



Windows Embedded Compact Test Kit User Guide

Windows Embedded Compact 7 Technical Article

Published: March 2011

Applies To: Windows Embedded Compact 7

Abstract

The Windows Embedded Compact Test Kit (CTK) is a tool that you can use to test the functionality and performance of device drivers and related hardware for a Windows Embedded Compact device. The test tools in the CTK provide feedback on the functionality of your drivers, which can in turn be used to further enhance the reliability of a Windows Embedded Compact device platform. This paper describes how to use the Windows Embedded Compact Test Kit.

Introduction to the Windows Embedded Compact Test Kit

The Windows Embedded Compact Test Kit (CTK) provides significant improvements to the user interface and the overall feature set when compared to the previous release known as Windows Embedded CE Test Kit (CETK).

The Windows Embedded CTK is a tool that you can use to test the functionality and performance of device drivers and related hardware for a Windows Embedded Compact device. The test tools in the CTK provide feedback on the functionality of your drivers, which can in turn be used to further enhance the reliability of your device. Additionally, more tests can be added to the CTK to test specific drivers.

You can use the integrated Graph Tool to convert performance test results into graphical charts. For information about the Graph Tool, see [Using the Graph Tool](#).

Prerequisites

The Windows Embedded CTK requires the following:

- Microsoft Windows Embedded Platform Builder 7
- Windows Embedded Compact 7
- A hardware device to test or Virtual CEPC
- Microsoft .NET Compact Framework 2.0 SP2

The Graph Tool requires the following:

- Microsoft Office 2007 or Microsoft Office 2010 (The Graph Tool is not compatible with earlier versions of Office)

What's New

The Windows Embedded CTK is a new application with the following key features:

- Improved user interface that resembles Visual Studio
- Ability to group user-selected test cases into test passes
- Improved test results viewer lets you store and review test results per test pass
- Upon connection to a device, detection of peripherals and drivers needed for tests
- Integration with the new Graph Tool to graph performance test results
- Users can add custom Tux test harness (TUX)-based tests to the CTK
- Support for x86, MIPSII, MIPSII_FP, ARMv5, ARMv6, ARMv7, and SH4 processors

Improved User Interface

The Windows Embedded Compact Test Kit (CTK) user interface (UI) consists of an integrated set of windows, tools, menus, toolbars, and other elements that allow you to execute, view and manage your Windows Embedded Compact tests in one place.

The user interface uses standard Windows interface functionality that resembles the Visual Studio UI. It displays multiple windows that show information and data pertinent to test cases.

