

TSHARC Linux Driver User Manual

Document #:	AD-120005-001
Title:	TSHARC Linux Driver User Manual
Subtitle:	
Date:	14-May -2012
Description:	This document describes how to use the TSHARC driver software developed for Linux.

TSHARC Linux Driver User Manual	1
General Installation Notes	1
Features and Changes	1
Setup	2
Installation Options	2
Configuration File Directory Selection	3
Install selected components	3
Configuring Daemon to Automatically Start	3
Using the Control Panel	4
Start the Control Panel	4
Controller Selection – By Entering ID	5
Controller Selection – By Touch	5
Calibration	6
GUI Calibration	6
Edge Acceleration	7
Click Settings	8
Touch Settings	9
Advanced Settings	9
Write Current Controller Data	12
Exit the Control Panel	12
Calibration Utility (“tscale”) Command Line Parameters	12
Calibration Utility (“tscale”) Configuring Timeouts	13
Uninstallation	13
Troubleshooting	14

General Installation Notes

Microchip TSHARC drivers use the display-driver-software settings to configure various touch screen driver setup files. Install the display and display driver cards properly before installing any TSHARC touch screen controller drivers. If the display is not configured and working properly prior to loading the TSHARC controller drivers, the TSHARC controllers will not function properly.

While Microchip goes to great lengths to insure that all of the controllers and drivers will provide the highest possible performance and will even improve the performance and extend the life of a poor quality or failing touch screen, the overall accuracy and stability of the calibration will be dependent on the quality of the touch screen. Linearity, sheet resistance, contact resistance, tail assembly, capacitance and the printed silver linearization pattern vary between touch screen manufacturers, construction, assembly and technology. For more information regarding touch screen constructions and types, contact Microchip Technology.

Microchip controllers are universal by design. Configuration of the controller is available for any number of touch screen types as well as communication and power settings. Please check the TSHARC controller board to insure that the user-configurable configuration settings are set correctly for the desired application. It is essential that the system used is set-up properly. Please review this manual to insure the system is ready. **The users' manual for each Microchip controller board is available at "www.microchip.com/tsharc".** Once at the website, select the "Products" tab along the left side of the window and navigate to the corresponding TSHARC controller product. Once the system is verified to be working properly, proceed with the TSHARC controller driver installation procedure.

Features and Changes

Features and changes to this driver from 3.0.4 include:

- Robustness improvements
- Removed PS2 support
- Daemon now uses UINPUT kernel mechanism to remove dependence on X11 for cursor movement
- Working daemon on systems without X-Windows
- ARM processors are no longer supported with driver. There are separate open source solutions that may be downloaded from the Microchip website for non-X86 target platforms.
- Daemon service now registered using Sys-V mechanism to auto start after reboot.
- Improved "tscal" command-line parameter support
- Bundle now self-extracts and installs.
- Added the ability to choose a custom serial device path during setup.
- Customizable calibration timeouts

Up to 6 controllers can be operating on the same system concurrently.

Since there is a wide variance among Linux distributions, there may be issues with the daemon automatically starting after a reboot for some distributions. However, this feature should work on most systems.

Setup

To start installation of the driver, run "sudo sh ./Tsharc323.run" at the command line.

Command:

```
sudo sh ./Tsharc323.run
```

Example output:

```
Verifying archive integrity... All good.  
Uncompressing TSHARC Daemon Version 3.23.....
```

Installation Options

Select up to six controller components to install from the listed menu. Make a selection by entering in the corresponding number and pressing "Enter". Each additional component will take a moment to add to the list to be installed.

