaMTouch device is operated for the first time at this PC, to perform the "*Clean Touch*" step once for this device. The exact procedure is described in chapter "4.3.3.3.1 Cleaning" on page 25.

3.4 Program Start

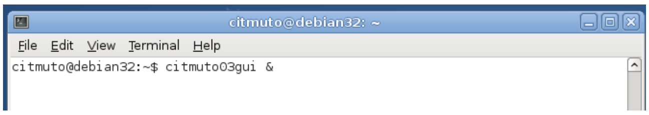
The driver itself is installed as a system service and started during installation (if selected during driver package configuration). That way a dreaMTouch device can be used even before log-in.

To make changes to the dreaMTouch information and to monitor driver operations a GUI application called "citmuto03gui" is provided.

citmuto03gui can be started from the Applications->Other desktop-menu.



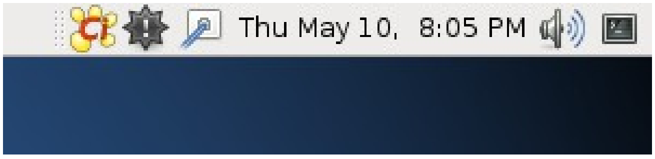
Or alternatively from any Terminal console via the "*citmuto03gui &*" command



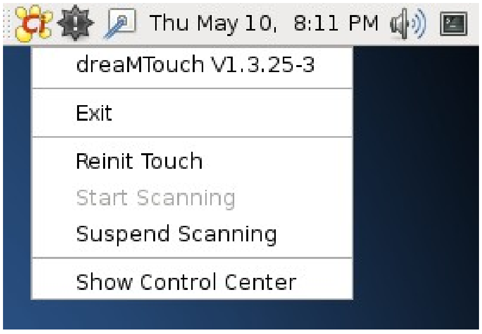
```
citmuto@debian32:~$ citmuto03gui &
```

All settings that control the way a dreaMTouch reports the detected touch points to the operating system may be changed by Administrators only. Therefore these settings are disabled (greyed-out) normally. To be able to change those settings, citmuto03gui must either be run in Administrator Mode or from a Root-Terminal.

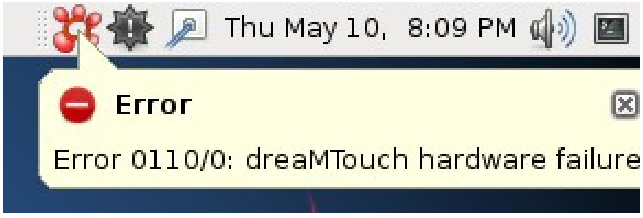
A running citmuto03gui can be identified by its icon in the system tray.



You can start the driver application GUI called “citmuto control center” by right clicking on that icon and selecting the “Show Control Center” menu item.



When no touch is connected or some other error occurs, a bubble message appears and the tray icon changes to red. More information on this can be found in chapter 8 “Trouble Shooting” on page 49.

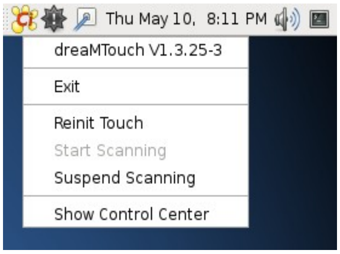


Error

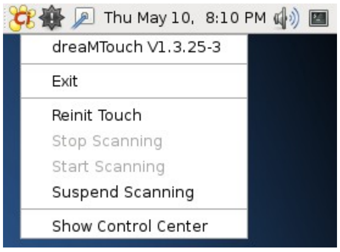
Error 0110/0: dreaMTouch hardware failure

3.5 Tray Menu

The major functions of citmuto can be controlled using a tray pop-up menu.



Tray Menue: Standard Mode



Tray Menue: Advanced Mode

The menu can be activated by a single left or right button click on the citmuto tray icon. Not all menu items are available to regular users. All functions that could stop touch from operating are allowed for administrators only. To switch to administrator mode click the “Change Settings” button in the Control Center.

Menu Item	Description
-	Just a header with dreaMTouch driver version; shows control center on click
⏻ Exit	Quit citmuto03gui. The driver service continues operation, touch is still functional after exiting the GUI.
↻ Reinit Touch	Reinitialises all attached touch devices
■ Stop Scanning	Stops scanning on all attached devices. **CAUTION:** After clicking that menu item no touch function is available any more! So make sure a mouse or other input device is connected to the PC. Otherwise the touch can be re-enabled by a reboot only.
▶ Start Scanning	Re-Start scanning after suspend or stop was activated.
⏸ Suspend Scanning	Stop scanning for 30 seconds and restart scanning after that time automatically. This is useful to prevent interference with infrared remote controls.
⬆ Show Control Center	The control center is shown.

4 Control Center “citmuto03gui”

The only application of the dreaMTouch Driver Suite that can be started by the user is the dreaMTouch Control Center “citmuto03gui”. This application communicates with the actual driver service. It is able to show current driver status and control the driver settings. But it is not required for driver operations. When started from “start-up” folder at system start citmuto03gui is placed into the system tray. If you start the application from “Start Menu” or from desktop shortcut the application window is opened up immediately.

4.1 Command Line Options

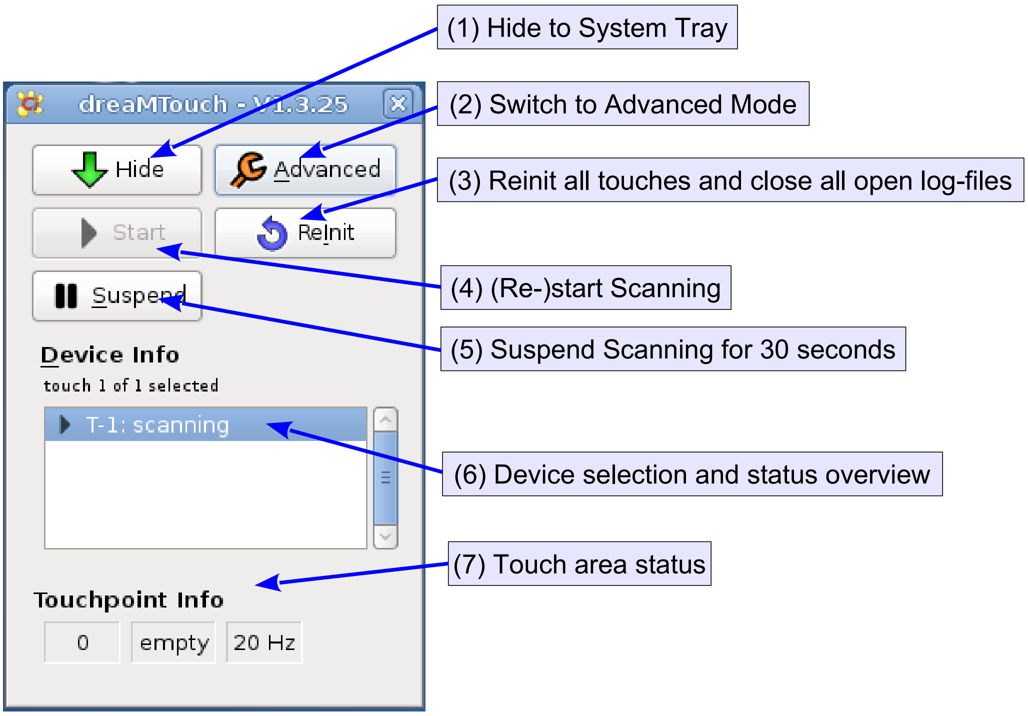
There are several command line options available to change citmuto03gui's general behavior. Options are case insensitive. If more than one option shall be given they must be separated by spaces.

Option	Description
-quiet	suppress any messages from popping up from the system tray. This option ist set by installer for all shortcuts if “Quiet” option was selected during installation.
-shownormal	Show GUI as regular window instead of hiding to system tray. This option is used by shortcuts from desktop or start menue, but not for startup folder.
-close	If there is another instance of citmuto03gui running close this instance. Do nothing if no other instance is found. This option is used by installer during uninstall.

4.2 Standard Mode

The dreaMTouch Control Center has to modes of operation: Standard Mode and Advanced Mode. Standard Mode is enough for a regular users needs to operate a dreaMTouch. For special settings and troubleshooting the advanced mode offers a lot more information and settings.

In Standard Mode the dreaMTouch Control Center shows essential controls and information only.



(1) Hide to System Tray

(2) Switch to Advanced Mode

(3) Reinit all touches and close all open log-files

(4) (Re-)start Scanning

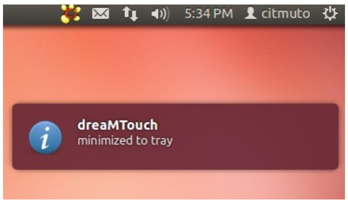
(5) Suspend Scanning for 30 seconds

(6) Device selection and status overview

(7) Touch area status

(1) Hide to System Tray

A click on this button closes the dialog window, but the application continues to execute. It is just hidden to system tray. The same happens when the close button at the top right corner of the window gets clicked. If not suppressed by the “-quiet” switch a balloon message pops up to inform the user that the application continues execution in tray.



dreaMTouch
minimized to tray

The only way to actually exit citmuto03gui is the Exit entry in the tray icon's pop-up menu.

(2) Switch to advanced mode

A click on this button switches to “Advanced Mode”.

(3) Reinit all touches and close all log-files

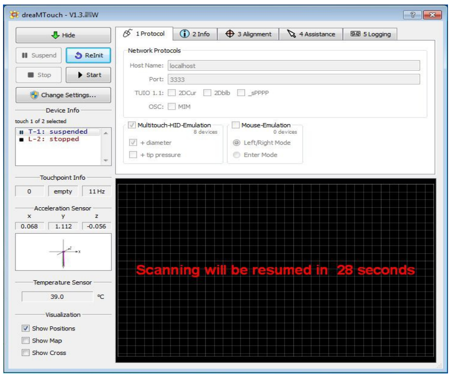
A click on this button restarts the search for connected touches. Usually this is done automatically, but sometimes a manual intervention is required.
On the other side a reinit is the only way to close opened log files.

(4) Start scanning

If scanning was stopped by a click on “Suspend” it can be resumed with this button without waiting until the 30 second timeout is over.

(5) Suspend scanning for 30 seconds

For some monitors the infrared beams of a dreaMTouch device interfere with the infrared remote control of the monitor. To be able to use the remote control scanning can be suspended for 30 seconds by a click on this button. After that time scanning is resumed automatically so no other mouse or keyboard device is needed to operate the dreaMTouch again.



touch 1 of 2 selected

```
T-1: suspended
L-2: stopped
```

Host Name: localhost
Port: 3333
TUIO 1.1: □ 2DCur □ 2Dblb □ _sPPPP
OSC: □ MIM
☑ Multitouch-HID-Emulation 8 devices
☑ + diameter
□ + tip pressure
□ Mouse-Emulation 0 devices
● Left/Right Mode
○ Enter Mode

0	empty	11 Hz

x	y	z
0.068	1.112	-0.056

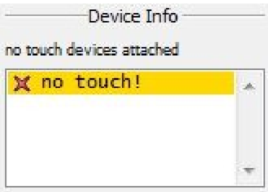
39.0 °C

☑ Show Positions
□ Show Map
□ Show Cross

Scanning will be resumed in 28 seconds

(6) Device selection and status overview

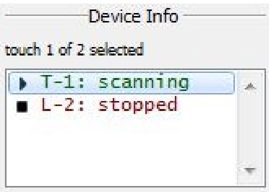
Each device attached and each log-file opened appears with its current status in this list box.



no touch devices attached

```
no touch!
```

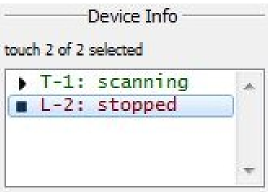
No touch attached



touch 1 of 2 selected

```
T-1: scanning
L-2: stopped
```

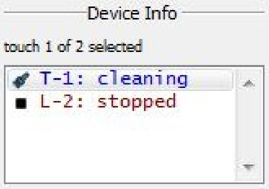
Touch 1 selected, one log open



touch 2 of 2 selected

```
T-1: scanning
L-2: stopped
```

Logfile selected



touch 1 of 2 selected

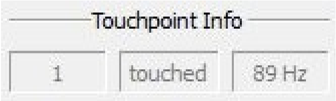
```
T-1: cleaning
L-2: stopped
```

Touch#1 is cleaning

The letter „T“ represents real dreaMTouch-devices while the letter „L“ is used for log-files. A letter „X“ is drawn whenever the driver core has not reported the full status of the device to the GUI, yet.

(7) Touchpoint Info

The tree elements of this sections show the current status of the touch area. The number of currently detected touch points is shown on the left hand side. The general status („empty“, „touched“, or „overflow“) is shown in the middle. And the current scanning speed in scan frames/second is shown on the right hand side.

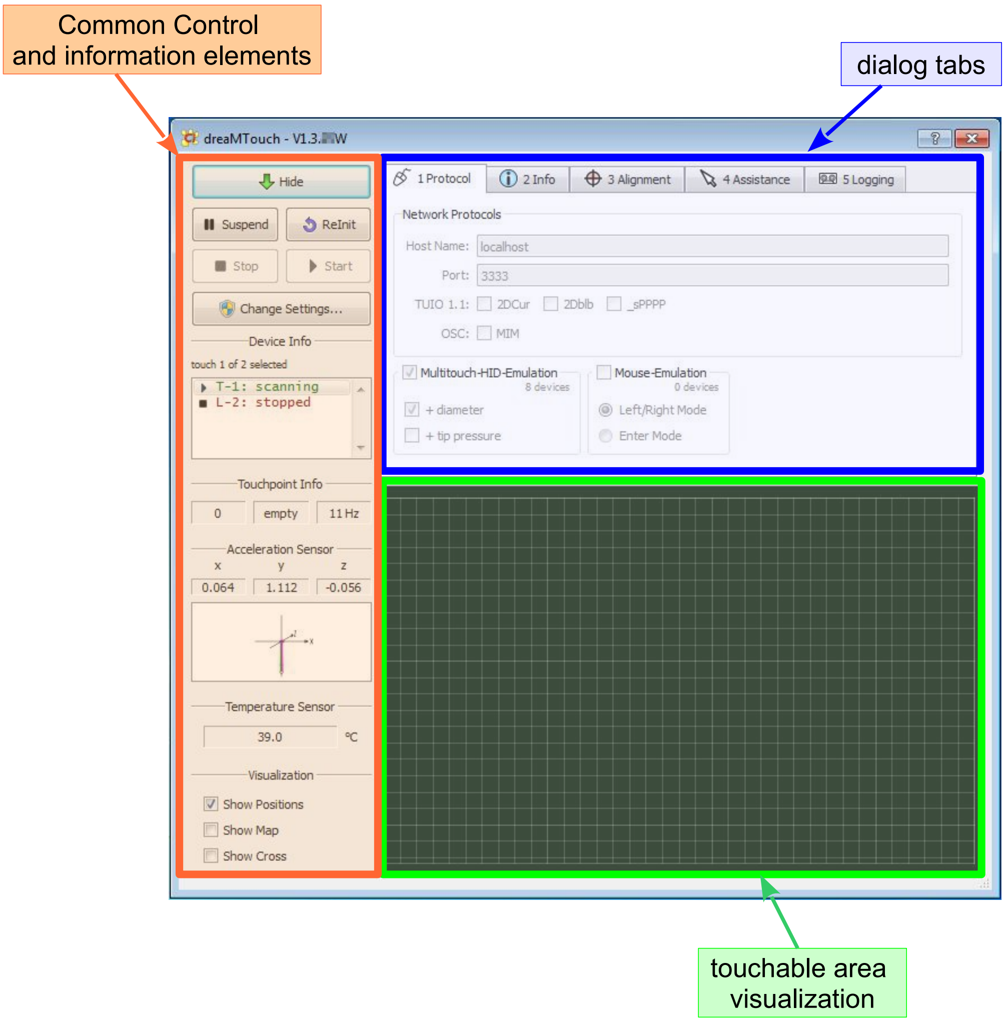


Touchpoint Info

1	touched	89 Hz

4.3 Advanced Mode

The dreaMTouch Control Center is divided into three areas. On the left hand side are some common control and information elements. At to lower right hand side is a visualization of the current touchable area. And at the upper right hand side are several dialog tabs to make settings or get more information.



