TSHARC™ Win CE Drivers Manual
Windows® CE version 4.2
RS-232, USB & PS/2

Rev 1.16

Document Revision and Copyright

Document Name: WinCE42Drivers116Man091906.doc
Document Date: 9/19/2006 2:58 PM

Contact Information

Hampshire Company, Inc.
9055 N. 51st Street Unit H
Brown Deer, WI 53223
Main Phone: 414-355-4675
Main Fax: 414-355-4775
www.hampshirecompany.com

Table of contents

General Notes

This Hampshire Windows CE 4.2 driver was created for CPUs and configurations shipped with CE 4.2. Additional hardware platforms may be made available upon request. Please call Hampshire to compile the drivers for your specific hardware platform.

TSHARC Controller products supported:

Hampshire TSHARC Boards	Hampshire TSHARC Chips
TSHARC-10 Octopus	TSHARC-12s
TSHARC-12 Octopus	TSHARC-10o
TSHARC-12m	TSHARC-12o
TSHARC-12v	TSHARC-8
TSHARC-8	TSHARC-12
TSHARC-12	

Supported Processors

Hampshire has recompiled the source code for the TSHARC WinCE driver to support the following processors.

ARMV4I
MIPSII
MIPSII_FP
MIPSIV
MIPSIV_FP
SH4
X86

Communications:

RS-232 Specific Information

The driver loads based on the "Port" value entry within the "TSHARCS.reg" file. This value corresponds to the nth available port. For example, if "Port"=dword:1, assigns the port as the first available port and if "Port" would equal "dword:2" then the second available port would be chosen. All of Hampshire current controllers are configured for 9600 baud.

If you are using a TSHARC-8 or TSHARC-12 Hampshire controller using other than a 9600 baud rate, the baud rate may be changed via the “Baud” registry entry which may also be found in the “TSHARCS.REG” file. In general, you will not have to change the baud rate to anything other than 9600. TSHARC-10o, TSHARC-12o, TSHARC-12v, TSHARC-12m and TSHARC-12s chips and boards are set to 9600 baud rate from the factory.

In the registry COM1 often is set to IRQ 3, IOBase 2F8 which is the base address and IRQ of COM2. This is usually because COM1 is being reserved for debugging purposes. However if the use of COM1 is desired, the following procedure may be followed:

1. Modify the PLATFORM.REG that can be found under the "ParameterView" tab
2. Find the entry [HKEY_LOCAL_MACHINE\Drivers\BuiltIn\Serial]
   The values may currently be set at COM2:
   "Irq"=dword:03
   "IoBase"=dword:02F8
   "IoLen"=dword:8
3. Change these values to the following (assuming COM1 on Irq 4 and Base 3F8):
   "Irq"=dword:04
   "IoBase"=dword:03F8
   "IoLen"=dword:8

Note: Changes on the platform.reg file will be seen on all future Windows CE images generated.

PS/2 Specific Information

The 8042 input driver must be included in the OS Design in order for the TSHARC PS/2 driver to function on CE device.

USB Specific Information

The USB host controller and the UHCI component must be installed in order for USB to function. These components are added automatically after adding the "USB TSHARC Touch Screen Driver" component to the OS Design.

Installing the Windows CE TSHARC Driver

1. Close any open instances of "Platform Builder 4.2".
2. Open Windows Explorer and browse to directory containing TSHARC installation package (MSI file).
3. Double-click on the “TSHARCCE42V116.MSI” installation package.
The TSHARC setup application will now appear.
4. Click on "Next". The license agreement will now appear.
5. Click on "Accept" and then on the "Next" Button.
6. Click on the "Install" button. The TSHARC Driver files will now be copied according to the following:

The Serial files are copied to
"<WINCEROOT>\PUBLIC\COMMON\OAK\DRIVERS\TSHARC\TSHARCS"
The PS/2 files are copied to
"<WINCEROOT>\PUBLIC\COMMON\OAK\DRIVERS\TSHARC\TSHARCP"
The USB files are copied to
"<WINCEROOT>\PUBLIC\COMMON\OAK\DRIVERS\TSHARC\TSHARCU"
Also, the TSHARC driver component is added to the Windows CE catalog.
7. Click on "Finish"

Adding a TSHARC Driver Component to an OS Design

1. Launch "Platform Builder 4.2"
2. Open or create your platform workspace. Note: The "Industrial Controller" design template that ships with Windows CE has been tested the most and verified as working correctly with all TSHARC CE drivers. However, other design templates should work as well.
3. From the "Catalog" pane, browse to "Core OS->Display base devices"
4. Right-click on TSHARC touch screen component that is to be included in the Windows CE image and then left-click on "Add to OS image". Note: Only one TSHARC interface component (Serial, PS2, USB) should be included in an OS design at any given time.

Adding shortcuts applications in an OS Design

To create a shortcut to the calibration utility, open the file"<WINCEROOT>\PUBLIC\COMMON\OAK\DRIVERS\TSHARC\HCECAL\HCECAL.DAT" and copy and paste the contents of this file to the "project.dat" file for the current project.