Example output:

```
TSHARC controller setup version 3.23  
This version supports a maximum of 6 controllers  
  
Currently selected TSHARC components:  
No controller components selected  
  
1. Toggle TSHARC USB Controller support  
2. Toggle TSHARC SERIAL Controller on COM1 (/dev/ttyS0)  
3. Toggle TSHARC SERIAL Controller on COM2 (/dev/ttyS1)  
4. Toggle TSHARC SERIAL Controller on COM3 (/dev/ttyS2)  
5. Toggle TSHARC SERIAL Controller on COM4 (/dev/ttyS3)  
6. Toggle TSHARC SERIAL Controller on COM5 (/dev/ttyS4)  
7. Toggle TSHARC SERIAL Controller on COM6 (/dev/ttyS5)  
8. Toggle TSHARC SERIAL Controller on COM7 (/dev/ttyS6)  
9. Toggle TSHARC SERIAL Controller on COM8 (/dev/ttyS7)  
10. Toggle TSHARC SERIAL Controller on custom device path  
11. Set configuration file directory (Current: "/etc/tsharc")  
12. Install selected components  
13. Exit  
  
Enter selection:
```

Configuration File Directory Selection

In this screen, the configuration directory can be modified. Enter the desired directory name and press "Enter".

Example output:

TSHARC controller setup version 3.23
This version supports a maximum of 6 controllers

Currently selected TSHARC components:
No controller components selected

Please enter the configuration file directory:

Install selected components

Select the install menu choice.

It is recommended that all controllers be connected before install, but it is not required.

Example output:

Starting installation
Creating configuration directory...
Copying files...
Setting permissions...
Preparing configuration file directory...
tsharcusb.conf configuration file does not exist, creating...
Installing TSHARC USB Controller support
Creating daemon script
Registering startup script with SysV
Copying daemon script
`./boot.tsharc' -> `/etc/init.d/boot.tsharc'
Creating uninstall script
Copying uninstall script
Setting permissions on uninstall script
Starting tsharcd boot.tsharc: Detected 1 controllers
Done.
Installation is complete.

Configuring Daemon to Automatically Start

The TSHARC daemon is automatically registered (no additional configuration necessary) to automatically start using runlevels one thru five on systems that support Sys -V initialization.

Using the Control Panel

The general process for setting up a controller on the target system follows these steps. This process must be repeated for every controller used on the system.

1. Start the control panel
2. Select a controller to configure
3. Calibrate the current controller
4. Set Edge Acceleration(Optional)
5. Set Click Settings(Optional)
6. Set Touch Settings(Optional)
7. Set Advanced Settings(Optional)
8. Write Current Controller Data
9. Exit Control Panel

Start the Control Panel

To run the control panel, enter "hlincl" in the command prompt console. Enter in the numbered choice and press "Enter" to continue. (Note: The daemon must be started in order for "hlincl" to run. However, since the daemon is automatically started, this should not be an issue.)

- "Controller Selection" allows the user to configure each controller individually on the system.
- "Calibration" runs the calibration/linearization routine
- "Edge Acceleration" modifies the edge acceleration settings.
- "Click Settings" modifies the click options of the touch screen.
- "Advanced Settings" includes EEPROM support, X and Y offset, and custom width and height.
- "OK" exits the control panel and writes the configuration data.
- "Cancel" exits the control panel and reads the original configuration data.
- "Apply" will apply the current configurations.

Example output:

```
Microchip Control Panel Version 3.23
Current controller: ID: 0, Interface: SERIAL on /dev/ttyS0
```

Main Menu

1. Controller Selection
2. Calibration
3. Edge Acceleration
4. Click Settings
5. Touch Settings
6. Advanced Settings
0. OK - Exit and Write Current Controller Data
- C. Cancel - Exit and Read Current Controller Data
- A. Apply - write Current Controller Data

Enter selection:

Controller Selection

If using multiple controllers on a system, each controller must be configured individually. Use this screen to choose between available connected controllers.

At the top of the screen, the current controller information, version number, and the controller's device path are listed.

If multiple controllers are to be configured, there are two ways to select the current controller.

- By entering the ID
- By touching the touch screen

If no other controllers are connected to the system, enter "P" and press "Enter" to return to the Control Panel options.