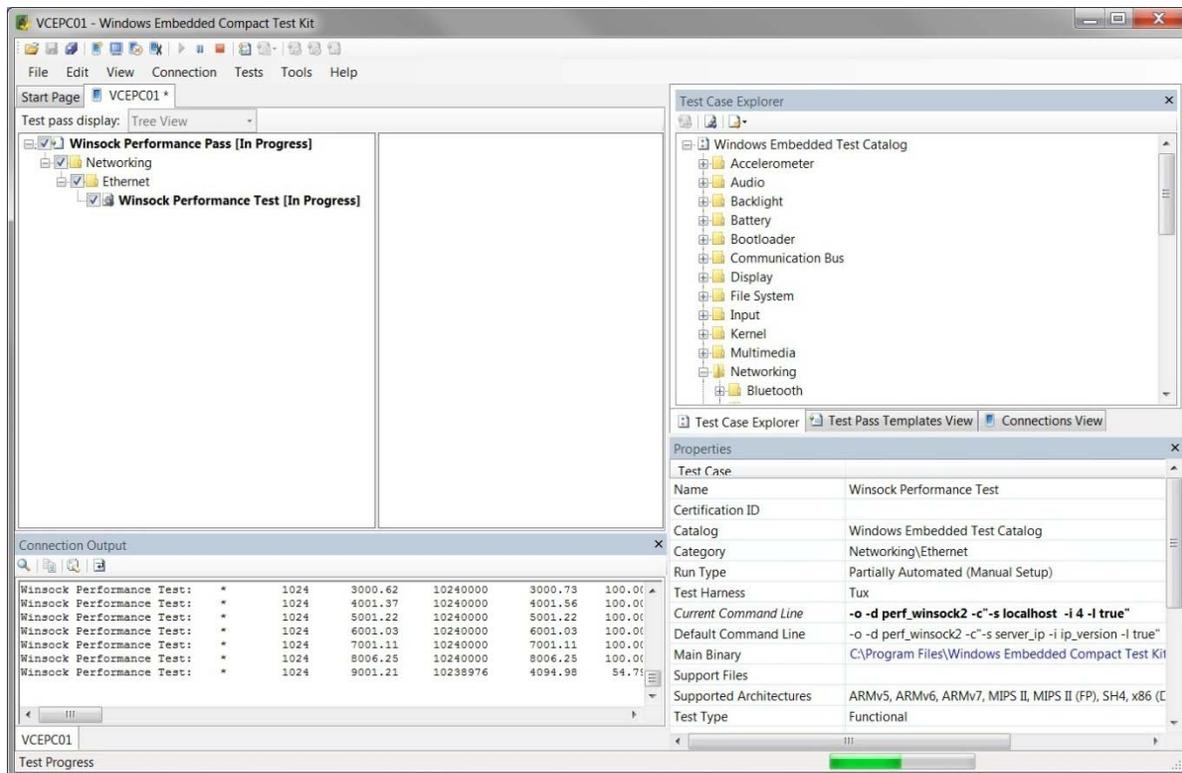


Figure 1 - The Windows Embedded Compact Test Kit

The Windows Embedded Compact Test Kit window, shown in **Figure 1**, is made up of an icon toolbar at the top, a menu toolbar, four primary windows, and a status bar at the bottom of the window.

The four primary windows are as follows:

1. The **Test pass display** window in the upper left displays the current test pass and is divided into two parts:
 - The left side of the window displays a test pass in either **Tree View** or a flat **List View**.
 - The right side of the window displays the Help documentation for the test case currently selected in the left side.
2. The **Test Case Explorer** window in the upper right displays the following tabs:
 - The **Test Case Explorer** tab displays the master test catalog in a tree view. It contains the Windows Embedded test catalog by default. Other catalogs may be added in the future. For more information, see [Windows Embedded Test Catalog](#).
 - The **Test Pass Template View** tab displays the user-created and built-in test pass templates. For more information on test pass templates, see [Creating a Test Pass Template](#).

- The **Connection View** tab displays the device the CTK is connected to. For more information, see [Connecting to a Device](#).
3. The **Properties** window in the lower right is similar to the **Properties** windows in Visual Studio. It displays the properties of the currently selected test case, test pass, template, or connection. For more information, see [Viewing Test Properties](#).
 4. The **Output** window in the lower left displays debug messages from the CTK and from a test case as it runs.

Test Pass Icons

The following icons are displayed next to test cases in the **Test pass display** window to describe the level of interaction needed to run the tests.

Icon	Description
	Fully automated
	Partially automated, requires some manual interaction
	Manual, requires total manual interaction
	The attached device does not have the required peripherals required to run the test as determined by peripheral detection

Quick Start

This section gives a quick introduction to using the Windows Embedded Compact Test Kit (CTK). The following steps guide you through the process of connecting to a device, creating a test pass, running a simple test, and viewing the results of the test.

- [Step 1 – Connect to a Device](#)
- [Step 2 - Create a Test Pass Template](#)
- [Step 3 – Create a Test Pass from the Template](#)
- [Step 4 – Run a Test](#)
- [Step 5 – View Test Results](#)

Step 1 – Connect to a Device

A device must be running and attached to Platform Builder before you can connect to the device from the Windows Embedded CTK. If you prefer, you can create a test pass template before you connect to a device.

To connect to a device

1. In Windows Embedded Compact Test Kit, on the **Connection** menu, click **Connect to Device**.
2. Select the device you want to connect to from the **Select a Device** dialog box and then click **OK**.

The **Connecting to a Device** dialog will appear displaying the name of the device you are trying to connect to and the progress of the connection attempt. If the connection to the device is successful, the **Connection Output** window displays the "Successfully established connection!" message. If a device has previously been connected to, the **Test pass display** window opens and displays "No test pass" or a default test pass."

3. You are now ready to create a test pass template.

For more information, see [Connecting to a Device](#).

Step 2 – Create a Test Pass Template

Before creating a test pass you must create a template to base the test pass on.