Device Info

touch 1 of 2 selected

```
▸ T-1: scanning
■ L-2: stopped
```

Touchpoint Info

0	empty	11 Hz

Acceleration Sensor

x	y	z
0.064	1.112	-0.056

Temperature Sensor

39.0 °C

Visualization

☑ Show Positions
☐ Show Map
☐ Show Cross

Network Protocols

Host Name: localhost

Port: 3333

TUIO 1.1: ☐ 2DCur ☐ 2Dblb ☐ _sPPPP

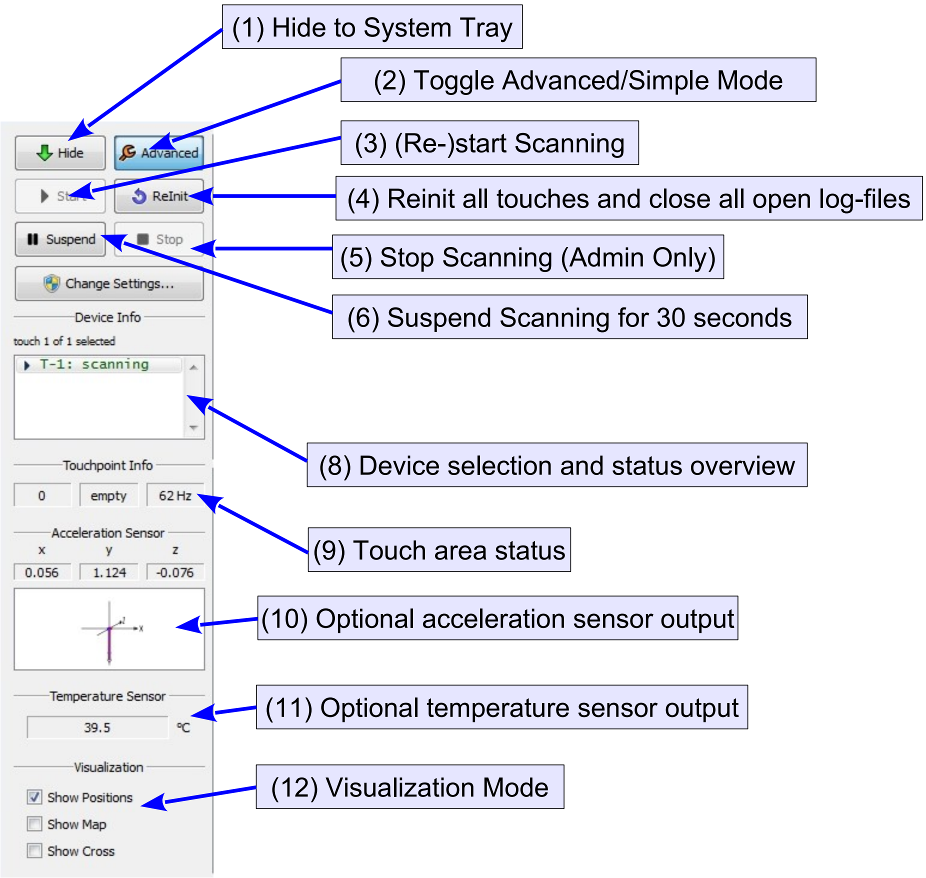
OSC: ☐ MIM

☑ Multitouch-HID-Emulation (8 devices)
☑ + diameter
☐ + tip pressure

☐ Mouse-Emulation (0 devices)
● Left/Right Mode
○ Enter Mode

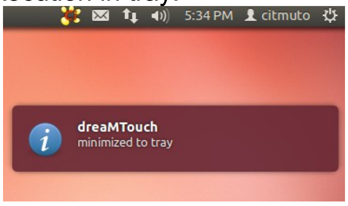
4.3.1 Common Control and Information Elements

Independent of the current dialog tab selected there are some control and informational elements on the left hand side of the dreaMTouch control center.



(1) Hide to System Tray

A click on this button closes the dialog window, but the application continues to execute. It is just hid to system tray. The same happens when the close button at the top right corner of the window gets clicked. If not suppressed by the “-quiet” switch a balloon message pops up to inform the user that the application continues execution in tray.



dreaMTouch
minimized to tray

The only way to actually exit citmuto03gui is the Exit entry in the tray icon's pop-up menu.

(2) Toggle Advanced/Standard Mode

A click on this button switches back to “Standard Mode”.

(3) (Re-)Start scanning

If scanning was stopped by a click on “Suspend” or “Stop” it can be resumed with this button.

(4) Reinit all touches and close all log-files

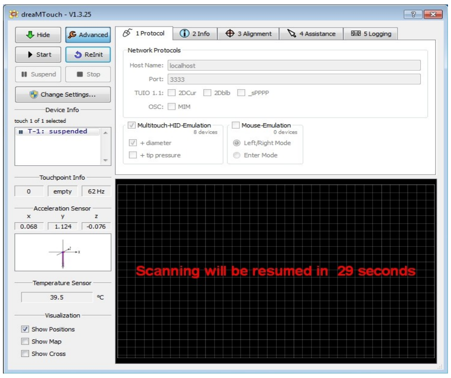
A click on this button restarts the search for connected touches. Usually this is done automatically, but sometimes a manual intervention is required.
On the other side a reinit is the only way to close opened log files.

(5)Stop scanning (Admin only)

If for some circumstances a complete stop of scanning is required (e.g to make longer setups with a remote control) this can be done by a click on stop-button.
Because scanning cannot be resumed without additional input devices (like mouse or keyboard), this action is allowed in administrator mode only. And there is a warning dialog box before scanning is actually stopped, too.

(6)Suspend scanning for 30 seconds

For some monitors the infrared beams of a dreaMTouch device interfere with the infrared remote control of the monitor. To be able to use the remote control scanning can be suspended for 30 seconds by a click on this button. After that time scanning is resumed automatically so no other mouse or keyboard device is needed to operate the dreaMTouch again.



dreaMTouch - V1.3.25

Network Protocols
Host Name: localhost
Port: 3333
TUIO 1.1: □ 2DCur □ 2Dblb □ _sPPPP
OSC: □ MIM
☑ Multitouch-HID-Emulation 8 devices
☑ + diameter
□ + tip pressure
□ Mouse-Emulation 0 devices
● Left/Right Mode
○ Enter Mode

Device Info
touch 1 of 1 selected
T-1: suspended

Touchpoint Info
0 | empty | 62 Hz

Acceleration Sensor
x 0.068 y 1.124 z -0.076

Temperature Sensor
39.5 °C

Visualization
☑ Show Positions
□ Show Map
□ Show Cross

Scanning will be resumed in 29 seconds

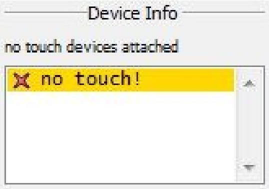
(7) Start GUI with elevated rights

By a click on the “Change Settings” button the system requests elevated rights from the user

If granted citmuto03gui is restarted in elevated mode

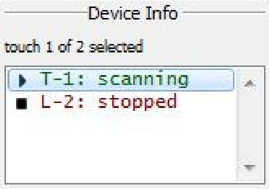
(8) Device selection and status overview

Each device attached and each log-file opened appears with its current status in this list box. By clicking on a line that devices is selected and its settings are shown in the other elements of the dialog.



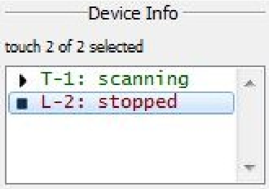
Device Info
no touch devices attached
no touch!

No touch attached



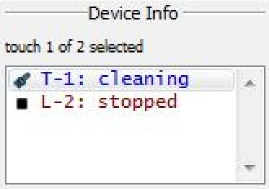
Device Info
touch 1 of 2 selected
T-1: scanning
L-2: stopped

Touch 1 selected, one log open



Device Info
touch 2 of 2 selected
T-1: scanning
L-2: stopped

Logfile selected



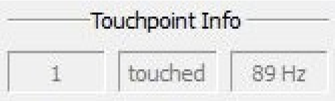
Device Info
touch 1 of 2 selected
T-1: cleaning
L-2: stopped

Touch#1 is cleaning

Also the current status of each device is shown here. The letter „T“ represents real dreaMTouch-devices while the letter „L“ is used for log-files. A letter „X“ is drawn whenever the driver core has not reported the full status of the device to the GUI, yet.

(9) Touchpoint Info

The tree elements of this sections show the current status of the touch area. The number of currently detected touch points is shown on the left hand side. The general status („empty“, „touched“, or „overflow“) is shown in the middle. And the current scanning speed in scan frames/second is shown on the right hand side.

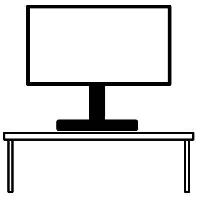
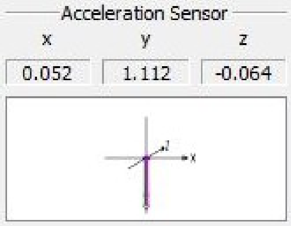
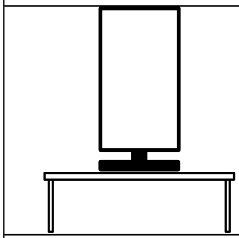
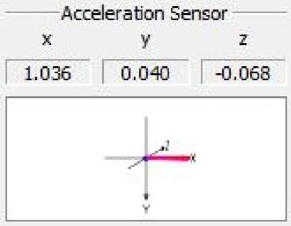
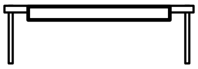
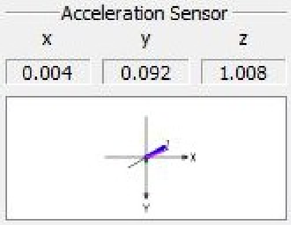


Touchpoint Info

1 | touched | 89 Hz

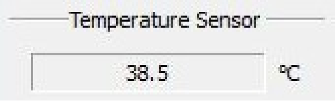
(10) Optional acceleration sensor output

If the device or log-file selected in (7) has an acceleration sensor assembled its output is shown here in a numerical and graphical manner.

Position	Visualization	Description
	 Acceleration Sensor x: 0.052 y: 1.112 z: -0.064	**Landscape** orientation; vertical wall or desktop mount: x ≈ 0 y ≈ +1 z ≈ 0
	 Acceleration Sensor x: 1.036 y: 0.040 z: -0.068	**Portrait** orientation; vertical wall or desktop mount: x ≈ +1 y ≈ 0 z ≈ 0
	 Acceleration Sensor x: 0.004 y: 0.092 z: 1.008	**Table mount** : x ≈ 0 y ≈ 0 z ≈ +1

(11) Optional temperature sensor output

If the device or log-file selected in (7) has a temperature sensor assembled its output is shown here in a numerical manner in degree Celsius.

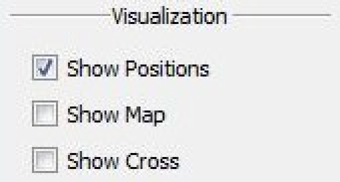


Temperature Sensor

38.5 °C

(12) Visualization mode

The lower right hand side of the control center window shows a graphical representation of the touch area of the currently selected device or log-file. The three check boxes in this section control how the touch map graphics are rendered. Details about visualization are given in the next chapter



Visualization

☑ Show Positions

☐ Show Map

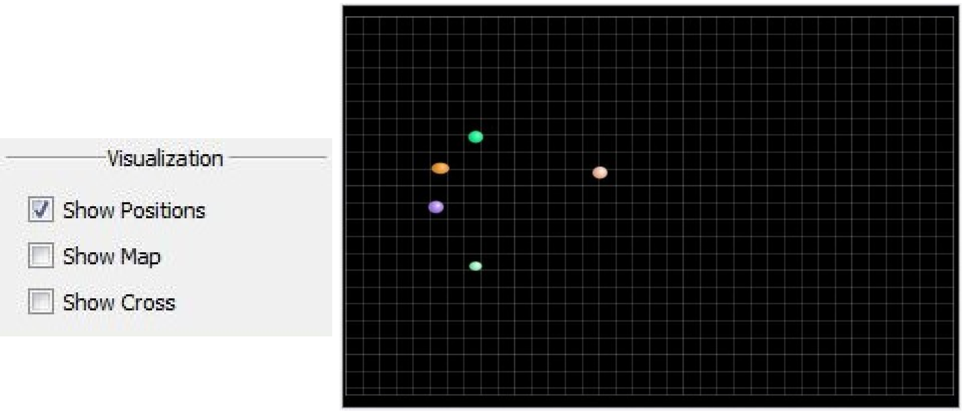
☐ Show Cross

4.3.2 Touch-Area Visualization

A scaled representation of the detected touch points of the currently selected device is given in the lower right hand area of the control center window.
The so called touch map has a certain extension on map-point units. The exact dimensions depend on the actual dreaMTouch device. The grid shown in this area has a thin line every 10 map pixels and a slightly thicker line every 100 map pixels.

The grid can be switched on and off with the “Show Grid” check-box.

Zooming can be done with a mouse wheel or by performing the pinch-zoom gesture on the visualization area. Panning of the zoomed area can be done by holding the left mouse button and moving around. To resize to fit press „F“-key or make a double click into the map.
Each touch point has a unique ID that is reported to the system via the various protocol options. That ID is represented by the color of the graphical touchpoint representation.



Visualization

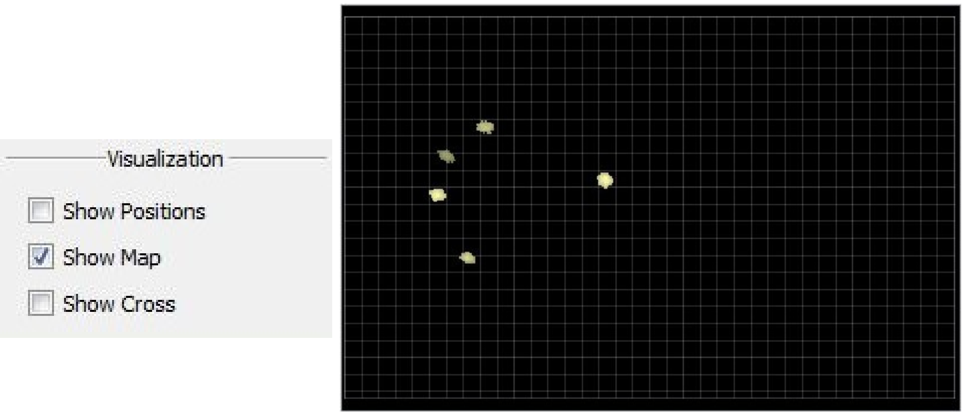
☑ Show Positions

☐ Show Map

☐ Show Cross

Touch Map: „Show Positions“

Initially „Show Positions“ is selected for visualization. In that mode each touch point is rendered at the exact coordinates and the size that is reported trough the various protocol options.



Visualization

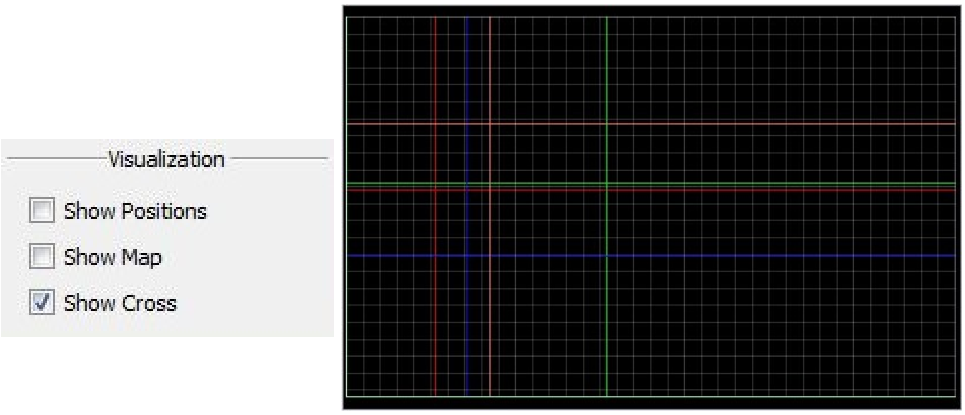
☐ Show Positions

☑ Show Map

☐ Show Cross

Touch Map: „Show Map“

When only the „Show Map” visualization option is selected, the raw map image before detection of touch points is shown.



Visualization

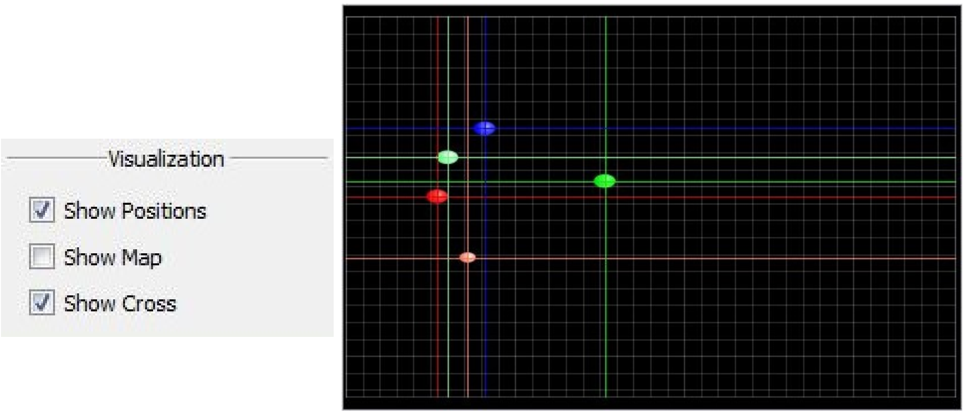
□ Show Positions

□ Show Map

☑ Show Cross

Touch Map: „Show Cross“

When only „Show Cross“ is selected, a cross is drawn at each reported touch points center position.



Visualization

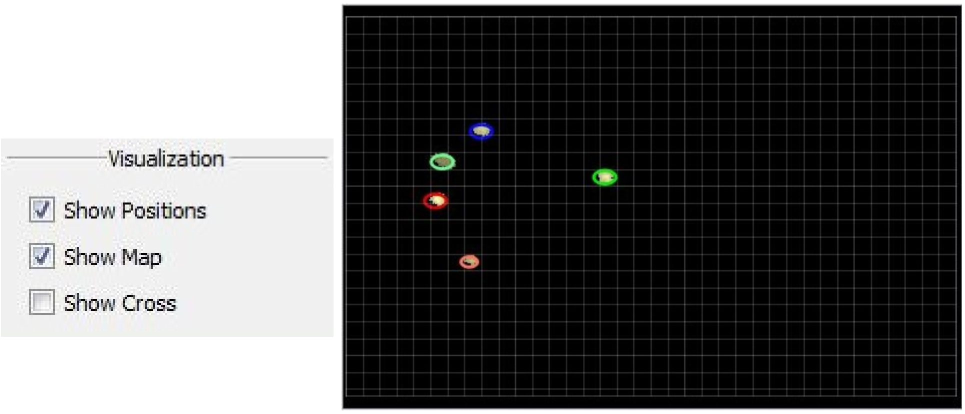
☑ Show Positions

□ Show Map

☑ Show Cross

Touch Map: „Show Positions“ + „Show Cross“

When both, „Show Positions and „Show Cross“, are selected, the size and coordinates of each reported Position are drawn as a filled ellipse with cross lines at its center.