Example output:

Microchip Control Panel Version 3.23

Current controller: ID: 0, Interface: SERIAL on /dev/ttyS0

Controller Selection Menu

1. Select Current Controller by Entering ID
 2. Select Current Controller by Touching Screen
- P. Return to Previous Menu

Enter selection:

Controller Selection – By Entering ID

After selecting this option, a list of controllers connected to the system will be available. The example output below shows only one controller connected to the system.

Enter the corresponding number for the desired controller. Press "Enter".

After selecting the controller, the current controller's information will be displayed on the top of the menus.

Example output:

Please select a controller:

1. ID: 0 TYPE: SERIAL DEVICEPATH: /dev/ttyS0

Enter selection:

Controller Selection – By Touch

A prompt will appear to touch the screen of the controller to be configured.

Touch the screen and press "Enter".

The new controller's information will be displayed on the "Selecting Controller to Configure" screen as shown in below example.

Example output:

Microchip Control Panel Version 3.23

Current controller: ID: 0, Interface: SERIAL on /dev/ttyS0

Controller Selection Menu

Please touch the touch screen of controller to be selected:
Press Enter after screen has been touched.

Calibration

To select the number of calibration points used in the routine, select one of the first five menu choices. The selection made is indicated by "(Current)" next to the selection. Shown in the below example, a 9 point calibration is selected. Enter a number, 1 through 5, and press "Enter". The (Current) will now be next to the chosen calibration.

To run a graphic calibration, a GUI calibration, enter 6 and press "Enter". A calibration routine will launch.

To run a text calibration, select 7 and press "Enter".

Calibration options are provided to allow for user control of the calibration results of the touch screen. Any calibration involving more than 9 points will linearize the touch screen. While a 9 point calibration will reduce the edge-bow, a 25 point calibration will be even more accurate. A 28 point calibration differs from a 25 point calibration in that more emphasis is added to the corners of the touch screen. A 4 or 5 point calibration will align the screen. Aligning the screen will move the screen relative to the current display. The more points used in the calibration routine, the more linear the touch screen will be.

The calibration targets are inset to bring the calibration routine targets away from the edge of the display. This eliminates physical limitations put into place by the screen's bezel. Default calibration inset values depend on the number of points selected.

4pt., 5pt. – 20% Inset
9pt., 25pt., 28pt. – 2% Inset

Example output:

Microchip Control Panel Version 3.23

Current controller: ID: 0, Interface: SERIAL on /dev/ttyS0

Calibration Menu

1. 4 Point Calibration
2. 5 Point Calibration
3. 9 Point Calibration (Current)
4. 25 Point Calibration
5. 28 Point Calibration
- I. Inset (Current: 5)
6. Begin GUI Calibration
7. Begin Text Calibration
- P. Return to Previous Menu