To create a test pass template

1. On the **File** menu, point to **New**, and then click **Test Pass Template**.
The **Test Manager** window appears, displaying the **Test Case Explorer** on the left side of the window and two tabs on the right side, **Test Case Information** and **Test Pass Templates**.
2. On the **Test Pass Templates** tab, under **Edit Templates**, click the **New** button.
3. In the new template name text box that appears, type **Registry APIs** to rename the template from **New Template**.
4. In the **Test Manager Test Case Explorer** tree view display area, double-click **Windows Embedded Test Catalog** to expand the catalog tree.
5. In the catalog tree, double-click the **File System** folder, and then double-click **Registry**.
6. Click **Registry API and Functionality Test**, and then click the **Add >** button to add the selected test to the new **Registry APIs** template.
7. Click **Done** to exit **Test Manager**. Your template will be saved.
8. You are now ready to create a test pass from the template you just created.

For more information, see [Creating a Test Pass Template](#).

Step 3 – Create a Test Pass from the Template

Once you have created the test pass template and connected to a device, you can create a test pass.

To create a test pass from a template

1. On the **View** menu, click **Test Pass Templates View**.
2. In the **Test Case Explorer** window **Test Pass Templates View** tab, click the **Registry APIs** template.

3. Right-click the highlighted **Registry APIs** template and then click **Use as Current Test Pass** on the pop-up menu.
Note If **Use as Current Test Pass** is not available on the pop-up menu, you most likely are not connected to a device.
Note If a test pass was already in the **Test Pass** display window, you'll be asked if you want to save the current test pass before a new test pass is created.
 4. In the **Test Pass** display window, click the new test pass **Windows Embedded** and type **Registry API Test Pass** to rename it.
- For more information, see [Creating a Test Pass from a Template](#).

Step 4 – Run a Test

You can run the test once **Registry API and Functionality Test** has been added to a test pass.

To run a test

1. In the **Test pass display** window, click the plus sign (+) preceding **Registry API Test Pass** to expand the test pass tree.
2. Right-click on **Registry API and Functionality Test** and click **Run Test**.
3. When the **Test pass display** window changes from **Registry API and Functionality Test [In Progress]** to **Registry API and Functionality Test [Passed]**, the test run is complete and you are ready to view the results of the completed test.

The green status bar at the bottom of the Windows Embedded CTK main window displays the progress of the test. As the test runs, Kernel Independent Transport Layer (KITL) debug message output is displayed in the **Connection Output** window.

For more information, see [Running a Test Pass](#).

Step 5 – View Test Results

Once the test run is complete you are ready to view the results of the test.

To view the results for a test

1. In the **Test pass display** window, select **Registry API and Functionality Test [Passed]**, right-click and then select **View Test Results**.
2. In the **Common Results View** window tab that opens, locate the completed test run in the **Test Case Results History** table, and click the folder icon in the **Log File(s)** column.
3. In the **Windows Explorer** window that opens, locate in the folder containing the log files for the test run and open the results.log file to view the test debug output in Tux test harness (Tux) format.

Note The test result logs are stored using the following folder hierarchy:
C:\Users\\Compact Test Kit\results

For more information, see [Viewing Test Results](#).

Connecting to a Device

Before trying to connect to a device from the Windows Embedded Compact Test Kit (CTK), your device must be running and attached to Platform Builder.

To connect to a device

1. In Windows Embedded Compact Test Kit, on the **Connection** menu, click **Connect to Device**.
2. In the **Select a Device** dialog box, select the device you want to connect to and then click **OK**.

The **Connecting to a Device** dialog will appear displaying the name of the device you are trying to connect to and the progress of the connection attempt. If the connection to the device is successful, the **Connection Output** window displays the "Successfully established connection!" message. The **Test pass display** window opens and displays "No test pass" or a default test pass.

3. Do one of the following:
 - Create a new test pass. For more information, see [Creating a Test Pass](#).
 - Run the default test pass. For more information, see [Running a Test Pass](#).
 - Add to the default test pass. For more information, see [Adding a Test Case to a Test Pass](#).

To disconnect a device

- In Windows Embedded Compact Test Kit, on the **Connection** menu, click **Remove Connection**.