Visualization

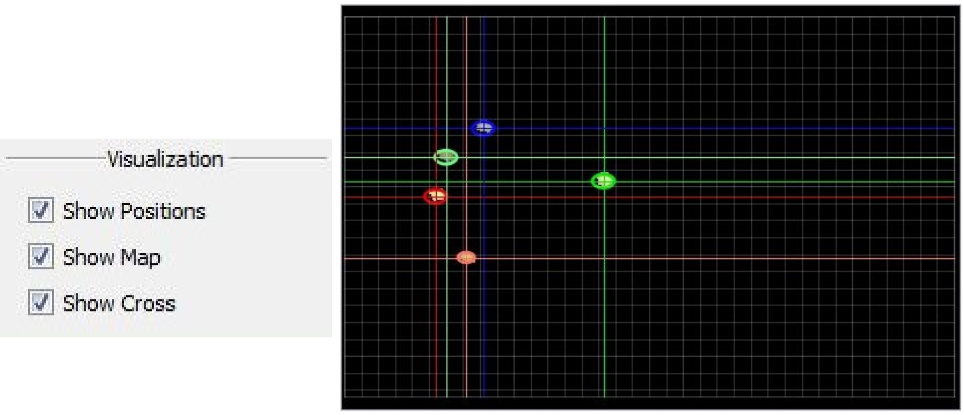
☑ Show Positions

☑ Show Map

□ Show Cross

Touch Map: „Show Positions“ + „Show Map“

When both, „Show Positions and „Show Map“, are selected, the size and coordinates of each reported Position are drawn as a hollow ellipse with the raw map drawn respectively within.



Visualization

☑ Show Positions

☑ Show Map

☑ Show Cross

Touch Map: „Show Positions“+„Show Map“+„Show Cross“

Finally, when all three options are selected simultaneously, all elements are drawn together. The big difference is now, that the position ellipses are still drawn at the reported coordinates. While the crosses are drawn at the raw positions before any smoothing or assistance functions. So this mode allows to evaluate the effects of the assistance functionality.

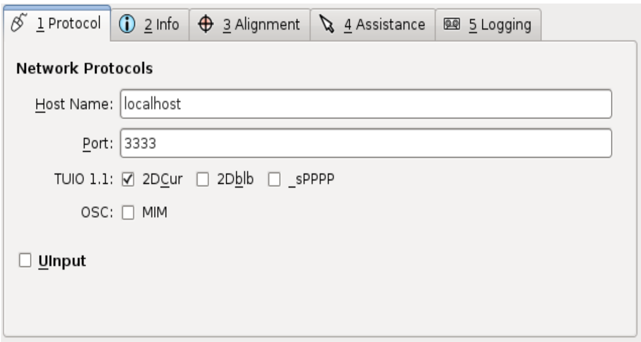
The visualization uses the settings from the simple alignment functions, with small exception that „Exchange X/Y“ is not used for raw map drawing.
But the „4-point Alignment“ transformation is completely ignored by touch area visualization.

4.3.3 Control Center Tabs

The upper right hand side contains a number of elements to gather information about the attached device. Most changes are allowed in Admin-Mode only.

4.3.3.1 Protocol

The way how detected touch points are reported to the operating system is controlled by the elements of this tab.



Network Protocols

Host Name: localhost

Port: 3333

TUIO 1.1: ☑ 2DCur ☐ 2Dblb ☐ _sPPPP

OSC: ☐ MIM

☐ UInput

The available protocols can be divided into 2 sections: Network based and uinput driver based.

4.3.3.1.1 Network Protocols

Network based protocols use UDP network packets to report detected touch points. The packet format is according to TUIO 1.1 protocol as described on this web-page: http://www.tuio.org/?specification.
Three TUIO-profiles are available:

- 2DCur, where only the center of the detected positions are reported
- 2Dblb, where the detected positions are reported as ellipses
- _sPPPP is a user defined protocol consisting of the session ID, the X-, Y-, and Z-accelerometer coordinate and the temperature as floating point values

The MIM protocol uses another packet format based on the OSC protocol standard. It is provided for compatibility of dreaMTouch devices with older multi-touch tables.
All these UDP packets have to be sent to a certain host at a certain port. These host/port settings can be made here, too.

4.3.3.1.2 TUIO-Protocol Definitions

When citmuto03drv is executed and one of the TUIO protocol boxes are checked UDP packets are send to a certain IP address (default: localhost) and port (default: 3333). These packets comply with the selected TUIO profile. Following profiles are supported:

/tuio/2Dcur: X/Y co-ordinates, velocity, and acceleration of touch points.

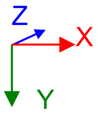
/tuio/25Dblb: X/Y co-ordinates, size, velocity, and acceleration of touch points

/tuio/_sPPPP: User defined TUIO Message with accelerometer and temperature information:

s	Session ID	touch device number*33, usually 0	int32
P	Temperature	[°C]	float32
P	Accelerometer X	[m/s²]	float32
P	Accelerometer Y	[m/s²]	float32
P	Accelerometer Z	[m/s²]	float32

dreaMTouch uses following co-ordinate system for accelerometer data, assuming touch is operated in standard orientation, that means display in landscape orientation on a vertical wall:

X	positive = right	negative = left
Y	positive = down	negative = up
Z	positive = back	negative = front



4.3.3.1.3 uinput

Citmuto03drv can directly send multi-touch events to the Linux kernel via the user level input system (uinput) interface. These events are processed by the kernel and fed to the event device (evdev). The X.org evdev driver then picks up the pre-processed evdev events and translates them to X11 input events.

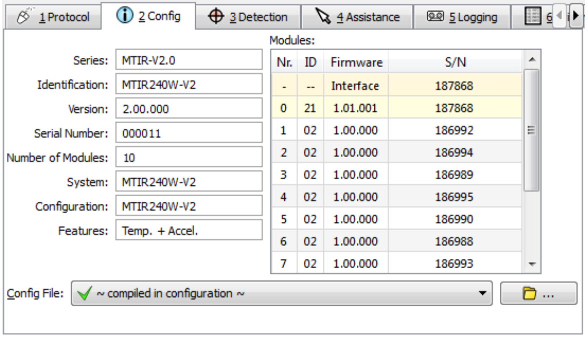
If Uinput is selected as output protocol in citmuto03gui, all touch-events will be sent to the kernel via the /dev/input/uinput device node.

Additionally both a /dev/input/eventXX (with XX being an integer number, depending on attached input devices) and a /dev/input/dreamtouch (being a symbolic link to /dev/input/eventXX) device node will be created, to which the X.org evdev driver (on Debian it does this automatically) or other software can be bound.

To make full use of this feature, a recent Xorg stack with multitouch input support and a graphical toolkit (Qt/GTK) is needed, which unfortunately isn`t provided by Debian “stable” yet.
On older systems not providing recent enough Xorg stack, citmuto03drv emulates a single-touchscreen, however your applications can pull multi-touch event information from /dev/input/dreamtouch.

4.3.3.2 Config

Detailed information about the attached dreaMTouch device is provided in this dialog tab. There is also the opportunity to load and select a configuration file for the touch. This option is especially useful for xtreMTouch®-systems.



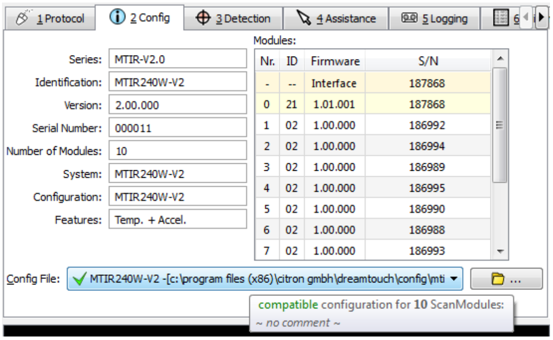
Series: MTIR-V2.0
Identification: MTIR240W-V2
Version: 2.00.000
Serial Number: 000011
Number of Modules: 10
System: MTIR240W-V2
Configuration: MTIR240W-V2
Features: Temp. + Accel.

Modules:

Nr.	ID	Firmware	S/N
-	--	Interface	187868
0	21	1.01.001	187868
1	02	1.00.000	186992
2	02	1.00.000	186994
3	02	1.00.000	186989
4	02	1.00.000	186995
5	02	1.00.000	186990
6	02	1.00.000	186988
7	02	1.00.000	186993

Config File: ✓ ~ compiled in configuration ~

touch using fixed configuration



Series: MTIR-V2.0
Identification: MTIR240W-V2
Version: 2.00.000
Serial Number: 000011
Number of Modules: 10
System: MTIR240W-V2
Configuration: MTIR240W-V2
Features: Temp. + Accel.

Modules:

Nr.	ID	Firmware	S/N
-	--	Interface	187868
0	21	1.01.001	187868
1	02	1.00.000	186992
2	02	1.00.000	186994
3	02	1.00.000	186989
4	02	1.00.000	186995
5	02	1.00.000	186990
6	02	1.00.000	186988
7	02	1.00.000	186993

Config File: ✓ MTIR240W-V2 -[c:\program files (x86)\citron gmbh\dreamtouch\config\mti

compatible configuration for 10 ScanModules:
~ no comment ~

using configuration file

(pop-up window shows information about config-file)

4.3.3.2.1 Information

The table below gives a description of each information element:

Element	Description
Series	The dreaMTouch hardware generation. The most current is MTIR-V2.0.
Identification	The identification string as stored in the dreaMTouch device.
Version	The detailed dreaMTouch device version.
Serial Number	The dreaMTouch device serial number.
Number of Modules	Each dreaMTouch is build of several so called ScanModules, depending on its size.
System	xtreMTouch only: If an installation is build of more than one device the name of the entire system is given here.
Configuration	xtreMTouch only: The name of the configuration that this device uses to find its place within the system.
Features	The special feature of the attached dreaMTouch device.
Modules	Detailed information about each ScanModule.
Config File	Absolut path to used configuration file for that touch; or „*compiled in configuration*“, if touch configuration compiled into driver core is used.

4.3.3.2.2 Config File

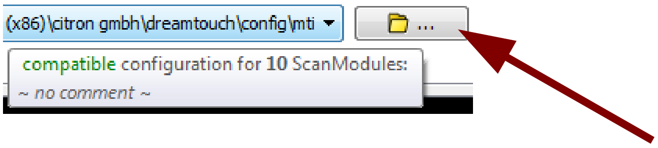
The driver needs some kind of description of the attached touch device, called “configuration”, in order to be able to calculate the geometry and detect touch points correctly.
There are two types of configurations:

1. Fixed, compiled into the driver
2. Loadable from an external configuration file (usually with extension “.dcf”)

External configuration files are created with the aid of the xtreMConfig configurator utility (available as download from our website) and are mostly useful with xtreMTouch systems. On dreaMTouch touch devices you shouldn't need an external configuration file as configurations for those models are compiled into the driver.

A configuration from an external configuration file is compatible to a certain touch device, if the number of described ScanModules in the configuration and in the connected touch device are equal. In this case a green “OK” checkmark is displayed next to the configuration name. Otherwise a red “X” cross is used to visualize the incompatibility.

The xtreMConfig configurator utility version 1.00.001 (or newer) allows to attach a comment to a config-file. This comment together with some basic information about the configuration is shown as a pop-up window when the mouse hovers over a configuration name in the list.



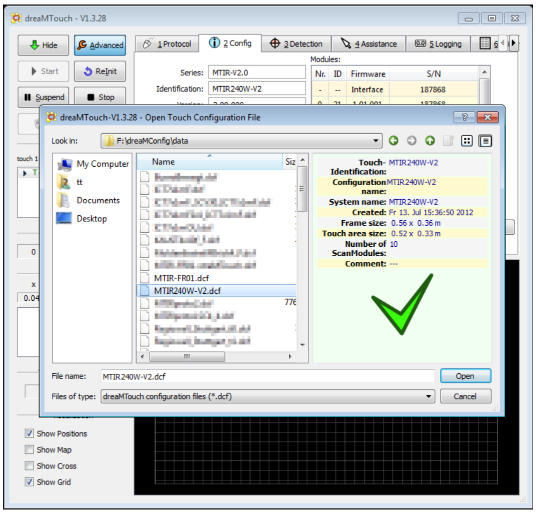
(x86)\citron gmbh\dreamtouch\config\mti

compatible configuration for 10 ScanModules:

~ no comment ~

To add a new configuration to the list, click on the folder symbol on the right hand side of the configuration selection box.

After a small delay a dialog pops up, that allows the selection of the configuration file.



dreaMTouch - V1.3.28

Series: MTIR-V2.0

Identification: MTIR240W-V2

Nr.	ID	Firmware	S/N
-	--	Interface	187868

dreaMTouch-V1.3.28 - Open Touch Configuration File

Look in: F:\dreaMConfig\data

MTIR-FR01.dcf

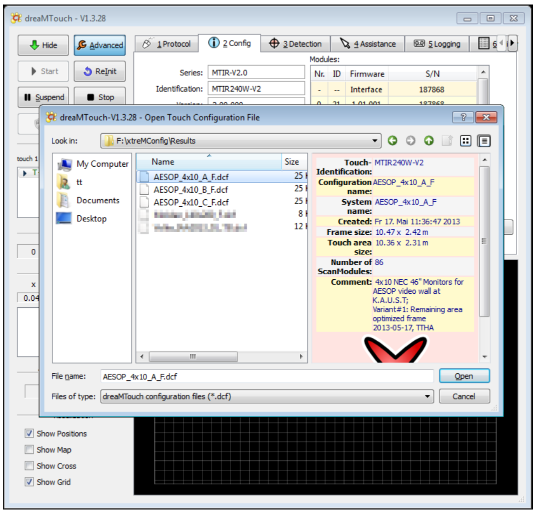
MTIR240W-V2.dcf

Touch-Identification: MTIR240W-V2
Configuration name: MTIR240W-V2
System name: MTIR240W-V2
Created: Fr 13. Jul 15:36:50 2012
Frame size: 0.56 x 0.36 m
Touch area size: 0.52 x 0.33 m
Number of ScanModules: 10
Comment: ---

File name: MTIR240W-V2.dcf

Files of type: dreaMTouch configuration files (*.dcf)

☑ Show Positions
□ Show Map
□ Show Cross
☑ Show Grid



dreaMTouch - V1.3.28

Series: MTIR-V2.0

Identification: MTIR240W-V2

Nr.	ID	Firmware	S/N
-	--	Interface	187868

dreaMTouch-V1.3.28 - Open Touch Configuration File

Look in: F:\xtreMConfig\Results

Name	Size
AESOP_4x10_A_F.dcf	25
AESOP_4x10_B_F.dcf	25
AESOP_4x10_C_F.dcf	25

Touch-Identification: MTIR240W-V2
Configuration name: AESOP_4x10_A_F
System name: AESOP_4x10_A_F
Created: Fr 17. Mai 11:36:47 2013
Frame size: 10.47 x 2.42 m
Touch area size: 10.36 x 2.31 m
Number of ScanModules: 86
Comment: 4x10 NEC 46" Monitors for AESOP video wall at K.A.U.S.T; Variant#1: Remaining area optimized frame 2013-05-17, TTHA

File name: AESOP_4x10_A_F.dcf

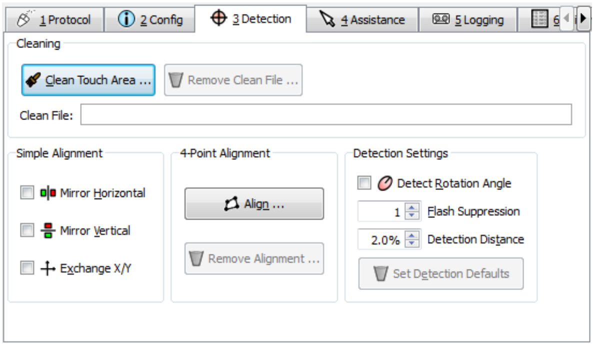
Files of type: dreaMTouch configuration files (*.dcf)

☑ Show Positions
□ Show Map
□ Show Cross
☑ Show Grid

Again, some information about the selected configuration is shown and compatible and incompatible configurations are indicated by a green checkmark or a red cross respectively. When the dialog is closed by a click on “*Open*” the selected configuration gets copied to the local configuration directory of the driver and the driver tries to select this configuration. A click on “Cancel” closes the dialog without any adjustments.
If the configuration file is compatible with the connected touch, the selection succeeds and the new configuration is used for that touch device from now on.
If the configuration is not compatible, it gets copied anyway, but it will not be used. Instead it appears in the list of selectable configurations for later use.

4.3.3.3 Detection

This dialog-tab holds some controls to optimize the touch point detection behavior of the dreaMTouch device for the user's application.



□ Mirror Horizontal
□ Mirror Vertical
□ Exchange X/Y
□ Detect Rotation Angle
1 Flash Suppression
2.0% Detection Distance

4.3.3.3.1 Cleaning

It is recommended, though not required, that each new dreaMTouch should be „cleaned” before first usage.

It is also useful if the bezel frame was damaged permanently on a single spot, so if you think the dreaMTouch is not working as proper as it was doing when it was new, use the “Clean..” function to mask out all IR beams that are not working anymore.

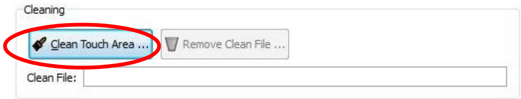
Another scenario when the „Clean Touch Area...“ function comes in handy, is if you have decided to set-up your dreaMTouch in such a way, that there is a permanent, static obstacle in the active touch area. To prevent an obstacle from creating a permanent touch point, click on the “Clean Touch Area…” button.

It is important to understand, that this “cleaning-information” is saved separately for each individual touch device. The devices are identified by their serial numbers. That means if you have successfully cleaned an external obstacle out of the dreaMTouch and you change that device, you have to re-clean.