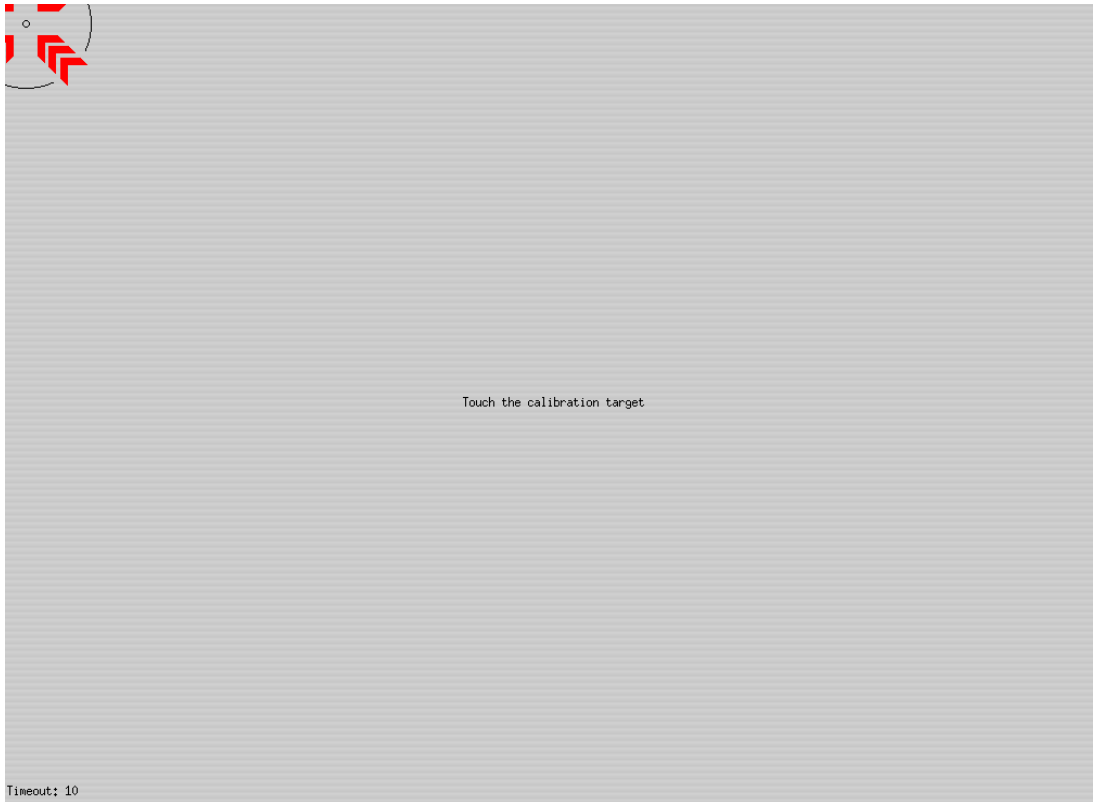
Enter selection:

GUI Calibration

It is important that all calibration routines be completed using a finger or a stylus. Using the available mouse will exit the routine immediately.

Each target of the calibration routine will appear one at a time. Touch the center of each target as accurately as possible. Microchip's specially designed calibration targets assist in calibrating the touch screen as accurately as possible. If using a stylus during operation, calibrate with a stylus. Be sure to position yourself in front of the touch screen as it will normally be used, sitting or standing. This will reduce error when calibrating the touch screen.

The calibration screen will automatically time out and return to the calibration menu if not touched within 10 seconds. This time-out feature insures that the user can exit the calibration screen in the event that the user has incorrectly calibrated the touch screen or the touch screen has been damaged or disconnected from the system.



Edge Acceleration

Edge acceleration enables the user to accelerate the cursor to the edge of a touch screen display without touching the active area edge. Without this feature, it is generally difficult to reach the outer edge because of the limitations the display bezel may impose on the user. Use edge-acceleration for systems that implement an "auto-hide" task bar or buttons at the edge of the display.

The Edge Acceleration menu enables the user to set the point at which edge acceleration will take effect.

Setting these values too high may make it difficult to use the screen normally. The maximum amount of edge acceleration is 25%. Typically, only a 5% edge acceleration is needed. Values between 0 and 255 are relative to 0% to 25% edge acceleration, where 5% has the value of 51.

Example output:

Microchip Control Panel Version 3.23

Current controller: ID: 0, Interface: SERIAL on /dev/ttyS0

Edge Acceleration Menu

1. Set acceleration for left edge of screen (Current: 0)
2. Set acceleration for top edge of screen (Current: 0)
3. Set acceleration for right edge of screen (Current: 0)
4. Set acceleration for bottom edge of screen (Current: 0)
- P. Return to Previous Menu

Enter selection:

Click Settings

Right Click Delay

Set the "Right-Click Delay" value to the preferred time, in milliseconds, needed to produce a right click event. A value of 0 will disable this function.

Right Click Area

The event area should be set to an area that is slightly larger than the activator tip. If the activator is a fingertip, the right click area should be at least as big as your fingertip.