To troubleshoot a device connection

- View the messages displayed in the **Connection Output** window.
- View the messages in Platform Builder in the Windows CE Debug **Output** window.
- Verify that the CPU type of the device that you are trying to connect to is supported by the CTK. Verification of the CPU type can be attained if you have access to your source code and by viewing what the CEInstructionSet is set to. If you have Windows Embedded Compact 7 installed, this information is located in the file, **C:\WINCE700\platform\<platform>\src\oal\oalib\init.c**.
- Verify that the device you are trying to connect to is running and attached to Platform Builder.

Creating a Test Pass

Before you create a test pass, you must create a template to base your test pass on. This section describes what a test pass is, how to create a test pass template, and how to create a test pass from a template.

What Is a Test Pass

The Windows Embedded Compact Test Kit uses the concept of a "test pass" as a container for a collection of test suites, which are collections of test cases. The test cases in a test pass can be run all together in one pass or run selectively.

A test pass provides a way to organize your testing according to the way you run your tests. For example, you could have one test pass called SetTopBox that contains all of the multimedia, file system, Ethernet, and display tests that you want to run on your device platform. Then you could have another test pass called SetTopBox_Multimedia that contains only the multimedia tests. You would use the SetTopBox_Multimedia test pass if you want to run only the multimedia tests and the SetTopBox test pass if you want to run the complete set of tests.

A test pass can be saved and reused at a later time. Saved test passes will be preserved during upgrades to new versions of the Windows Embedded Compact Test Kit.

To create a test pass you first create a test pass template to base your test pass on.

Creating a Test Pass Template

A test pass is created from a test pass template. Once you've created a test pass template you can add and remove test cases from the template.

You can organize test cases into logically related test pass groups using the Windows Embedded Compact Test Kit (CTK) to create test pass templates. You can view these groups in the **Test Case Explorer** on the **Test Pass Templates View** tab.

There are two categories of test pass templates:

1. **Microsoft Windows Embedded Templates**, which are provided by Microsoft and organized into the following groups: Basic Verification, Functional, Performance, and Windows Embedded Compact.
 - The Basic Verification, Functional, and Performance templates are subsets of all tests available in the CTK. The selection of the tests that make up each test pass template is based on relevance to the goal of that test pass.
 - The Windows Embedded Compact template contains all the tests available in the CTK.
2. **My Templates**, which are all the templates created by you.

You cannot edit the **Microsoft Windows Embedded Templates**, but you can edit the **My Templates** that you create using the Test Manager. For more information about the Test Manager, see [Managing Tests](#).

Creating a New Test Pass Template

You use the Test Manager to create new and edit existing test pass templates. For more information about the Test Manager, see [Managing Tests](#).

To create a new test pass template

1. In Windows Embedded Compact Test Kit, on the **File** menu, point to **New**, and then click **Test Pass Template**.
2. On the right side of the **Test Manager** window that appears, on the **Test Pass Templates** tab, under **Edit Templates**, click the **New** button.
3. In the new template that appears, type the name you want to rename the template.

4. In the **Test Case Explorer** tree view display area, double-click **Windows Embedded Test Catalog** to expand the catalog tree.
5. Click on a test case tree node or an individual test case, and then click the **Add >** button to add the selected test case to your new template.
6. Continue to add test cases from the **Windows Embedded Test Catalog** until your template is complete.
7. If a test case needs to be removed, click on a test case tree node or an individual test case in your new template to highlight the node or test case you want to remove, then click the **< Remove** button.
8. Click **Done** to exit **Test Manager**.

Copying an Existing Test Pass Template

The Windows Embedded Compact Test Kit (CTK) provides prebuilt test pass templates that you can copy to create your own templates. The provided templates are the Windows Embedded Compact template that is based on the test cases in the Windows Embedded Test catalog and a Desktop template. You can copy test pass templates that you create as well.

To copy an existing test pass template

1. In Windows Embedded Compact Test Kit, on the **File** menu, point to **New**, and then click **Test Pass Template**.

The **Test Manager** window opens displaying the **Test Case Explorer** on the left side of the window and two tabs on the right side: **Test Case Information** and **Test Pass Templates**.

2. On the **Test Pass Templates** tab, click to highlight the template you want to copy.

Note You cannot copy test cases from a template. You can only copy the template as a whole.

3. Under **Edit Templates**, click the **Copy** button.
4. In the copied template that appears, type the name you want to rename the copied template.
5. If you want to customize the template, you can add and remove test cases.
6. Click **Done** to exit **Test Manager**.