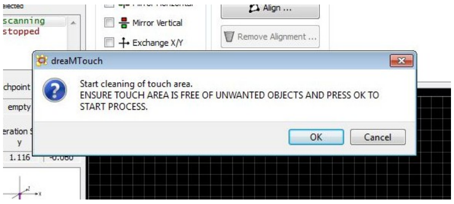
4.3.3.3.1.1 Cleaning-Process

The cleaning-process itself works the following way:

1) Click “Clean Touch Area...”

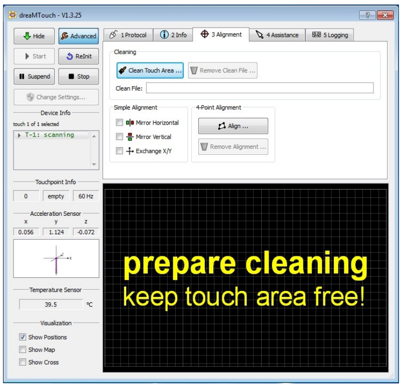


2) Read the warning message and click “OK” to proceed or “Cancel” to abort.



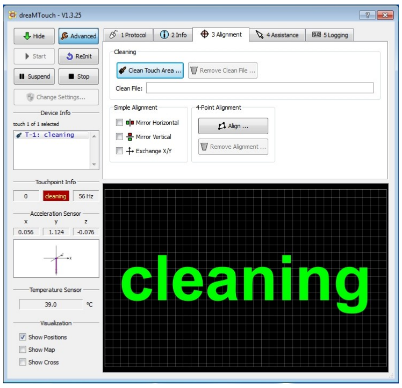
Start cleaning of touch area.
ENSURE TOUCH AREA IS FREE OF UNWANTED OBJECTS AND PRESS OK TO START PROCESS.

3) You have now about 2 seconds to move your finger out of the touch area before cleaning actually starts. During this time a yellow warning message is rendered to the touch map.



prepare cleaning
keep touch area free!

4) While touch cleaning is in progress do not touch the screen. Simply wait until the “Cleaning in progress” message disappears after some seconds.



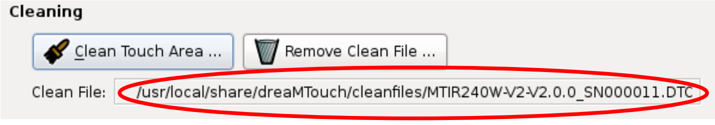
4.3.3.3.1.2 Clean File Location

If cleaning was successful a DTC-file (Dream Touch Cleaning) was saved to the common application data area of your computer so the information is kept persistent between driver restarts.

On Linux the standard cleanfile location is “/usr/local/share/citmuto03/cleanfiles”

The file name itself is assembled from the touch model, configuration name, version, and serial number. If the touch model name and the configuration name are identical, the configuration name is omitted.

For example an MTIR240W-V2 with serial number 11 and compiled-in configuration will have the file name “/usr/local/share/citmuto03/cleanfiles/MTIR240W-V2-V2.0.0-SN000011.DTC”
The actual clean file name and path is shown in the element “Clean File:” after a successful cleaning.

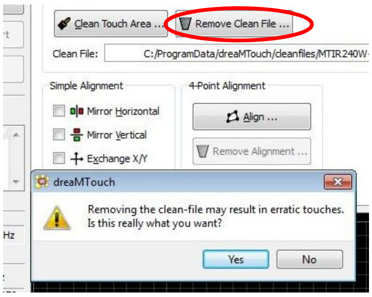


Clean File: /usr/local/share/dreaMTouch/cleanfiles/MTIR240W-V2-V2.0.0_SN000011.DTC

4.3.3.3.1.3 Removal of Cleaning-Information

If the user decides that the cleaning-information is not required any more it can be removed in to different ways.

- Restart the “Clean..” process with a completely empty touch area. The cleaning process always uses the basic scan matrix information provided with the driver firmware. Any previously generated cleaning information is ignored during the cleaning-process.
- Click the “Remove Clean File..” button and accept the warning message.

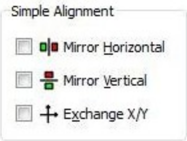


Clean File: C:/ProgramData/dreaMTouch/cleanfiles/MTIR240W-

Removing the clean-file may result in erratic touches.
Is this really what you want?

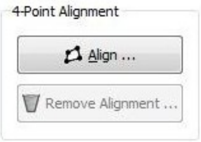
4.3.3.3.2 Simple Alignment

The simple alignment features are useful for fast and easy basic adjustments to the reported coordinates. These settings are ignored when a 4-point touch alignment is active.



4.3.3.3.3 4-Point Alignment

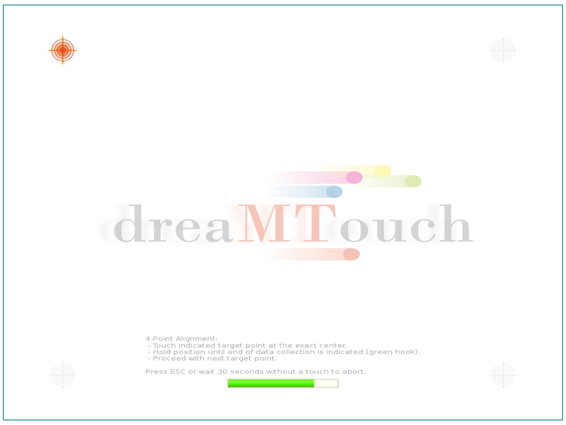
If there is a more complex alignment requirement, for example when a projector is used to create the touchable image, a true 4-point alignment is provided.



With 4 point alignment the touch area can be defined within an arbitrary shaped 4 sided polygon. That way - besides others - the following corrections can be made:

- arbitrary translation
- arbitrary rotation
- keystone correction

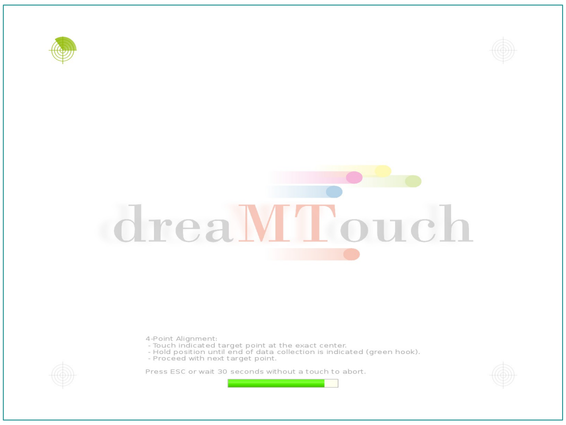
The 4-point alignment is started by a click on the „Align ...“ button. Now a full screen dialog comes up with 4 targets on it. If you want to abort 4-point alignment either wait 30 seconds (green bar shows time left) or press escape key.



4-Point Alignment:
- Touch indicated target point at the exact center.
- Hold position until end of data collection is indicated (green hook).
- Proceed with next target point.

Press ESC or wait 30 seconds without a touch to abort.

All four targets have to be touched in the order indicated. Which target has to be touched next is indicated by a red animation. In the image above this is the target in the top left corner of the screen.
As soon as the target gets touched its color changes from red to green. Keep your finger steady at the precise center of the target while data is collected.

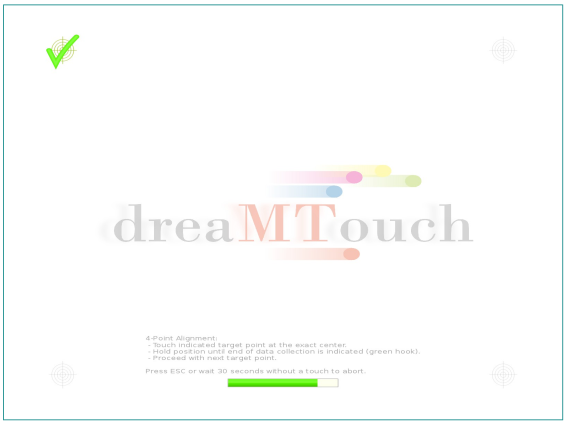


4-Point Alignment:
- Touch indicated target point at the exact center.
- Hold position until end of data collection is indicated (green hook).
- Proceed with next target point.

Press ESC or wait 30 seconds without a touch to abort.

If your finger moves, data collection is restarted as soon as the finger is steady again. The collection process is shown by a closing green circle animation. In the image above 1/3 of the data collection is done.

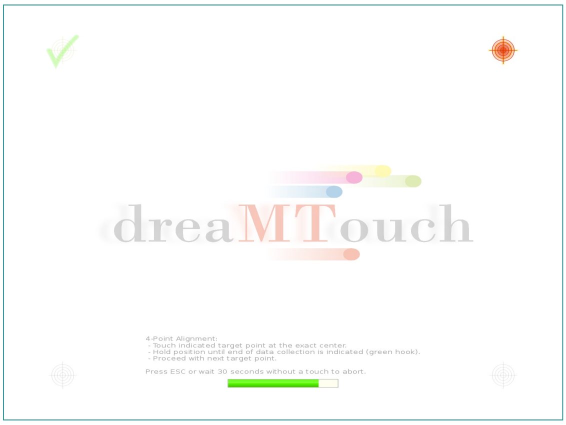
Eventually all data for this point is collected. That state is indicated by a green „OK-hook“.



4-Point Alignment:
- Touch indicated target point at the exact center.
- Hold position until end of data collection is indicated (green hook).
- Proceed with next target point.

Press ESC or wait 30 seconds without a touch to abort.

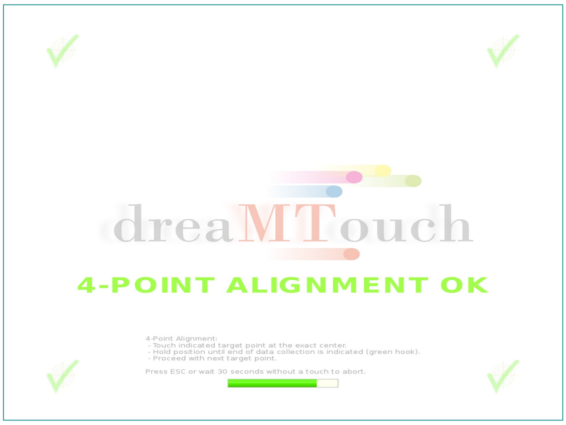
Now it is time to release the touch point on the monitor and proceed to the next target.



4-Point Alignment:
- Touch indicated target point at the exact center.
- Hold position until end of data collection is indicated (green hook).
- Proceed with next target point.

Press ESC or wait 30 seconds without a touch to abort.

Repeat these steps until data collection for all 4 targets is completed. This state is indicated by the message „4-POINT ALIGNMENT OK“.



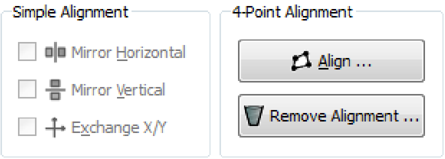
4-POINT ALIGNMENT OK

4-Point Alignment:
- Touch indicated target point at the exact center.
- Hold position until end of data collection is indicated (green hook).
- Proceed with next target point.

Press ESC or wait 30 seconds without a touch to abort.

As soon as the touch area is untouched again the full-screen window is closed and 4-point alignment is finished.

After a successful 4-point alignment the „Simple Alignment“ settings are ignored by the driver and consequently these settings are disabled in the Control Center.


Simple Alignment

□ Mirror Horizontal

□ Mirror Vertical

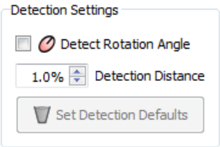
□ Exchange X/Y

4-Point Alignment

To remove a 4-point alignment simply click on „Remove Alignment ...“ and accept the up-popping warning by a click on the „Yes“ button.

4.3.3.3.4 Detection Settings

Usually the default detection settings are fine for most users. But some advanced dreaMTouch operators need to tweak some settings of the detection algorithm. This is possible using the elements of the „Detection Settings“-group on the „Alignment“ dialogue-tab.

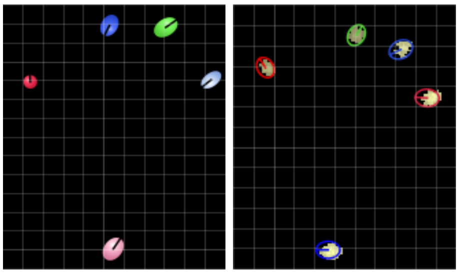

Detection Settings

□ Detect Rotation Angle

1.0% Detection Distance

4.3.3.3.4.1 Detect Rotation Angle

Since detected touch-point has an elliptical shape its rotation angle can be detected. This is an experimental feature in version 1.3.26 of the dreaMTouch suite. So it is switched off by default. It can be enabled by checking the box at „Detect Rotation Angle“. The detected angle is visualized by a line in the touch visualization area.


Show Positions **Show Positions+Map**

The ellipse rotation angle is reported to applications using TUIO protocol with 2Dblb profile, only.

4.3.3.3.4.2 Flash Suppression

Since the dreaMTouch uses short infrared light pulses to detect touch events, it can be disturbed by strong photoflashes.

To improve the photoflash immunity, a special algorithm is implemented that suppresses touch events if they do not have a typical signature.

Nevertheless there is a trade-off between flash light suppression and minimum touch-point diameter and distance. If your application has no problems with flash light you can switch the flash light suppression off completely, or leave it at the default level “1”.

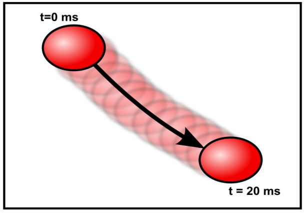
But if you need stronger flash light suppression, and a lower touch resolution is ok for your application, you can increase the level up to a value of 5.

4.3.3.3.4.3 Detection Distance

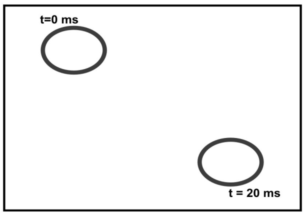
To understand what this setting does a little knowledge about how a dreaMTouch works internally is required.

The dreaMTouch scan the touch area in regular intervals, lets say for example every 20 ms. Every scan represents a snapshot of the touch points at a certain, discrete time. Between scans the touch cannot „see“ what happens within the touch area. It is important to understand this. The dreaMTouch does not has a continuous knowledge of what happens inside the touch area. Only discrete snapshots are taken, at (very) short time intervals.

First assume a (fast) moving touch point:

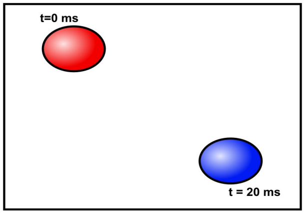


User moving finger

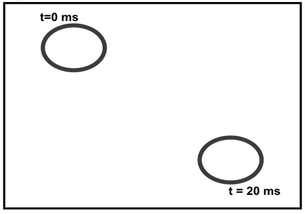


What dreaMTouch detects

Now assume a user making two clicks at two different points at a short time interval:



User touching different spots



What dreaMTouch detects

As can be seen there is no difference for the dreaMTouch in the detected pattern.

But the user expects dreaMTouch to assign the same ID in the first case and different IDs in the second case.

As a solution to this conflict the driver assumes a certain maximum speed for touch points. If a touch point moves more than a given distance between two scans, a new ID is assigned. Otherwise the same ID is used. This parameter is called „*Detection Distance*“. It is measured in percentage of touch area diagonal. The default value is 2.0 %, which is best for most applications.

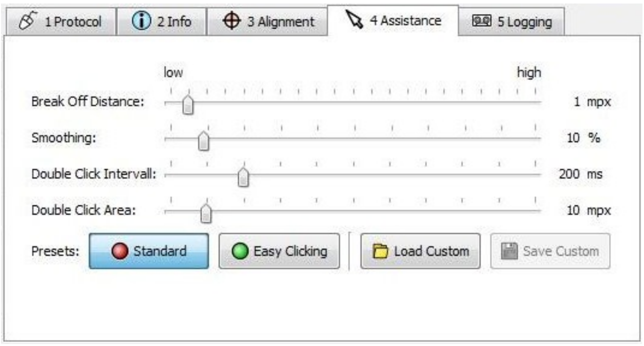
But sometimes a different value is required. Larger values allow higher movement speeds, but will assign the same ID the close by touch points.

4.3.3.3.4.4 Set Detection Defaults

To reset all the detection setting to their defaults, simply click this button. If this button is disabled, or greyed-out, defaults are set already.

4.3.3.4 Assistance

With a touch device it is sometime difficult to perform a precise double click because of input coordinate jitter.



Break Off Distance: 1 mpx
Smoothing: 10 %
Double Click Intervall: 200 ms
Double Click Area: 10 mpx

To help with making these often required standard gestures the dreaMTouch driver provides some assistance functions.

Two predefined settings called „Standard“ and „Easy Clicking“ are available. Furthermore a user defined setting can be saved and recalled later.

Each predefined setting can be recalled by a click on the appropriate button.

If these presets are not enough some fine tuning is possible using the 4 sliders. The unit „mpx“ means in this case „Touch Map Pixel“.

„Break Off Distance“

This parameter defines the amount of map pixels a touch point has to move, before the movement is actually reported. As soon as the touch point is moving, it follows the finger immediately.

„Smoothing“

Touch points coordinates are reported after passing an averaging filter. The amount of averaging is controlled by this parameter.

„Double Click Interval“ and „Double Click Area“

These two parameters work together to allow a precise double clicking. When a touch point is gone and a new touch-point appears within the „Double Click Interval“. And that new touch point is within a circle with diameter „Double Click Area“-map pixels the coordinates of the last touch point are reported instead of the actually detected coordinates.