Use the following formula to find the appropriate value:

$$area = 4096 \times \frac{event\ width}{width\ of\ touch\ screen}$$

Button used for Left Clicks

This is used to change the button used for a left-click event when the screen is touched.

Button used for Right Clicks

This is used to change the button used for right-click events when the screen is touched. This may need to be changed to "2" depending on the target system configuration.

Enter in the corresponding menu number and press "Enter".

Example output:

Microchip Control Panel Version 3.23

Current controller: ID: 0, Interface: SERIAL on /dev/ttyS0

Click Settings Menu

1. Set right click delay in milliseconds (Current: 0)
2. Set right click area (Current: 100)
3. Set button used for left clicks (Current: 1)
4. Set button used for right clicks (Current: 3)
- P. Return to Previous Menu

Enter selection:

Touch Settings

Normal Mode

This mode emulates a standard mouse. Selecting "Normal" will allow for single click, double click, drawing, dragging and right click option. This mode will allow the cursor to operate as a computer mouse typically would.

Touch Down Mode

Touch-down mode will allow for a click event to take place at "touch-down". This mode will allow the cursor to operate as a single button-press or a single left-click of the computer mouse. The user will not be able to draw or drag if selecting this option.

Touch Up Mode

A touch is sent only at touch-up in this mode. Once lifted, the touch will register as a single left-click or button press. This disables right click and double click functions.

Enter in the corresponding menu number and press "Enter".

Example output:

Microchip Control Panel Version 3.23

Current controller: ID: 0, Interface: SERIAL on /dev/ttyS0

Touch Settings Menu

1. Select Normal Mode (Current)
2. Select Touch Down Mode
3. Select Touch Up Mode
- P. Return to Previous Menu

Enter selection:

Advanced Settings

These settings normally only need to be modified for multi-monitor configurations. Each controller in a multi-monitor scenario will need to be configured individually. The main purpose of these settings is to restrict cursor movement and touch click to the rectangular region starting at the X and Y Offset (upper-left of region) and ending at <X offset + X width>, <Y offset + Y height> (lower-right of region). These settings need to be set prior to calibration so that the calibration application ("tscal") can display the calibration targets at the proper locations of the intended region. The example output below demonstrates this menu.

Example output:

Microchip Control Panel Version 3.23

Current controller: ID: 0, Interface: SERIAL on /dev/ttyS0

Advanced Settings Menu

1. Enable EEPROM Reading and Writing (Current: 1)

Note: X offset and Y offset is normally only need in xinerama

environments with multiple controllers and displays

2. Set X Offset in Pixels (Current: 0)

3. Set Y Offset in Pixels (Current: 0)

Note: A value of zero for width or height will cause driver to use current screen size

4. Set Custom Width in Pixels (Current: 0)

5. Set Custom Height in Pixels (Current: 0)

P. Return to Previous Menu

Enter selection:

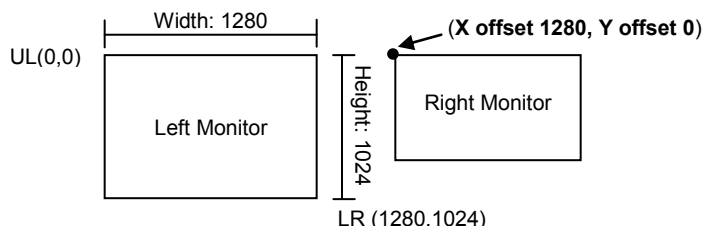
Enable EEPROM Reading and Writing

Configurations changes that correspond to controller features made within the control panel will be saved to the controller's EEPROM. This allows the controller to maintain calibration and touch settings on any system with an EEPROM supporting TSHARC driver installed. Currently, the driver supports the EEPROM features on Devil Ray C-ray TSHARC controllers. A value of "1" indicates to the driver that EEPROM data should be written to the controller while a value of "0" indicates that EEPROM data should not be written to the controller.