Creating a Test Pass from a Template

Once you have a created a test pass template, you can create a test pass.

To create a test pass from a template

1. In Windows Embedded Compact Test Kit, connect to a device. For more information, see [Connecting to a Device](#).
2. On the **View** menu, click **Test Pass Templates View**.
3. In the **Test Case Explorer** window **Test Pass Templates View** tab, click to highlight the test pass template you want to base your test pass on.
4. Right-click the highlighted template and then click **Use as Current Test Pass** on the pop-up menu.

Note If **Use as Current Test Pass** is not available on the pop-up menu, you most likely are not connected to a device.

5. In the **Test Pass** display window, click the new test **Windows Embedded** that is highlighted and type a new name.

Note If a test pass was already in the **Test Pass** display window, you'll be asked if you want to save the current test pass before a new test pass is created.

6. Click the plus sign (+) preceding the test pass name to expand the test pass tree.

The new test pass contains all the test cases that are in the template used to create the test pass.

7. On the **File** menu, click **Save Test Pass**.

Adding a Test Case to a Test Pass

You can add a test case directly to a test pass without adding the test case to a template.

To add a test case to a test pass

1. In Windows Embedded Compact Test Kit, connect to a device. For more information, see [Connecting to a Device](#).
2. On the **File** menu, point to **Open**, and then click **Test Pass**.
3. In the **Open** dialog box, select a test pass file and then click **Open**.

Note If a test pass was already in the **Test pass display** window, you'll be asked if you want to save the current test pass before a new test pass is created.
4. In the **Test pass display** window, on the **View** menu, click **Test Case Explorer**.
5. In the **Text Case Explorer** window, select a test case node or a test case from the **Windows Embedded Test Catalog** tree.
6. Drag and drop the selected test case or test case node onto the **Test Pass** display window. If you select a test case node, the node and all children of the node are copied to the current test pass.
7. To save the test pass, on the **File** menu, click **Save Test Pass**.

Running a Test Pass

There are several options for running a test from the currently active test pass in the **Test pass display** window.

- You can run all of the tests in the test pass, which are run sequentially. This is the default option unless you deselect specific tests.
- Choose which specific tests you want to run. To select the test to be run, select the check box next to the individual test case or test case node.
- Select and run only one test or a particular test node.

Running All Tests in a Test Pass

To run a test, you first need to create a new test pass or open an existing test pass. For information about creating a new test pass, see [Creating a Test Pass](#).

To run all the tests in a test pass

1. In Windows Embedded Compact Test Kit, connect to a device. For more information, see [Connecting to a Device](#).
2. On the **File** menu, point to **Open**, and then click **Test Pass**.
3. Select a test pass file in the **Open** dialog box, and then click **Open**.
4. In the **Test Pass** display window, on the **Tests** menu, click **Run Test Pass**, or press the **F5** key.

All the tests in the test pass will run, sequentially from the top down.

Running Select Test Cases

You can choose which specific tests you want to run during the test pass run.

To run multiple selected test cases in the test pass

1. In Windows Embedded Compact Test Kit, connect to a device. For more information, see [Connecting to a Device](#).
2. On the **File** menu, point to **Open**, and then click **Test Pass**.
3. In the **Open** dialog box, select a test pass file and then click **Open**.
4. In the **Test Pass** display window, select the checkboxes of the test cases or test case nodes you want to run. Deselect the checkboxes of the test cases or test case nodes that you do not want to run.
5. On the **Test** menu, click **Run Test Pass**, or press the **F5** key.

All the selected test cases in the test pass will run, sequentially from the top down.

To run one specific test in the test pass

1. In Windows Embedded Compact Test Kit, connect to a device. For more information, see [Connecting to a Device](#).
2. On the **File** menu, point to **Open**, and then click **Test Pass**.
3. In the **Open** dialog box, select a test pass file and then click **Open**.
4. In the **Test Pass** display window, select the individual test case or test case node you want to run.
5. Right-click the selection and click **Run Test**.

The selected test or all of the tests in the selected node will run, sequentially from the top down.

Viewing Test Progress

As each test in the test pass is run, the **Test pass display** window shows the status of each test appended to each test name in brackets. For example, if a test named **DirectDraw Performance Test** is running the name is displayed as **DirectDraw Performance Test [In Progress]**. Also, the green status bar at the bottom of the

Windows Embedded Compact Test Kit main window displays the progress of the test.