Removing a TSHARC Driver Component from an OS Design

1. Click on the "FeatureView" tab.
2. Browse to "Device Drivers->Input Devices".
3. Right-click on the appropriate TSHARC component:
If Serial, right-click on "Serial TSHARC Touch Screen Driver"
If PS2, right-click on "PS/2 TSHARC Touch Screen Driver"
If USB, right-click on "USB TSHARC Touch Screen Driver"
4. Left-click "Delete".

Removing TSHARC Driver Components from the Catalog

1. Open "Platform Builder" version 4.2.
2. Select "File->Manage Catalog Features…" from the applications menu system.
3. Click on "tsharc.cec" file from the list of catalog feature files.
4. Click on the "Remove" button.
5. Click on the "OK" button.
The TSHARC driver components are now removed from the Catalog.

To remove the TSHARC files as well, the directory
"<WINCEROOT>\PUBLIC\COMMON\OAK\DRIVERS\TSHARC" now can safely be deleted.

Calibrating the TSHARC Windows CE Driver on Windows CE Device

1. Select the Start menu
2. Under the "Programs" group select "TSHARC Calibration" (Note: If shortcut not present under "Programs" group, a shortcut to the calibration application may be found in under the "Windows" directory on the CE Device.
3. Touch all targets for at least a couple seconds each.
Note: The point calibration used is determined by the "CalType" registry entry described in section "Changing the Calibration Type" below.

Windows CE Calibration Registry Entries

Calibration parameters are created by Hcecal after calibration and are stored in the registry under the following key:

[HKEY_LOCAL_MACHINE\SOFTWARE\Hampshire\CurrentVersion\Global]

Upon boot, if the registry entries are not located, the driver defaults to a default uncalibrated state.

Enabling Right-Click Emulation

Method 1:

1. In "Platform Builder 4.2", click on the “FileView” tab
2. Double click on the appropriate registry file:
If Serial, edit the file
"<WINCEROOT>\PUBLIC\COMMON\OAK\DRIVERS\TSHARC\TSHARCS\TSHARCS.reg"
If PS/2, edit the file
"<WINCEROOT>\PUBLIC\COMMON\OAK\DRIVERS\TSHARC\TSHARCP\TSHARCP.reg"
If USB, edit the file
"<WINCEROOT>\PUBLIC\COMMON\OAK\DRIVERS\TSHARC\TSHARCU\TSHARCU.reg"
3. Remove the semi-colons (uncomment) from the last four lines of the file.
4. Rebuild and recreate run-time image.

Method 2:

1. Using a registry editor for the target device, change the “xEventArea”, “yEventArea” and “RightClickTime” registry items located within the “HKEY_LOCAL_MACHINE\SOFTWARE\Hampshire\CurrentVersion\Global” branch according to the description of what these values are responsible for in the next paragraph.
2. Browse to the “Windows” directory using Windows Explorer and double-click the shortcut “TouchIOCTL”. This will cause current TSHARC driver to re-read and apply the new registry settings.

There are three entries that can be configured for right-click emulation, which are "xEventArea", "yEventArea" and "RightClickTime". The "RightClickTime" is the time elapsed (measured in clock ticks) on a touch that is within an area on the screen (the event area) before a right-click event is sent. The "xEventArea" and "yEventArea" entries are horizontal and vertical components of the area that a touch remains in for a time specified by "RightClickTime" before a right-click event is sent. The Values for "xEventArea" and "yEventArea” entries can range from between 0x0000 and 0xFFFF. The "xEventArea" and "yEventArea" are relative to the touch screen coordinates rather than the screen coordinates.

To determine the values for xEventArea and yEventArea, the following formula may be used:
xEventArea = 65535 * (desired_event_area_width / width_of_touchscreen)
yEventArea = 65535 * (desired_event_area_height / height_of_touchscreen)
These values must then be converted to hexadecimal.

Note: To disable Right-Click emulation, “RightClickTime” should have of value or zero, or the registry entries “xEventArea”, “yEventArea”, and “RightClickTime” can be removed or kept absent from the registry.

Changing the Calibration Type

Method 1:

1. In "Platform Builder 4.2", click on the FileView tab
2. Double click on the appropriate registry file:
If Serial, edit the file
"<WINCEROOT>\PUBLIC\COMMON\OAK\DRIVERS\TSHARC\TSHARCS\TSHARCS.reg"
If PS/2, edit the file
"<WINCEROOT>\PUBLIC\COMMON\OAK\DRIVERS\TSHARC\TSHARCP\TSHARCP.reg"
If USB, edit the file
"<WINCEROOT>\PUBLIC\COMMON\OAK\DRIVERS\TSHARC\TSHARCU\TSHARCU.reg"
3. Change the value of "CalType" appropriately:
For 3-point calibration, set value to "dword:3"
For 4-point calibration, set value to "dword:4"
For 7-point calibration, set value to "dword:7"
For 20-point calibration, set value to "dword:14"
4. Rebuild and recreate run-time image.