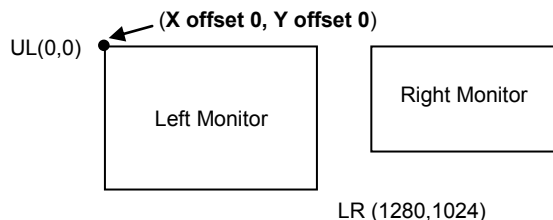
X and Y Offset

This value in pixels will select the upper-left of rectangular region where cursor movement and touch clicks can occur.

An example of configuring X and Y offset of a touch controller on right monitor: If the monitor on the left having a resolution in pixels of 1280x1024, then 1280 would be entered for the X offset and zero for the Y offset.



An example of configuring X and Y offset of a touch controller on left monitor: In most cases, both the X and Y offset for the left monitor will be zero.



Note: in order for X and Y offset to be correctly applied, the custom width and height must not be zero.

Example output of configuring X and Y offset of a touch controller on right monitor:

```
Microchip Control Panel Version 3.23
Current controller: ID: 0, Interface: SERIAL on /dev/ttyS0
```

Advanced Settings Menu

Enter X offset in pixels: 1280

```
Microchip Control Panel Version 3.23
Current controller: ID: 0, Interface: SERIAL on /dev/ttyS0
```

Advanced Settings Menu

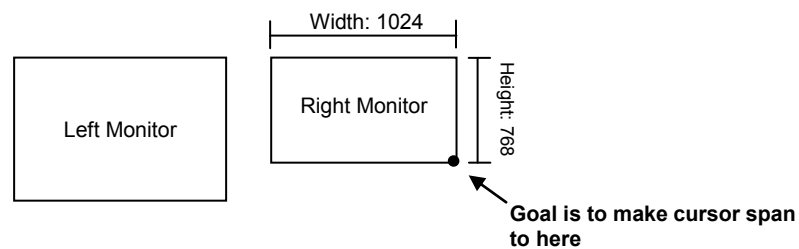
Enter Y offset in pixels: 0

Custom Width/Height Size

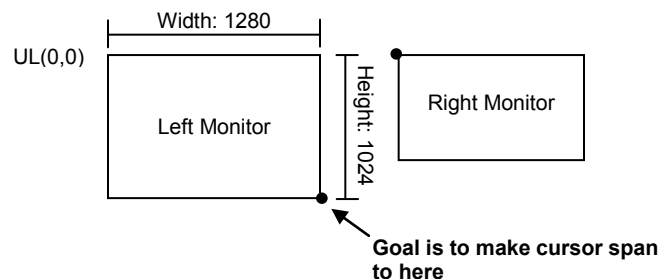
This value in pixels will specify the width and height of the rectangular region where cursor movement and touch clicks can occur.

Enter a custom height or width using these settings before calibration. This value will select the end of the range in the X or Y direction for the mouse cursor to remain in as the result of a touch. Note: The default value of 0 for width and height will cause the driver to span the full width and height of all attached displays.

An example of configuring custom width and custom height for a touch controller on right monitor: If the monitor on the right having a resolution in pixels of 1024x768, then 1024 would be entered for the custom width and 768 for the custom height.



An example of configuring custom width and custom height for a touch controller on left monitor: If the monitor on the left having a resolution in pixels of 1280x1024, then 1280 would be entered for the custom width and 1024 for the custom height.



Example output of configuring custom width/height for a touch controller on right monitor:

```
Microchip Control Panel Version 3.23
Current controller: ID: 0, Interface: SERIAL on /dev/ttyS0
```

Advanced Settings Menu

Enter custom width in pixels: 1024

```
Microchip Control Panel Version 3.23
Current controller: ID: 0, Interface: SERIAL on /dev/ttyS0
```

Advanced Settings Menu

Enter custom height in pixels: 768

Write Current Controller Data

Apply

To save settings for the current controller so that the next time the TSHARC daemon is started the current settings are used, select “Apply – Write Current Controller Data” from the Control Panel. These settings are saved in the configuration directory (normally “/etc/tsharc”) under a filename that corresponds to the current serial device path (for example “ttyS0.conf”).