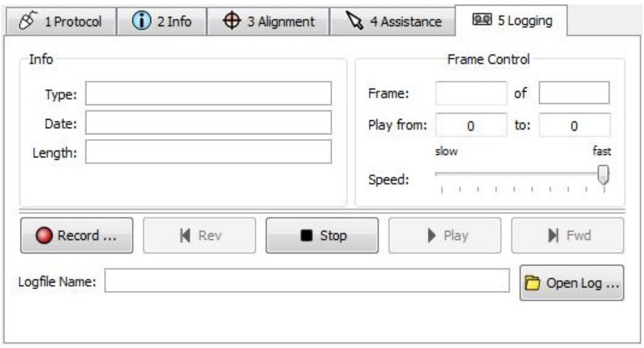
Consider the following example:

Double Click Interval = 200 ms
Double Click Area = 10 mpx

A touch point at position (125, 80) is gone. A new touch point is detected 100 ms later at the position (127, 92). That new touch point is detected within the 200 ms interval. The distance between the old touch point and the new one is 8.2 mpx and therefore withing the 10 mpx area. In that case the new touch point is reported at position (125, 80). But if the new touch point would have been detected later than 200 ms after the old one was got, its actual coordinates (127,92) would have been reported.

4.3.3.5 Logging

To assist support when a problem with a dreaMTouch device occurs a logging function is implemented in the driver.



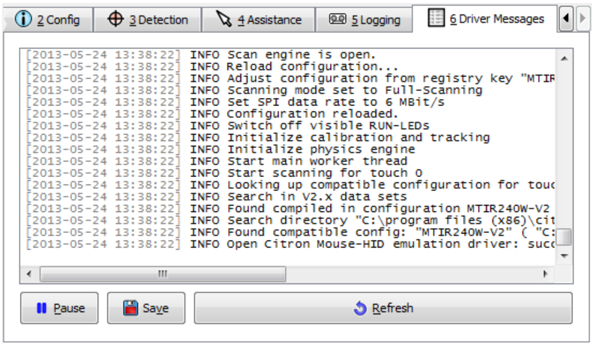
A dreaMTouch user would click on the „Record“ button in that case and record the problematic gestures. The location for the log file is selected in a dialog right after the click on „Record“. During the recording information about the number of recorded frames and the duration of the recording are permanently updated. The recording ends by a click on the „Stop“ button.
Such a log file can then be sent by email to the dreaMTouch support for further evaluation. Beside the recorded touch events some information about the used computer are stored in that log-file, too. In detail these additional information are:

- Exact name and version of the operating system
- Type and speed of the CPU
- Size of system memory
- Graphics card information and screen resolution

All these pieces of information help the support desk to identify a problem faster. The log file replay and cropping functions should be required by support personal only. So a detailed description is not given here.

4.3.3.6 Driver Messages

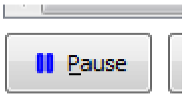
For even more technical support the detailed driver messages are logged to a file and shown at this dialog-tab.



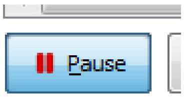
```
[2013-05-24 13:38:22] INFO Scan engine is open.
[2013-05-24 13:38:22] INFO Reload configuration...
[2013-05-24 13:38:22] INFO Adjust configuration from registry key "MTIR
[2013-05-24 13:38:22] INFO Scanning mode set to Full-Scanning
[2013-05-24 13:38:22] INFO Set SPI data rate to 6 MBit/s
[2013-05-24 13:38:22] INFO Configuration reloaded.
[2013-05-24 13:38:22] INFO Switch off visible RUN-LEDs
[2013-05-24 13:38:22] INFO Initialize calibration and tracking
[2013-05-24 13:38:22] INFO Initialize physics engine
[2013-05-24 13:38:22] INFO Start main worker thread
[2013-05-24 13:38:22] INFO Start scanning for touch 0
[2013-05-24 13:38:22] INFO Looking up compatible configuration for touc
[2013-05-24 13:38:22] INFO Search in V2.x data sets
[2013-05-24 13:38:22] INFO Found compiled in configuration MTIR240W-V2
[2013-05-24 13:38:22] INFO Search directory "C:\program files (x86)\cit
[2013-05-24 13:38:22] INFO Found compatible config: "MTIR240W-V2" ( "C:
[2013-05-24 13:38:22] INFO Open Citron Mouse-HID emulation driver: succ
```

Each line is tagged with a date and time so long term logs can be evaluated easily. Special events such as errors or warnings and reboots are colorized or printed with a bold font. User settings made via control-center are indicated with an italics font.

The log window automatically scrolls down to the latest log entry. This makes it difficult to examine older entries when there are frequent log events, for example in case of a touch error. To solve this, the “Pause” button can be pressed. To return to automatic scroll, simply press the pause button again.

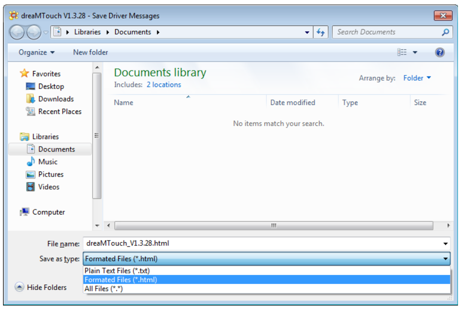


Scroll to last line



Scrolling paused

To send the log via Email it can be saved to a file. After a clock on the “Save...” button this file selection dialog appears:



File name: dreaMTouch_V1.3.28.html

Save as type: Formated Files (*.html)

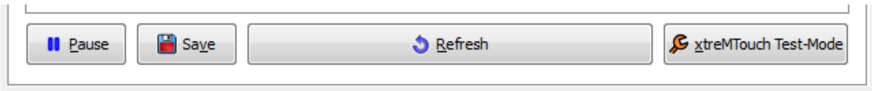
Plain Text Files (*.txt)
Formated Files (*.html)
All Files (*.*)

Two different formats for the saved file are available:

- “Plain Text”, for smallest pure text files
- “Formatted”, for fully colorized and styled HTML-files

Sometimes the log does not get refreshed automatically. In this case a click on the “Refresh” button forces an update of the log.

The log window comes in handy when debugging an xtreMTouch system for errors in the electrical interconnection. For further assistance an additional button “xtreMTouch Test-Mode” is displayed on this dialog tab, if a xtreMTouch is detected by the driver.



A click on this button activates the “xtreMTouch Test-Mode” as described in the xtreMTouch User's Manual. In this test-mode some of the green LEDs, located at the xtreMBrick connectors, start blinking. Details can be found in the xtreMTouch User's Manual.

5 Embedded Web Server

Since version 1.0.2.5 there is a web server embedded into citmuto, which provides status information on dreaMTouch hardware and the citmuto03 driver.

5.1 Enabling/Disabling the Web Server

The embedded web-server can be enabled/disabled and configured via the *\/etc/xdg/Citron\ GmbH/dreaMTouch.conf* configuration file.

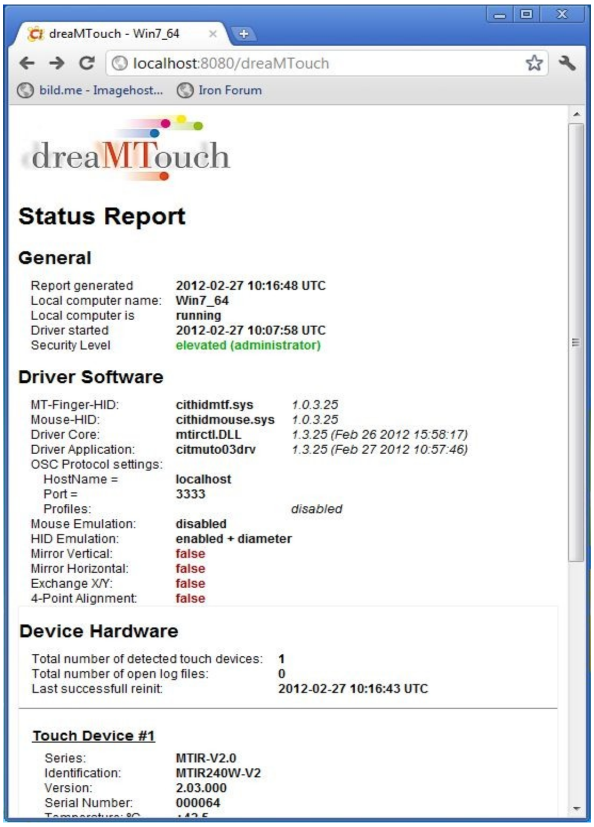
Within the *[General]* section a key named “*WebServer*” with the value “*1*” has to be added in order to enable the web-server. A value of “*0*” or a missing entry disables the web server.

The driver daemon reads these registry settings at start-up. Therefore the citmuto03drv daemon needs to be restarted to take any changes into effect.

In a Root-Terminal type:

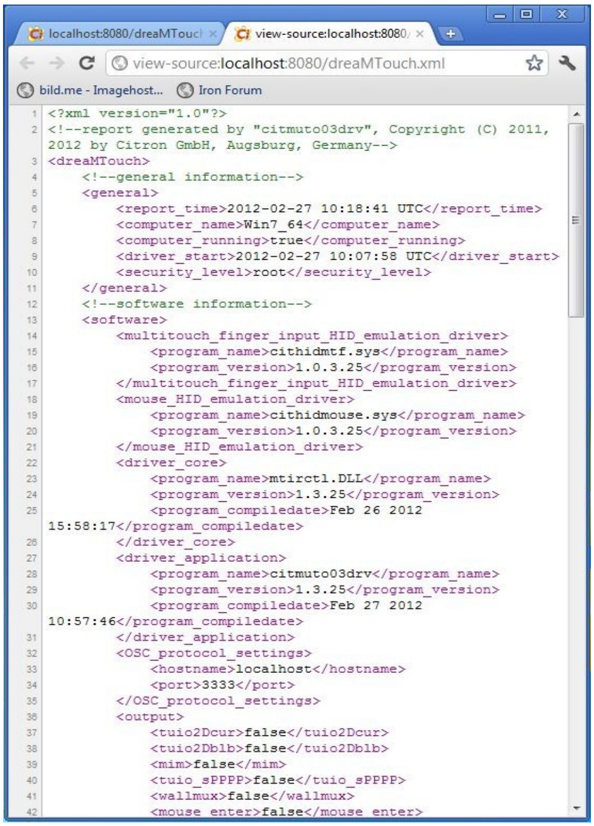
```
/etc/init.d/citmuto03 restart
```

The web-server can be accessed for a first test by directing a browser of your choice to http://localhost:8080/dreaMTouch.html (URL is case sensitive, please take care for the upper-/lower case letters!)



```
Status Report

General
  Report generated          2012-02-27 10:16:48 UTC
  Local computer name:      Win7_64
  Local computer is         running
  Driver started            2012-02-27 10:07:58 UTC
  Security Level            elevated (administrator)

Driver Software
  MT-Finger-HID:            cithidmtf.sys     1.0.3.25
  Mouse-HID:                cithidmouse.sys   1.0.3.25
  Driver Core:              mtirctl.DLL       1.3.25 (Feb 26 2012 15:58:17)
  Driver Application:       citmuto03drv      1.3.25 (Feb 27 2012 10:57:46)
  OSC Protocol settings:
    HostName =              localhost
    Port =                  3333
    Profiles:                                 disabled
  Mouse Emulation:          disabled
  HID Emulation:            enabled + diameter
  Mirror Vertical:          false
  Mirror Horizontal:        false
  Exchange X/Y:             false
  4-Point Alignment:        false

Device Hardware
  Total number of detected touch devices:   1
  Total number of open log files:           0
  Last successfull reinit:                  2012-02-27 10:16:43 UTC

  Touch Device #1
    Series:                 MTIR-V2.0
    Identification:         MTIR240W-V2
    Version:                2.03.000
    Serial Number:          000064
```

example HTML Web Page


localhost:8080/dreaMTouc
view-source:localhost:8080/
view-source:localhost:8080/dreaMTouch.xml
bild.me - Imagehost...
Iron Forum

```
<?xml version="1.0"?>
<!--report generated by "citmuto03drv", Copyright (C) 2011,
2012 by Citron GmbH, Augsburg, Germany-->
<dreaMTouch>
    <!--general information-->
    <general>
        <report_time>2012-02-27 10:18:41 UTC</report_time>
        <computer_name>Win7_64</computer_name>
        <computer_running>true</computer_running>
        <driver_start>2012-02-27 10:07:58 UTC</driver_start>
        <security_level>root</security_level>
    </general>
    <!--software information-->
    <software>
        <multitouch_finger_input_HID_emulation_driver>
            <program_name>cithidmtf.sys</program_name>
            <program_version>1.0.3.25</program_version>
        </multitouch_finger_input_HID_emulation_driver>
        <mouse_HID_emulation_driver>
            <program_name>cithidmouse.sys</program_name>
            <program_version>1.0.3.25</program_version>
        </mouse_HID_emulation_driver>
        <driver_core>
            <program_name>mtirctl.DLL</program_name>
            <program_version>1.3.25</program_version>
            <program_compiledate>Feb 26 2012
15:58:17</program_compiledate>
        </driver_core>
        <driver_application>
            <program_name>citmuto03drv</program_name>
            <program_version>1.3.25</program_version>
            <program_compiledate>Feb 27 2012
10:57:46</program_compiledate>
        </driver_application>
        <OSC_protocol_settings>
            <hostname>localhost</hostname>
            <port>3333</port>
        </OSC_protocol_settings>
        <output>
            <tuio2Dcur>false</tuio2Dcur>
            <tuio2Dblb>false</tuio2Dblb>
            <mim>false</mim>
            <tuio_sPPPP>false</tuio_sPPPP>
            <wallmux>false</wallmux>
            <mouse_enter>false</mouse_enter>
```

example XML Status Report

5.2 Web Server Configuration

The appearance and behavior of the web server can be configured by setting additional configuration keys in the [httpd] section of the configuration file. All settings are read at citmuto03drv start only, so you have to exit and restart citmuto03drv before any change will come to effect.

Configuration Key	Type	Default	Description
WebRoot	STRING	„/var/www/ciwebs/dreaMTouch/“	Directory containing web-page resources
HttpPort	DWORD	8080	Port number where the web server is waiting for connection
MaxClients	DWORD	32	Maximum number of clients that can be served simultaneously
MaxReqPerConn	DWORD	99	Maximum number of requests that can be handled per connection
PageRefresh	DWORD	10	0 = no auto refresh of web page any other value will define the number of seconds between automatic web page refreshes

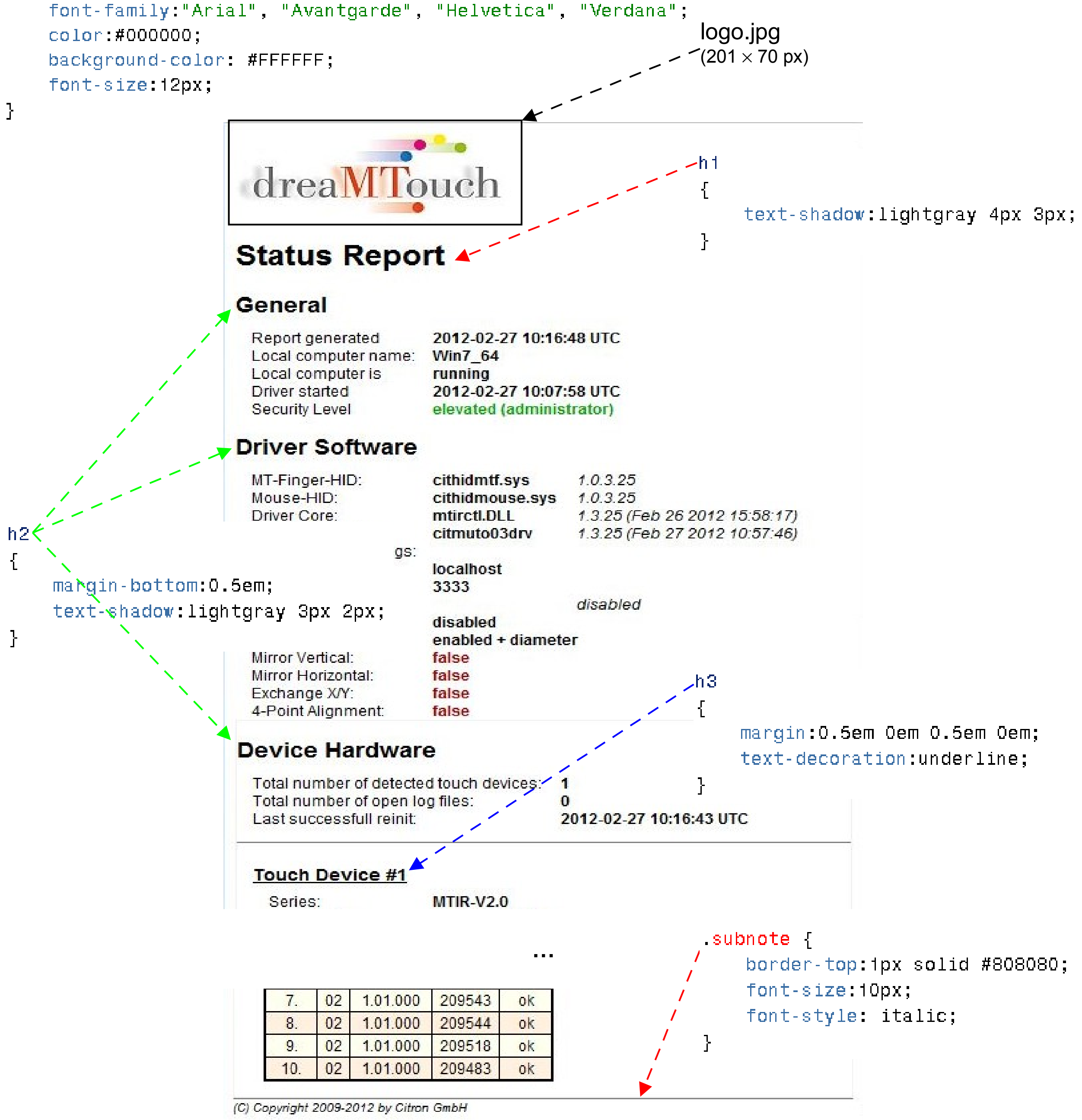
5.3 dreaMTouch Web Page Configuration (CSS)

The look of the web page is defined entirely using CSS. The file “default.css” located in the “WebRoot” directory (/var/www/ciwebs/dreaMTouch/) contains all recognized definitions.