Method 2:

1. Using a registry editor for the target device, change the “CalType” registry item located within the “HKEY_LOCAL_MACHINE\SOFTWARE\Hampshire\CurrentVersion\CalType” branch according to the following:
For 3-point calibration, set value to "dword:3"
For 4-point calibration, set value to "dword:4"
For 7-point calibration, set value to "dword:7"
For 20-point calibration, set value to "dword:14" (Note: 20 if using decimal)
2. Browse to the “Windows” directory using Windows Explorer and double-click the shortcut “TouchIOCTL”. This will cause current TSHARC driver to re-read and apply the new registry settings.

The TSHARC calibration application uses this value upon start to determine which calibration prompts to display.

Changing Calibration Timeout Values

There are two timeout values that are adjustable in the registry that correspond to the timeout value when first launching the calibration application (“CalTimeIn”) and the timeout value after touching all of the calibration points(“CalTimeOut”). If the countdown timer for “CalTimeIn” expires, the calibration application simply exits. If the counter for “CalTimeOut” expires then the calibration application reverts the calibration settings to the previous settings. The default value “CalTimeIn” is 15 seconds and the default value for “CalTimeOut” is 90 seconds.

Method 1:

1. In "Platform Builder 4.2", click on the “FileView” tab
2. Double click on the appropriate registry file:
If Serial, edit the file
"<WINCEROOT>\PUBLIC\COMMON\OAK\DRIVERS\TSHARC\TSHARCS\TSHARCS.reg"
If PS/2, edit the file
"<WINCEROOT>\PUBLIC\COMMON\OAK\DRIVERS\TSHARC\TSHARCP\TSHARCP.reg"
If USB, edit the file
"<WINCEROOT>\PUBLIC\COMMON\OAK\DRIVERS\TSHARC\TSHARCU\TSHARCU.reg"
3. Added the following lines to the end of the file:

[HKEY_LOCAL_MACHINE\SOFTWARE\Hampshire\CurrentVersion\Global]
"CalTimeIn"=dword:F
“CalTimeOut”=dword:5A

4. Rebuild and recreate run-time image.

Method 2:

1. Using a registry editor for the target device, change the “CalTimeIn”, “CalTimeOut” registry items located within the “HKEY_LOCAL_MACHINE\SOFTWARE\Hampshire\CurrentVersion\Global” branch according to timeout value(s) that are desired.
2. Browse to the “Windows” directory using Windows Explorer and double-click the shortcut “TouchIOCTL”. This will cause current TSHARC driver to re-read and apply the new registry settings.

Changing the mouse event stream rate

When the mouse cursor moves and changes state as a result of a touch, the TSHARC CE driver sends a series of mouse events to the Windows CE event queue. If the mouse events are sent too fast, other CE applications and background process may take longer to process their own events while the touch screen is being touched. For this reason, a new feature has been added to the TSHARC Serial and USB drivers to allow adjustment of the speed at which the driver sends mouse events. The default value is a minimum duration 16 clock ticks for every mouse packet (except for pen up and pen down packets which are always processed immediately). For slower embedded devices, it may be desirable to increase this value to decrease to rate at which mouse packets are sent. For faster devices, it may be desirable to decrease this value to increase the rate at which mouse packets are sent.

Method 1:

1. In "Platform Builder 4.2", click on the FileView tab
2. Double click on the appropriate registry file:
If Serial, edit the file
"<WINCEROOT>\PUBLIC\COMMON\OAK\DRIVERS\TSHARC\TSHARCS\TSHARCS.reg"
If PS/2, edit the file
"<WINCEROOT>\PUBLIC\COMMON\OAK\DRIVERS\TSHARC\TSHARCP\TSHARCP.reg"
If USB, edit the file
"<WINCEROOT>\PUBLIC\COMMON\OAK\DRIVERS\TSHARC\TSHARCU\TSHARCU.reg"
3. Add the following lines:

[HKEY_LOCAL_MACHINE\SOFTWARE\Hampshire\CurrentVersion\Global]
"mouseMoveThreshold"=dword:10

4. Rebuild and recreate run-time image.

Method 2:

1. Using a registry editor for the target device, add or change the “mouseMoveThreshold”, registry item located within the “HKEY_LOCAL_MACHINE\SOFTWARE\Hampshire\CurrentVersion\Global” branch to the desired minimum duration length to be applied for every mouse stream packet.
2. Browse to the “Windows” directory using Windows Explorer and double-click the shortcut “TouchIOCTL”. This will cause current TSHARC driver to re-read and apply the new registry settings.

Saving registry settings to Persistent storage such as Compact Flash

After calibrating the touch screen controller the calibration registry settings are automatically saved to the persistent storage if persistent storage is setup correctly on the CE device. However, if the TSHARC registry settings are edited using a registry editor or third-party application, the new registry settings may be saved using two methods.

Method 1:

Click “Start->Suspend” on the Windows CE target.

Method 2:

If changing the registry settings programmatically, call RegFlushKey() function in source code after making registry modifications.