Exit the Control Panel

Cancel

To exit and revert settings to their last save for the current controller, select “Cancel – Exit and Read Current Controller Data” from the Control Panel.

Save

To exit and save settings for the current controller’s configuration to be read the next time the TSHARC daemon is started, select “Ok – Exit and Write Current Controller Data” from the Control Panel.

Calibration Utility (“tscal”) Command Line Parameters

Parameter	Usage	Description
-p	-p <Number of calibration points>	Specify number of calibration points for quick calibration
-c	-c <Controller Number>	Specify controller to use. Note: First controller is zero, second is one, and so on.
-x	-x <X Offset>	Specify X offset (multi-monitor specific)
-y	-y <Y Offset>	Specify Y offset (multi-monitor specific)
-w	-w <Custom Width>	Specify custom width (multi-monitor specific)
-h	-h <Custom Height>	Specify custom height (multi-monitor specific)

The below example calls out tscal by specifying parameters such that it calibrates the second controller on the second display using a four point calibration.

```
Example output:
tscal -p 4 -c 1 -x 1280 -y 0 -w 1024 -h 768
```

Calibration Utility (“tscal”) Configuring Timeouts

The “TIME_IN_DURATION” and “TIME_OUT_DURATION” are used to modify the default GUI calibration utility times from their defaults to a different value. The below table describes an example of how it is used.

Variable	Example Command-Line Usage	Description
TIME_IN_DURATION	export TIME_IN_DURATION= < Target Timeout>	Sets the maximum number of seconds waiting for a touch on a target before exiting.
TIME_OUT_DURATION	export TIME_OUT_DURATION= < Test Timeout>	Sets the maximum number of seconds waiting for “Accept” or “Cancel” button to be touched during calibration testing before exiting.

To have these values go into effect permanently, the appropriate distribution startup script needs to be modified to automatically set these environmental variables. Below is example is an example of setting the time in duration to 30 and the time out duration to 40 under Red Hat 6.1.

```
su
echo “export TIME_IN_DURATION=30” >> /etc/profile
echo “export TIME_OUT_DURATION=40” >> /etc/profile
```

Uninstallation

To uninstall the TSHARC daemon:

- Return to command prompt
- Run “sudo uninstallTSHARC.sh”
- Select “Y” to uninstall

```
Example output:

Stopping tsharcd: Done.
Warning: Are you sure you wish to erase all TSHARC data? (y/n): y
Removing TSHARC files...
Removing files in /etc/tsharc
The TSHARC daemon files have been removed from the system.
```

Troubleshooting

Issue: If after installation the controller does not respond to touch, these additional steps may correct the problem.

Test 1:

- Enter “boot.tsharc status” at the command line to verify the daemon process is running. If it is not running, run the command “boot.tsharc start”

Test 2:

- As part of the daemon installation files, tsharc_state is an executable file that is included. This utility is used to ensure that basic communication is made with the controller.
 - Run “boot.tsharc stop” to ensure the daemon is not running during this communications test.
 - At the command prompt, change to the directory containing the installation files and run “./tsharc_state <device path>” where <device path> is the path for the device.
 - /dev/ttyS0 for COM1 serial controller
 - If coordinates do not appear as a result of a touch, try another available COM port.

Test 3:

- Enter “ls /dev/uinput” at the command line to verify that the “uinput” kernel component is setup in the manner that the TSHARC daemon expects (uinput in kernel versions shortly after 2.6.27 mapped to “/dev/input/uinput” instead of “/dev/uinput”). If “/dev/uinput” does not exist, the following command can be run to create this path.

```
mknod /dev/uinput c 10 223
modprobe uinput
```

Finally, run “boot.tsharc restart” and the cursor should now be moving correctly. If the above commands result in the cursor moving, please add these two lines to the appropriate startup script on your Linux distribution.