5.3.1 Body

The main elements are formatted using body, header and div styles as illustrated in this image:

```
body {
     font-family:"Arial", "Avantgarde", "Helvetica", "Verdana";
     color:#000000;
     background-color: #FFFFFF;
     font-size:12px;
}
```



logo.jpg
(201 × 70 px)

```
h1
{
     text-shadow:lightgray 4px 3px;
}
```

Status Report

General

Report generated	2012-02-27 10:16:48 UTC
Local computer name:	Win7_64
Local computer is	running
Driver started	2012-02-27 10:07:58 UTC
Security Level	elevated (administrator)

Driver Software

MT-Finger-HID:	cithidmtf.sys	*1.0.3.25*
Mouse-HID:	cithidmouse.sys	*1.0.3.25*
Driver Core:	mtirctl.DLL	*1.3.25 (Feb 26 2012 15:58:17)*
	citmuto03drv	*1.3.25 (Feb 27 2012 10:57:46)*
gs:		
	localhost	
	3333	
		disabled
	disabled	
	enabled + diameter	
Mirror Vertical:	false	
Mirror Horizontal:	false	
Exchange X/Y:	false	
4-Point Alignment:	false	

```
h2
{
     margin-bottom:0.5em;
     text-shadow:lightgray 3px 2px;
}
```

Device Hardware

Total number of detected touch devices:	1
Total number of open log files:	0
Last successfull reinit:	2012-02-27 10:16:43 UTC

```
h3
{
     margin:0.5em 0em 0.5em 0em;
     text-decoration:underline;
}
```

Touch Device #1

Series:	MTIR-V2.0

...

7.	02	1.01.000	209543	ok
8.	02	1.01.000	209544	ok
9.	02	1.01.000	209518	ok
10.	02	1.01.000	209483	ok

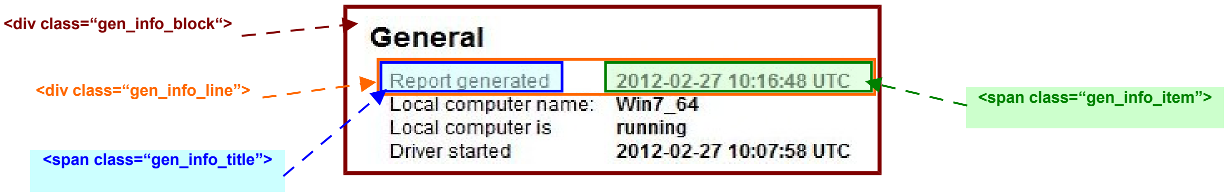
```
.subnote {
     border-top:1px solid #808080;
     font-size:10px;
     font-style: italic;
}
```

(C) Copyright 2009-2012 by Citron GmbH

The entire report is divided further using several <div> elements.

5.3.2 Section “General”

General information is given in a block with definition **<div class=”gen_info_block”> … </div>**. Each line within that block is formatted using **<div class=”gen_info_line”> … </div>** blocks. Individual parts of these lines are further classified using **<span class=”gen_info_title”> … </span>** and **<span class=”gen_info_item> … </span>** definitions.



General

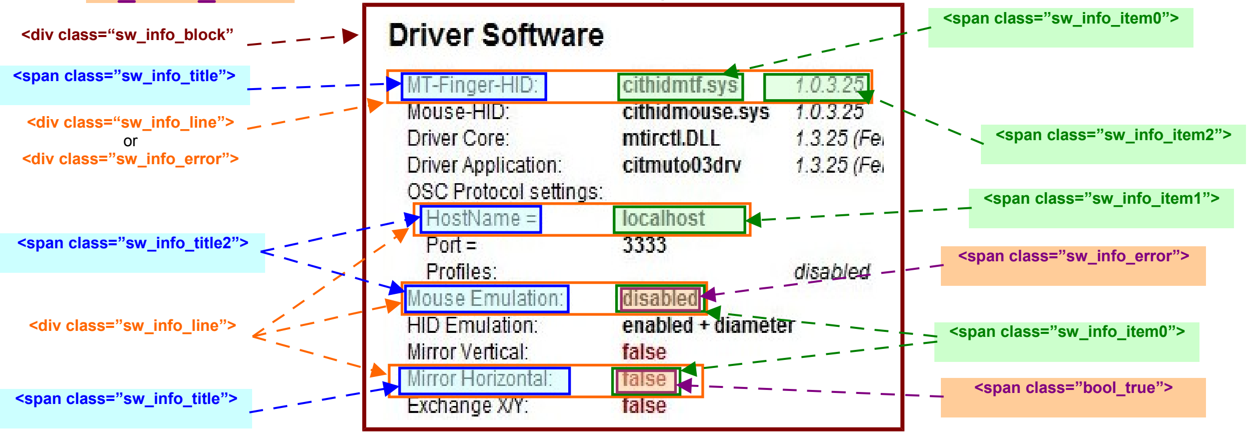
Report generated 2012-02-27 10:16:48 UTC
Local computer name: Win7_64
Local computer is running
Driver started 2012-02-27 10:07:58 UTC

The default definitions for these classes are given in “default.css” as follows:

```
/* generic information */
.gen_info_block
{
}

.gen_info_line
{
    margin-left:1em;
}

.gen_info_title
{
}

.gen_info_item
{
    font-weight:bold;
    position:absolute; left:13em;
}
```

5.3.3 Section “Driver Software”

Information regarding driver software is given in a block with definition
`<div class=”sw_info_block”> … </div>`.
Lines are formatted in two different ways:
In the regular, error free case each line is enclosed within `<div class=”sw_info_line”> … </div>` blocks. If the version of a dreaMTouch driver software module cannot be examined, the corresponding line is enclosed within `<div class=”sw_info_error”> … </div>` instead.
Individual parts of each line are further formatted using different combinations of spans of class “`sw_info_title`”, “`sw_info_title2`”, “`sw_info_item0`”, “`sw_info_item1`”, and “`sw_info_item2`”. Furthermore if there are Boolean values involved an additional span of class “`bool_true`” or “`bool_false`” is used. And to make it even more complicated there are also spans of class “`sw_info_error`” embedded if no transmission protocol is selected at all.



```
Driver Software
MT-Finger-HID:          cithidmtf.sys     1.0.3.25
Mouse-HID:              cithidmouse.sys   1.0.3.25
Driver Core:            mtirctl.DLL       1.3.25 (Fe
Driver Application:     citmuto03drv      1.3.25 (Fe
OSC Protocol settings:
  HostName =            localhost
  Port =                3333
  Profiles:                               disabled
Mouse Emulation:        disabled
HID Emulation:          enabled + diameter
Mirror Vertical:        false
Mirror Horizontal:      false
Exchange X/Y:           false
```

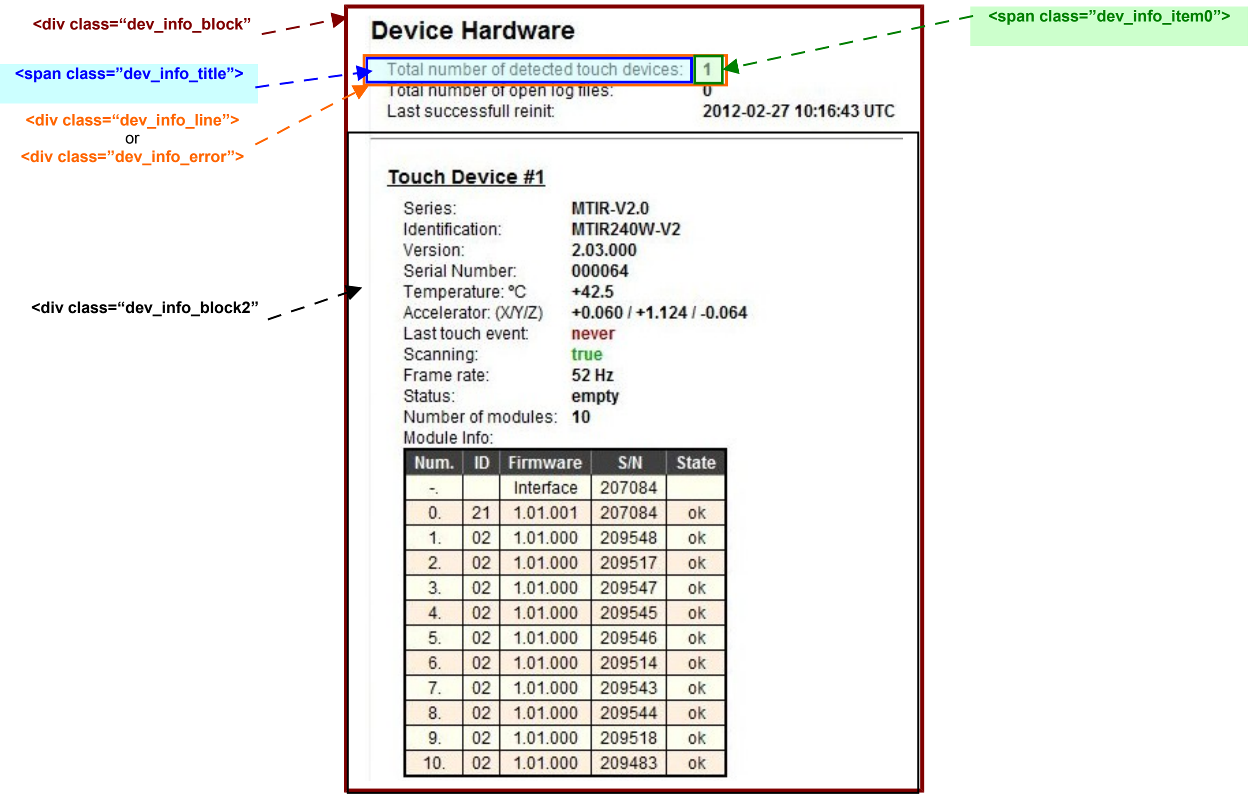
<div class=“sw_info_block”
<span class=”sw_info_title”>
<div class=“sw_info_line”>
or
<div class=”sw_info_error”>
<span class=”sw_info_title2”>
<div class=”sw_info_line”>
<span class=”sw_info_title”>
<span class=”sw_info_item0”>
<span class=”sw_info_item2”>
<span class=”sw_info_item1”>
<span class=”sw_info_error”>
<span class=”sw_info_item0”>
<span class=”bool_true”>

The default definitions for these classes are given in “default.css” as follows:

```
/* software information */
.sw_info_block
{
}

.sw_info_line
{
    margin-left:1em;
    font-weight:normal;
}

.sw_info_title
{
}

.sw_info_title2
{
    position:relative; left:1em;
}

.sw_info_item0
{
    font-weight:bold;
    position:absolute; left:13em;
}

.sw_info_item1
{
    font-weight:bold;
    position:absolute; left:13em;
}

.sw_info_item2
{
    font-weight:bold;
    position:relative; left:13em;
}

.sw_info_error
{
    font-weight:bold;
    background-color:#ff2020;
    color:#ffff00;
}
```

5.3.4 Section “Device Hardware”

Information regarding attached device hardware is given in a block with definition
`<div class=”dev_info_block”> … </div>`.
Lines are formatted in two different ways:
In the regular, error free case each line is enclosed within
`<div class=”dev_info_line”> … </div>` blocks. If no touch is connected at all, the error message line is enclosed within `<div class=”dev_info_error”> … </div>` instead.
Individual parts of each line are further formatted using different combinations of spans of class “`dev_info_title`”, and “`dev_info_item0`”.
Furthermore sub-blocks of `<div class=”dev_info_block2> … </div>` are embedded for each attached touch device. Details for those blocks follow in next chapter.


<div class="dev_info_block"
<span class="dev_info_title">
<div class="dev_info_line">
or
<div class="dev_info_error">
<div class="dev_info_block2"
<span class="dev_info_item0">

Device Hardware

Total number of detected touch devices: 1
Total number of open log files: 0
Last successfull reinit: 2012-02-27 10:16:43 UTC

Touch Device #1

Series: MTIR-V2.0
Identification: MTIR240W-V2
Version: 2.03.000
Serial Number: 000064
Temperature: °C +42.5
Accelerator: (X/Y/Z) +0.060 / +1.124 / -0.064
Last touch event: never
Scanning: true
Frame rate: 52 Hz
Status: empty
Number of modules: 10
Module Info:

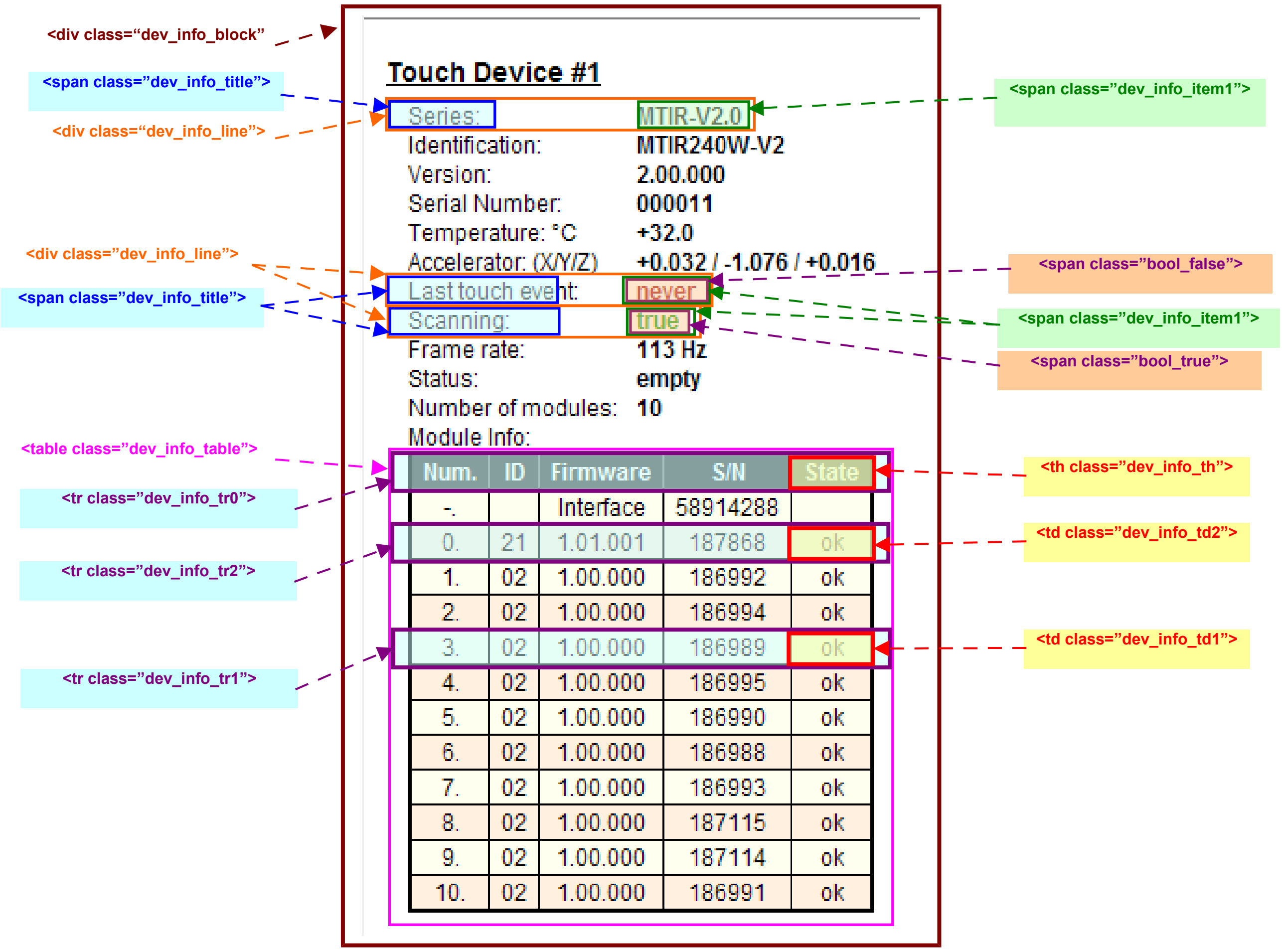
Num.	ID	Firmware	S/N	State
-.		Interface	207084	
0.	21	1.01.001	207084	ok
1.	02	1.01.000	209548	ok
2.	02	1.01.000	209517	ok
3.	02	1.01.000	209547	ok
4.	02	1.01.000	209545	ok
5.	02	1.01.000	209546	ok
6.	02	1.01.000	209514	ok
7.	02	1.01.000	209543	ok
8.	02	1.01.000	209544	ok
9.	02	1.01.000	209518	ok
10.	02	1.01.000	209483	ok

The default definitions for these classes are given in “default.css” as follows:

```
/* device information */
.dev_info_block
{
    border:1px solid #F0F0F0;
}

.dev_info_block2
{
    border-top:1px solid #808080;
    margin-top:1em;
    padding:1em;
}

.dev_info_line
{
    margin-left:1em;
    font-weight:normal;
}

.dev_info_title
{
}

.dev_info_item0
{
    font-weight:bold;
    position:absolute; left:21em;
}
```

5.3.5 Sub-Section “Touch Device”

Information regarding a specific touch device is enclosed in a block with definition
<div class=”dev_info_block2”> … </div>.
Lines are enclosed within <div class=”dev_info_line”> … </div> blocks.
Individual parts of each line are further formatted using different combinations of spans of class “dev_info_title” and “dev_info_item1”. Furthermore if there are Boolean values involved an additional span of class “bool_true” or “bool_false” is used. And to make it even more complicated there are also spans of class “dev_info_error” used if there is an “overflow” condition on the touch.
There is also a table containing information about individual modules of the touch. The table itself, its header, lines and rows are formatted using individual styles of class “dev_info_table”, “dev_info_tr0”, “dev_info_tr1”, “dev_info_tr2”, “dev_info_th”, “dev_info_td1”, and “dev_info_td2”


<div class=“dev_info_block”
<span class=”dev_info_title”>
<div class=“dev_info_line”>
<div class=”dev_info_line”>
<span class=”dev_info_title”>
<table class=”dev_info_table”>
<tr class=”dev_info_tr0”>
<tr class=”dev_info_tr2”>
<tr class=”dev_info_tr1”>
<span class=”dev_info_item1”>
<span class=”bool_false”>
<span class=”dev_info_item1”>
<span class=”bool_true”>
<th class=”dev_info_th”>
<td class=”dev_info_td2”>
<td class=”dev_info_td1”>

Touch Device #1

Series:	MTIR-V2.0
Identification:	MTIR240W-V2
Version:	2.00.000
Serial Number:	000011
Temperature: °C	+32.0
Accelerator: (X/Y/Z)	+0.032 / -1.076 / +0.016
Last touch event:	never
Scanning:	true
Frame rate:	113 Hz
Status:	empty
Number of modules:	10

Module Info:

Num.	ID	Firmware	S/N	State
-.		Interface	58914288	
0.	21	1.01.001	187868	ok
1.	02	1.00.000	186992	ok
2.	02	1.00.000	186994	ok
3.	02	1.00.000	186989	ok
4.	02	1.00.000	186995	ok
5.	02	1.00.000	186990	ok
6.	02	1.00.000	186988	ok
7.	02	1.00.000	186993	ok
8.	02	1.00.000	187115	ok
9.	02	1.00.000	187114	ok
10.	02	1.00.000	186991	ok

The default definitions for these classes are given in “default.css” as follows:

```
.dev_info_item1
{
    font-weight:bold;
    position:absolute; left:13em;
}

.dev_info_error
{
    font-weight:bold;
    background-color:#ff2020;
    color:#ffff00;
}

.dev_info_table
{
    border:solid 2px #000000;
    margin-left:1em;
    text-align:center;
    border-collapse:collapse;
}

.dev_info_tr0
{
    background:#404040;
    color:#ffffff;
}

.dev_info_tr1
{
    background:#fffff0;
    color:#000000;
}

.dev_info_tr2
{
    background:#fff0e0;
    color:#000000;
}

.dev_info_th
{
    border: solid 1px #808080;
    padding-left:0.5em;
    padding-right:0.5em;
}

.dev_info_td1
{
    border: solid 1px #000000;
    padding-left:0.5em;
    padding-right:0.5em;
}

.dev_info_td2
{
    border: solid 1px #000000;
    padding-left:0.5em;
    padding-right:0.5em;
}
```

5.4 XML Elements

For easy integration of status reports into user applications an XML the embedded web-server can output dreaMTouch status as well formed XML document. This document can be accessed at the URL http://localhost:8080/dreaMTouch.xml or if the default Port has been changed (e. g. to 9000) http://localhost:9000/dreaMTouch.xml.

dreaMTouch driver outputs XML version 1.0.

Elements that contain other elements only (in alphabetical order):

Element	Children	Parents	Description
<accelerometer>	*<x>* *<y>* *<z>*	<touch_*n*>	acceleration sensor output
<coordinates>	*<mirror_v>* *<mirror_h>* *<exchange_xy>* *<align_4p>*	<software>	information about coordinate processing
<detection>	<detectRotationAngle> <d2maxFactor> <flashSupression>	<software>	information about user settable detection parameters
<device_hardware>	*<touch_count>* *<logfile_count>* *<last_reinit>* <touch_*n*> <logfile_*n*> <error>	<dreaMTouch>	information about attached hardware and opened log-files
<dreaMTouch>	<general> <software> <device_hardware>	-	document root
<driver_application>	*<program_name>* *<version>* *<compiledate>*	<software>	information about the dreaMTouch protocol dispatcher service
<driver_core>	*<program_name>* *<version>* *<compiledate>*	<software>	information about the dreaMTouch driver core DLL
<error>	*<string>* *<number>*	<device_hardware> <touch_*n*> <logfile_*n*>	detailed error information
<general>	*<report_time>* *<computer_name>* *<computer_running>* *<driver_start>* *<security_level>*	<dreaMTouch>	general information about dreaMTouch driver, not related to certain applications or devices
<hid_settings>	*<add_diameter>* *<add_pressure>*	<output>	information about multi-touch finger input HID emulation details
<logfile_*n*>	*<series>* *<identification>* *<version>* *<serial_number>* <configFilePath> *<playing>* *<number_of_modules>* <module_info> <error>	<device_hardware>	information about lof-file #*n*
<module_info>	<usb_adapter> <sensor_interface> <module_*n*>	<touch_*n*> <logfile_*n*>	detailed information about ScanModules of touch or log-file #*n*
<module_*n*>	*<id>* *<version>* *<serial_number>* *<functional>*	<module_info>	detailed information about module #*n*
<mouse_HID_emulation>	*<program_name>* *<version>*	<software>	information about the installed mouse HID emulation driver
<mtf_HID_emulation>	*<program_name>* *<version>*	<software>	information about the installed multi-touch finger input HID emulation driver
<OSC_protocol_settings>	*<hostname>* *<port>*	<software>	information about settings common to all network based protocols
<output>	*<tuio2Dcur>* *<tuio2Dblb>* *<mim>* *<tuio_sPPPP>* *<wallmux>* *<mouse_enter>* *<mouse_leftright>* *<mouse_exchange_fingers>* *<mouse_wheel_distance>* *<mouse_wheel_sensitivity>* *<mouse_wheel_reverse>* *<hid_emulation>* <hid_settings>	<software>	information about selected coordinate output
<sensor_interface>	*<id>*	<module_info>	detailed information about sensor interface

Element	Children	Parents	Description
	<version> *<serial_number>*		
<software>	<mtf_HID_emulation> <mouse_HID_emulation> <driver_core> <driver_application> <OSC_protocol_settings> <output> <coordinates>	<dreaMTouch>	information about driver software and settings
<touch_*n*>	*<series>* *<identification>* *<version>* *<serial_number>* <configFilePath> *<temperature>* <accelerometer> *<last_touch_event>* *<scanning>* *<frame_rate>* *<position_count>* *<number_of_modules>* <module_info> <error>	<device_hardware>	information about touch device #*n*
<usb_adapter>	*<serial_number>*	<module_info>	detailed information about USB-adapter

Elements that contain text only (in alphabetical order):

Element	Parents	Type	Format	Description
<add_diameter>	<hid_settings>	boolean		if touch point size is reported as diameter at the multi-touch finger input HID device
<add_pressure>	<hid_settings>	boolean		if touch point size is reported as tip-pressure at the multi-touch finger input HID device
<align_4p>	<coordinates>	boolean		if 4-point alignment is active, or not
<compiledate>	<driver_core> <driver_application>	string	MMM-DD-YYYY hh:mm:ss	compile date and time of the executable as returned by __DATE__ and __TIME__ macros
<computer_name>	<general>	string		name of the computer where citmuto03 is running
<computer_running>	<general>	boolean		indicates if cimuto03 thinks computer is running (true) or suspended (false)
<configFilePath>	<touch_n> <logfile_n>	string		absolut path of configuration file (*.dcf) for this touch, or „compiled in config“, or „false“
<d2maxFactor>	<detection>	float		percentage of touch area diagonale that is used as „Detection Distance“
<detectRotationAngle>	<detection>	boolean		ellipse rotation angle detection is on (true) or off (false)
<driver_start>	<general>	string	*YYYY-MM-DD hh:mm:ss* UTC	Date and time when citmuto03 service was started
<exchange_xy>	<coordinates>	boolean		if exchange of X/Y coordinates is enabled, or not
<frame_rate>	<touch_*n*>	integer		number of scan frames per second
<functional>	<module_*n*>	boolean		if this ScanModule is functional (true) or defective (false)
<hid_emulation>	<output>	boolean		if multi touch finger HID emulation is enabled, or not
<hostname>	<OSC_protocol_settings>	string		Name or IP of the host where UDP packets are sent to
<id>	<sensor_interface> <module_*n*>	hex-integer		module identification number
<identification>	<touch_*n*> <logfile_n>	string		identifications string of dreaMTouch device
<last_reinit>	<device_hardware>	string	*YYYY-MM-DD hh:mm:ss* UTC	Date and time when touched where successfully initialized
<last_touch_event>	<touch_*n*>	string	*YYYY-MM-DD hh:mm:ss* UTC	Date and time device was touch last
<logfile_count>	<device_hardware>	integer		number of open log-files
<mim>	<output>	boolean		if MIM protocol is enabled, or not
<mirror_h>	<coordinates>	boolean		if horzontal mirroring is enabled, or not
<mirror_v>	<coordinates>	boolean		if vertical mirroring is enabled, or not
<mouse_enter>	<output>	boolean		if mouse emulation in „enter“ mode is enabled, or not
<mouse_leftright>	<output>	boolean		if mouse emulation in „left/right“ mode is enabled, or not
<number_of_modules>	<touch_*n*> <logfile_n>	integer		number of scan-modules for this device
<number>	<error>	integer		error numberi
<playing>	<logfile_*n*>	boolean		whether log-file is playing, or not
<port>	<OSC_protocol_settings>	integer		UDP packet destination port
<position_count>	<touch_*n*>	integer		number of touch points
<program_name>	<mtf_HID_emulation> <mouse_HID_emulation> <driver_core> <driver_application>	string		name of the executable
<report_time>	<general>	string	*YYYY-MM-DD hh:mm:ss* UTC	Date and time when this report was generated

Element	**Parents**	**Type**	**Format**	**Description**
<scanning>	<touch_*n*>	boolean		whether device is scanning, or not
<security_level>	<general>	string	*root* or *user*	Indicates if citmuto03drv is running with elevated rights (root), as it should be. Or if running in user mode (user), whis is not recomended.
<serial_number>	<touch_*n*> <logfile_n> <usb_adapter> <sensor_interface> <module_*n*>	integer		Serial number of the device or module
<series>	<touch_*n*> <logfile_n>	string		name of the dreaMTouch hardware series
<string>	<error>	string		readable description of error
<temperature>	<touch_*n*>	float		temperature of the sensor interface
<touch_count>	<device_hardware>	integer		number of detected touch devices
<tuio_sPPPP>	<output>	boolean		if TUIO/_sPPPP profile is enabled, or not
<tuio2Dblb>	<output>	boolean		if TUIO/2Dblb profile is enabled, or not
<tuio2Dcur>	<output>	boolean		if TUIO/2Dcur profile is enabled, or not
<version>	<mtf_HID_emulation> <mouse_HID_emulation> <driver_core> <driver_application> <touch_*n*> <logfile_*n*> <sensor_interface> <module_*n*>	string		version of the executable, device, or module
<wallmux>	<output>	boolean		if wallmux protocol is enabled, or not (for xtreMTouch)
<x>	<accelerometer>	float		x-component of acceleration sensor output
<y>	<accelerometer>	float		y-component of acceleration sensor output
<z>	<accelerometer>	float		z-component of acceleration sensor output

6 Low Level SDK

If you don’t want to use TUIO protocol *mtirctl.dll* can be used for own applications as well. Therefore the files **`mtirctldll.h`** , **`mtirdefs.h`** and **`libmtirctl.so`** are provided in the libmtirctl1-dev package.

To use the DLL functions link your application against libmtirctl.so. The header file provides C-function prototypes and type definitions. The usage of the functions is documented by comments in the header file.

Hint: The service citmuto03drv may not be running, when the library shall be used by a user application!

Before any other function of mtirlctl.dll can be used the function **`MtirctlEnum()`** has to be called to get the number of attached devices. If more than zero devices are reported call **`MtirctlOpen(0)`** to open the first device.
After that set default calibration by calling **`MtirctlSetCalibration(0, &cal)`** with **`cal`** set to

```
cal.bMirrorH = false;
cal.bMirrorV = false;
cal.bExchangeXY = false;
cal.x0 = 0.0;
cal.y0 = 0.0;
cal.x1 = 1.0;
cal.y0 = 1.0;
```

Finally start scanning by calling **`MtirctlStartScan(0, NULL)`**.

A recommended way to read scan results is a timer function that frequently (e. g. every 10 ms) calls **`MtirctlGetScanResult(0, pScanResult)`**. If new scan results are available this function returns TRUE, otherwise FALSE.
If no scan results are read libmtirctl pauses scanning as soon as its result buffer is filled. At the moment this buffer has a size of 4 scan results.

The pointer pScanresult passed to the function must have allocated enough memory to hold the scan results. No memory is allocated by **`MtirctlGetScanResult()`** itself!
But the function is programmed in a way that unused parts of pScanresult may be passed as a NULL pointer. This helps user to save memory and keep the usage simple.

7 Advanced Usage

7.1 Multi Destination TUIO output

Usually TUIO packets are sent to one host address and one port number at a time only. This host address / port combination can be set in the GUI at the “protocol”-tab.
But sometimes it is required to send the same TUIO packet to additional host addresses and / or additional ports.
To allow such configurations the special key [extraTUIO] in the configuration file (*/etc/xdg/Citron\ GmbH/dreaMTouch.conf*) is used:

Under this key string DWORD value pairs are used to define additional address/port combinations where TUIO packets will be duplicated to:

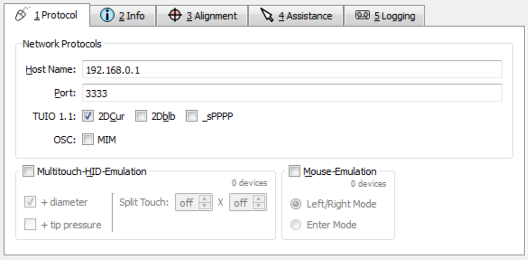
Value	**Description**	**Paramerter Range**	**Example**
host<n>	host name or IPV4-address	<n> = [1..255]	host1=192.168.1.1
port<n>	port number	<n> = [1..255]	port1=3334

The driver searches for values with the name “host1”, “host2”, “host3”, …, and so on. The number <n> in the host value name must be consecutive and without leading zeros. Searching for additional hosts stops at the first gap.
For each host an individual port can be defined. If no port is defined, the last found port number is used.

Examples:
a) Define 4 TUIO-hosts with 4 different ports:

192.168.0.1:3333
192.168.0.2:3334
192.168.0.3:3335
192.168.0.4:3336

The first host and port is defined using the GUI.



Network Protocols
Host Name: 192.168.0.1
Port: 3333
TUIO 1.1: ☑ 2DCur □ 2Dblb □ _sPPPP
OSC: □ MIM
□ Multitouch-HID-Emulation 0 devices
☑ + diameter
□ + tip pressure
Split Touch: off X off
□ Mouse-Emulation 0 devices
● Left/Right Mode
○ Enter Mode

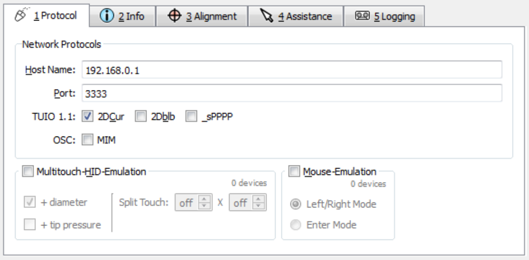
The remaining 3 hosts and ports are defined in the configuration file */etc/xdg/Citron\ GmbH/dreaMTouch.conf*.

```
[extraTUIO]
host1=192.168.0.2
host2=192.168.0.3
host3=192.168.0.4
port1=3334
port2=3335
port3=3336
```

b) Define 2 TUIO-hosts with 4 different ports

192.168.0.1:3333
192.168.0.1:3334
192.168.0.2:3335
192.168.0.2:3336

The first host and port is defined using the GUI.



Network Protocols
Host Name: 192.168.0.1
Port: 3333
TUIO 1.1: ☑ 2DCur □ 2Dblb □ _sPPPP
OSC: □ MIM

□ Multitouch-HID-Emulation
0 devices
☑ + diameter
□ + tip pressure
Split Touch: off X off

□ Mouse-Emulation
0 devices
● Left/Right Mode
○ Enter Mode

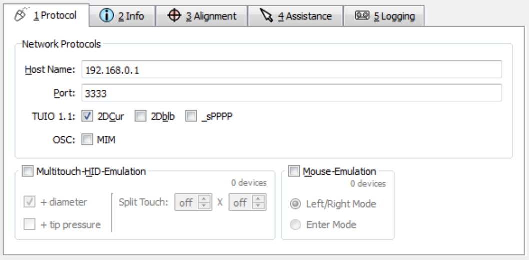
The remaining 3 hosts and ports are defined in the configuration file */etc/xdg/Citron\ GmbH/dreaMTouch.conf*.

```
[extraTUIO]
host1=192.168.0.1
host2=192.168.0.2
host3=192.168.0.2
port1=3334
port2=3335
port3=3336
```

c) define 4 TUIO-hosts, all using the same port

192.168.0.1:3333
192.168.0.2:3333
192.168.0.3:3333
192.168.0.4:3333

The first host and port is defined using the GUI.



Network Protocols
Host Name: 192.168.0.1
Port: 3333
TUIO 1.1: ☑ 2DCur □ 2Dblb □ _sPPPP
OSC: □ MIM

□ Multitouch-HID-Emulation
0 devices
☑ + diameter
□ + tip pressure
Split Touch: off X off

□ Mouse-Emulation
0 devices
● Left/Right Mode
○ Enter Mode

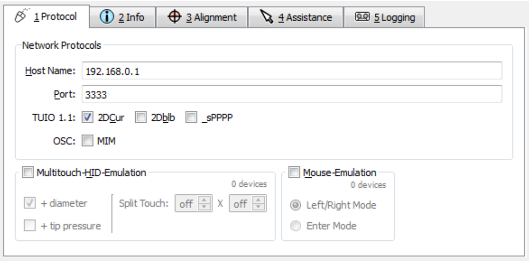
The remaining 3 hosts are defined in the configuration file. The ports are duplicates of the first port and need not to be defined.

```
[extraTUIO]
host1=192.168.0.2
host2=192.168.0.3
host3=192.168.0.4
```

d) define 4 TUIO-hosts, first two and second two use the same port

192.168.0.1:3333
192.168.0.2:3333
192.168.0.3:3334
192.168.0.4:3334

The first host and port is defined using the GUI.



Host Name: 192.168.0.1
Port: 3333
TUIO 1.1: ☑ 2DCur □ 2Dblb □ _sPPPP
OSC: □ MIM
□ Multitouch-HID-Emulation 0 devices
☑ + diameter
□ + tip pressure
Split Touch: off X off
□ Mouse-Emulation 0 devices
● Left/Right Mode
○ Enter Mode

The remaining 3 hosts are defined in the configuration file. The port of the second host is duplicated by the driver from the first host:port. The third host gets a new port definition and the fourth port is duplicated from the third one, since this was the last valid definition.

```
[extraTUIO]
host1=192.168.0.2
host2=192.168.0.3
host3=192.168.0.4
port2=3335
```

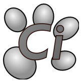
8 Trouble Shooting

This chapter describes all possible error messages, their causes and likely solutions or work-rounds.

8.1 Error Signaling

When the dreaMTouch GUI citmuto03gui is running, it shows error conditions in four different ways:

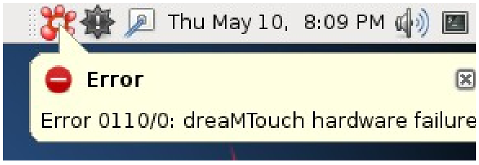
1) The tray icon, on the task bar, changes its color:

driver service not running: 

regular operations:

error occurred:

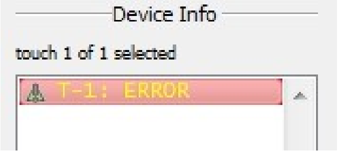
2) An error message pops-up from the system tray
(if not suppressed by "-quiet" command line option):



Error

Error 0110/0: dreaMTouch hardware failure

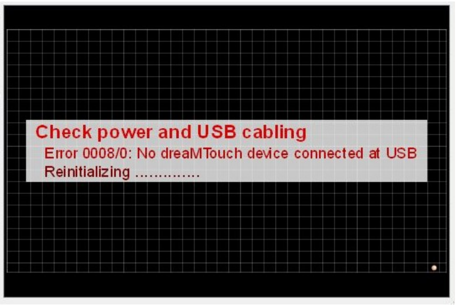
3) The error is indicated in the touch status list box:



touch 1 of 1 selected

T-1: ERROR

4) In advanced mode a detailed error message is shown in the touch map area:



Check power and USB cabling

Error 0008/0: No dreaMTouch device connected at USB

Reinitializing ....

8.2 Error Message Description

The following error messages may be reported by citmuto03

No.	Error Message	Short Description
0x001	"LastToken not received"	MTIR-V1 Hardware error
0x002	"Cannot open USB device"	FTDI chip driver has a communication problem
0x003	"Cannot reset USB device"	FTDI chip driver has a communication problem
0x004	"Cannot examine queue status"	FTDI chip driver has a communication problem
0x005	"Cannot set bitmode"	FTDI chip driver has a communication problem
0x006	"Cannot sync MPSSE"	Synchronization problem during initialization
0x007	"Cannot initialize USB device"	FTDI chip driver has a problem finding the dreaMTouch device
0x008	"No dreaMTouch device connected at USB"	no dreaMTouch device found
0x009	“Communication error (unplugged device?)”	Error occurred on the communication protocol
0x00A	“Cannot reset USB port”	While trying to reconnect a device the USB port reset failed
0x101	"Unexpected Module ID: [Module=*n*, ID=*n*]”	dreaMTouch hardware shows irregular behaviour
0x102	"Cannot initialize thread handler"	OS problem
0x103	"Cannot identify device (not powered?)"	Attached device cannot be identified
0x104	"Device configuration error"	dreaMTouch configuration problems
0x105	"Unknown dreaMTouch hardware detected"	dreaMTouch identification or power supply problems
0x106	"<Internal driver error: invalid touch handle>"	Software bug in citmuto03
0x107	"Fatal hardware error at dreaMTouch modules"	more than two ScanModules failed during first initialization
0x109	“Timed out”	timeout occurred during driver internal operations
0x10A	“Device is already open”	software bug in citmuto03
0x10C	“No touch attached to logger”	tried to write a log file without a real device attached
0x10D	“Cannot allocate touch device”	out of memory while allocating buffers
0x10E	“Unknown dreaMTouch V2.x hardware detected”	internal dreaMTouch communication problems
0x10F	“Unknown dreaMTouch V1.x hardware detected”	Internal dreaMTouch communication problems
0x110	"dreaMTouch hardware failure"	dreaMTouch hardware defect
0x201	“File not found”	Log-file was not found or could not be opened
0x202	“Unsupported file format”	Log-file format not supported by this driver
0x203	“This is not a logger”	Software bug in citmuto03
0xFFF	“dreaMTouch not initialized”	No dreaMTouch device initialized, yet
-	"<Internal driver error: Unknown>"	Software bug in citmuto03

8.2.1 Typical Error Causes

Even if it sounds trivial it’s true: most errors are cause by simple connection problems.
Please check power supply to the dreaMTouch device and USB cabling in case of an error first.

- Are all power supply connections plugged in correctly (mains, power supply, dreaMTouch)?
- Are USB cables connected correctly?
- Are all cables in sound condition (no sharp bends or cuts)?
- Try other USB ports on the computer. Often the front USB ports are usable for low speed devices (keyboard, mouse) only.

8.2.2 Cannot Change Protocol or Calibration Settings

Elevated rights are required to change any settings that influence touch operation. See chapter 4.3.1 on page 16 for a description on how the get elevated rights.

8.2.3 “Disconnect - Reinit” Procedure

This procedure should be followed exactly to solve many of the errors described below.

1) Disconnect USB cable between dreaMTouch device and computer
2) Switch dreaMTouch power off (MTIR240W/550W/FR01/FR02: Power off monitor completely, MTIR460W: Unplug power supply from touch)
3) Wait for at least 15 seconds
4) Reconnect power to dreaMTouch device
5) Reconnect USB cable
6) Wait for another 10 seconds
7) Click “Reinit” button if touchscreen is not found

8.2.4 Detailed Error Descriptions

8.2.4.1 "LastToken not received"

Affected Hardware: MTIR-V1.x

Error Class: Hardware Error

Error Cause: ScanModule Firmware error or broken cabling.

Solution: Try “Disconnect - Reinit” Procedure.
If this does not help, send hardware back for repair.

8.2.4.2 "Cannot open USB device"

Affected Hardware: MTIR-V1, MTIR-V2

Error Class: FTDI chip driver error

Error Cause: FTDI drivers are corrupted, not running or not installed correctly.

Solution: Check cabling, then try “Disconnect - Reinit” procedure.
Connect to another USB port, then try “Disconnect - Reinit” procedure.
If this does not help, restart Windows.
If this does not help, exit citmuto03, disconnect hardware, uninstall citmuto03 and FTDI drivers. Then do a fresh install of citmuto03 and FTDI drivers.

8.2.4.3 "Cannot reset USB device"

Affected Hardware: MTIR-V1, MTIR-V2

Error Class: FTDI chip driver communications error

Error Cause: The USB communication queue is corrupted.

Solution: Try “Disconnect - Reinit” Procedure.
If this does not help, reboot Windows.

8.2.4.4 "Cannot examine queue status"

Affected Hardware: MTIR-V1, MTIR-V2

Error Class: FTDI chip driver communications error

Error Cause: The USB communication queue is corrupted.

Solution: Try “Disconnect - Reinit” Procedure.
If this does not help, reboot Windows.

8.2.4.5 "Cannot set bitmode"

Affected Hardware: MTIR-V1, MTIR-V2

Error Class: FTDI chip driver communications error

Error Cause: The USB communication queue is corrupted.

Solution: Try “Disconnect - Reinit” Procedure.
If this does not help, reboot Windows.

8.2.4.6 "Cannot sync MPSSE"

Affected Hardware: MTIR-V1, MTIR-V2

Error Class: FTDI chip driver communications error

Error Cause: The USB communication queue is corrupted.

Solution: Try to click “Reinit” button in citmuto control center several times.
If this does not help, try “Disconnect - Reinit” Procedure.

8.2.4.7 "Cannot initialize USB device"

Affected Hardware: MTIR-V1, MTIR-V2

Error Class: Communication error on USB

Error Cause: Device not powered,
bad USB cabling,
corrupted USB driver stack

Solution: Check cabling, then follow “Disconnect - Reinit” Procedure.
If this does not help try another USB port, then follow “Disconnect - Reinit” procedure.
If this does not help, use another USB port and/or USB cable
If this does not help, reboot.

8.2.4.8 "No dreaMTouch device connected at USB"

Affected Hardware: MTIR-V1, MTIR-V2

Error Class: Configuration error on dreaMTouch device
Hardware error

Error Cause: Wrong configuration of USB interface on dreaMTouch device,
corrupted USB driver stack

Solution: Check cabling, then follow “Disconnect - Reinit” Procedure.
If this does not help try another USB port, then follow “Disconnect - Reinit” procedure.
If this does not help, reboot.
If this does not help, send device back for repair

8.2.4.9 "Communication error (unplugged device?)"

Affected Hardware: MTIR-V1, MTIR-V2

Error Class: Hardware error

Error Cause: Device got unplugged or USB stopped communications.

Solution: Check cabling, usually device gets reconnected automatically after some seconds.
If this does not help follow “Disconnect - Reinit” Procedure.
If this does not help try another USB port, then follow “Disconnect - Reinit” procedure.
If this does not help, reboot.
If this does not help, send device back for repair

8.2.4.10 "Cannot reset USB port"

Affected Hardware: MTIR-V1, MTIR-V2

Error Class: Hardware error

Error Cause: Failure while trying to reconnect a device.

Solution: Follow “Disconnect - Reinit” Procedure.
If this does not help try another USB port, then follow “Disconnect - Reinit” procedure.
If this does not help, reboot.
If this does not help, send device back for repair

8.2.4.11 Unexpected Module ID: [Module=*n*, ID=*n*]”

Affected Hardware: MTIR-V1, MTIR-V2

Error Class: Configuration error on dreaMTouch device
Hardware error

Error Cause: Wrong configuration of USB interface on dreaMTouch device,
corrupted USB driver stack

Solution: Try “Disconnect - Reinit” Procedure.
If this does not help, reboot.
If this does not help, send device back for repair

8.2.4.12 "Cannot initialize thread handler"

Affected Hardware: MTIR-V1, MTIR-V2

Error Class: citmuto03 software error

Error Cause: Other applications taking too much resources from Windows.
Low memory condition.
Other operating system related problems

Solution: Reboot.
If this does not help, look for running applications that are not required and close them. Alternatively install more system memory.

8.2.4.13 "Cannot identify device (not powered?)"

Affected Hardware: MTIR-V1, MTIR-V2

Error Class: Power supply error

Error Cause: The attached dreaMTouch device reports an identification string that is unknown to this citmuto03 driver version.
This is causes by missing power supply to the device in most cases.

Solution: Check device power supply. Most dreaMTouch monitors require a separate desktop power supply to operate. Monitor has to be switched on as well.

8.2.4.14 "Device configuration error"

Affected Hardware: MTIR-V1, MTIR-V2

Error Class: Communication or configuration error

Error Cause: The device could not be initialized after it was identified correctly. This might be caused by illegal registry settings, damaged device hardware or communication problems.

Solution: Try “Disconnect - Reinit” procedure.
If this does not help, reboot.
If this does not help, open rename the configuration file /etc/xdg/Citron\ GmbH/dreaMTouch.conf to /etc/xdg/Citron\ GmbH/dreaMTouch.bak and restart citmuto03.
If this does not help write an email to support@citron.de. This email must contain a phone number to call back and an exact description of the device you are using including its serial number and the driver version you have tested.

8.2.4.15 "Unkown dreaMTouch hardware detected"

Affected Hardware: MTIR-V1, MTIR-V2

Error Class:

- Communication or configuration error
- Power supply error

Error Cause: The device could not be identified as MTIR-V1 or MTIR-V2 during initialization.
A common cause is missing power supply for the device.
This might be caused by damaged device hardware or communication problems, too.

Solution: Check cabling, then follow “Disconnect - Reinit” Procedure.
If this does not help try another USB port, then follow “Disconnect - Reinit” procedure.
If the error still exists, reboot.
If this does not help, send device back for repair

8.2.4.16 "<Internal driver error: invalid touch handle>"

Affected Hardware: MTIR-V1, MTIR-V2

Error Class: citmuto03 driver bug

Error Cause: Software bug in citmuto03 driver

Solution: Contact support@citron.de with an exact description of the circumstances that lead to this error. Give detailed description of used driver version and dreaMTouch hardware.

8.2.4.17 "Fatal hardware error at dreaMTouch modules"

Affected Hardware: MTIR-V2

Error Class:

- Communication error
- Hardware error

Error Cause: Up to one ScanModule may fail without affecting touch performance severely. If more than one module fails during first initialization, this error is emitted.

Solution: Check cabling, usually device gets reconnected automatically after some seconds.
If this does not help follow “Disconnect - Reinit” Procedure.
If this does not help, send device back for repair

8.2.4.18 "Timed out"

Affected Hardware: MTIR-V1, MTIR-V2

Error Class:

- Communication error
- Hardware error

Error Cause: During device communications a thread timed out. This is most probably caused by a disconnected USB or power supply.

Solution: Check cabling, usually device gets reconnected automatically after some seconds.
If this does not help follow “Disconnect - Reinit” Procedure.
If this does not help try another USB port, then follow “Disconnect - Reinit” procedure.
If the error still exists, reboot.
If this does not help, send device back for repair

8.2.4.19 "Device is already open"

Affected Hardware: MTIR-V1, MTIR-V2

Error Class: citmuto03 driver bug

Error Cause: Software bug in citmuto03 driver

Solution: Contact support@citron.de with an exact description of the circumstances that lead to this error. Give detailed description of used driver version and dreaMTouch hardware.

8.2.4.20 "No touch attached to logger"

Affected Hardware: MTIR-V1, MTIR-V2

Error Class: citmuto03 driver bug

Error Cause: Software bug in citmuto03 driver

Solution: Contact support@citron.de with an exact description of the circumstances that lead to this error. Give detailed description of used driver version and dreaMTouch hardware.

8.2.4.21 "Cannot allocate touch device"

Affected Hardware: MTIR-V1, MTIR-V2

Error Class: citmuto03 software error

Error Cause:

- Low memory condition.
- Other operating system related problems
- Software bug in citmuto03 driver

Solution: Reboot.
If this does not help, look for running applications that are not required and close them. Alternatively install more system memory.
If this does not help, contact support@citron.de with an exact description of the circumstances that lead to this error. Give detailed description of used driver version and dreaMTouch hardware.

8.2.4.22 "Unknown dreaMTouch V2.x hardware detected"

Affected Hardware: MTIR-V2

Error Class: Communication or configuration error

Error Cause: The device could not be identified as MTIR-V2 during initialization.
This might be caused by damaged device hardware or communication problems

Solution: Check cabling, then follow “Disconnect - Reinit” Procedure.
If this does not help try another USB port, then follow “Disconnect - Reinit” procedure.
If the error still exists, reboot.
If this does not help, send device back for repair

8.2.4.23 "Unknown dreaMTouch V1.x hardware detected"

Affected Hardware: MTIR-V1

Error Class: Communication or configuration error

Error Cause: The device could not be identified as MTIR-V2 during initialization.
This might be caused by damaged device hardware or communication problems

Solution: Check cabling, then follow “Disconnect - Reinit” Procedure.
If this does not help try another USB port, then follow “Disconnect - Reinit” procedure.
If the error still exists, reboot.
If this does not help, send device back for repair

8.2.4.24 "dreaMTouch hardware failure"

Affected Hardware: MTIR-V2

Error Class:

- Communication error
- Hardware error

Error Cause: Up to one ScanModule may fail without affecting touch performance severely. If more than one module fails, this error is emitted.

Solution: Check cabling, usually device gets reconnected automatically after some seconds.
If this does not help follow “Disconnect-Reinit” Procedure.
If this does not help, send device back for repair

8.2.4.25 "File not found"

Affected Hardware: None

Error Class: citmuto03 software error

Error Cause: The file name given to open as a log-file was not found or could not be opened.

Solution:

- Retry to open with existing file name
- Check file access permissions

8.2.4.26 "Unsupported file format"

Affected Hardware: None

Error Class: citmuto03 software error

Error Cause: The file that should be opened as log file for replay has a wrong format

Solution: Try to open a real citmuto03 log file.

8.2.4.27 "This is not a logger"

Affected Hardware: None

Error Class: citmuto03 driver bug

Error Cause: Software bug in citmuto03 driver

Solution: Contact support@citron.de with an exact description of the circumstances that lead to this error. Give detailed description of used driver version.

8.2.4.28 "dreaMTouch not initialized"

Affected Hardware: MTIR-V1, MTIR-V2

Error Class: Informational message

Error Cause: The device was connected but the “Reinit” was not performed for some reasons.

Solution: Click on “Reinit” button of citmuto03 Control center

8.2.4.29 "<Internal driver error: Unknown>"

Affected Hardware: MTIR-V1, MTIR-V2

Error Class: citmuto03 driver bug

Error Cause: Software bug in citmuto03 driver

Solution: Contact support@citron.de with an exact description of the circumstances that lead to this error. Give detailed description of used driver version and dreaMTouch hardware.