Once a test has completed its run, the results of the tests are displayed in the brackets appended to the test name. For example, **DirectDraw Performance Test [Passed]** or **DirectDraw Performance Test [Failed]**. In addition, the text color of the test name is changed to green if the test passed or red if the test fails.

As the tests are run, the Kernel Independent Transport Layer (KITL) debug message output is displayed in the **Connection Output** window.

Stopping a Test

You can stop a test in progress.

To stop a test

- On the **Test** menu, click **Stop Test Pass**.

Viewing Test Properties

Once you select a test case in the test pass, you can view the properties of the test case.

To view test properties

1. In Windows Embedded Compact Test Kit, connect to a device. For more information, see [Connecting to a Device](#).
2. On the **File** menu, point to **Open**, and then click **Test Pass**.
3. In the **Open** dialog box, select a test pass file and then click **Open**.
4. On the test pass list, click on the test you want to view the properties for.
5. If the **Properties** window is not visible, on the **View** menu, click **Properties Window**.

The **Properties Window** displays properties for the selected test.

Changing Command-Line Options

You can change the command-line options that a test case will use. For information on Tux test harness (TUX) command-line parameters, see [Tux Command-Line Parameters](#) (<http://go.microsoft.com/fwlink/?LinkID=192154&clcid=0x409>).

To change the command-line option for a test

1. In Windows Embedded Compact Test Kit, connect to a device. For more information, see [Connecting to a Device](#).
2. On the **File** menu, point to **Open**, and then click **Test Pass**.
3. In the **Open** dialog box, select a test pass file and then click **Open**.
4. On the test pass list, click on the test case you want to change.
5. If the **Properties** window is not visible, on the **View** menu, click **Properties Window**.
6. In the **Properties** window, select the **Current Command Line** text box.
7. In the text box, type or modify the command line text to be used when you run the test.
8. On the **File** menu, click **Save Test Pass**.

Viewing Test Results

After all of the tests have completed running, the **Test pass display** window shows whether each test passed or failed. Test logs and results are saved for each test.

To view the overall results for a test pass

1. In the Windows Embedded Compact Test Kit toolbar, click on the **View Results for the Selected Test Pass** icon.
2. View the results for each test and a summary of the results for all the tests run in the test pass in the **Common Results View** window tab.
3. In the **Test Pass Results** table, click the folder icon in the **Log File(s)** column.
4. In the Windows Explorer window that opens, open the results.log file to view the test debug output in Tux test harness (TUX) format.

Note The test result logs are stored using the following folder hierarchy:

C:\Users\\Compact Test Kit\results

5. In the **Test Pass Results** table, double-click a test results row to open a **Common Results View** window tab showing the detailed history of the runs for the test case.

To view the results for a test

1. In the **Test pass display** window, select the test, right-click and then select **View Test Results**.
2. In the **Common Results View** window tab that opens, in the **Test Case Results History** table, in the **Log File(s)** column, click the folder icon.
3. In the **Windows Explorer** window that opens, open the results.log file to view the test debug output in TUX format.

Note The test result logs are stored using the following folder hierarchy:

C:\Users\\Compact Test Kit\results

Managing Tests

You can add your own test cases to a test case catalog and create test pass templates using the Test Manager.

To open the Test Manager

1. In Windows Embedded Compact Test Kit, on the **Tests** menu, click **Manage Tests**.
2. In the **Test Manager** window that opens, click the **Test Case Information** tab on the right to add a test case to a test catalog or to edit the properties of an existing text case. For more information, see [Adding a Test Case to Test Catalog](#).
3. Click the **Test Pass Templates** tab to create a new test pass template or to modify existing templates. For more information, see [Creating a Test Pass Template](#).
4. Click **Done** to exit the Test Manager.

Adding a Test Case to a Test Catalog

You can add your own test cases to a test case catalog using the Test Manager.

To add a test case to a catalog

1. In Windows Embedded Compact Test Kit, on the **File** menu, point to **New**, and then click **Test Case**.
2. In the **Test Manager** window that opens, on the **Test Case Information** tab on the right, click the **New** button.
3. Under the Selected test case information heading, do the following:
 - a. In the **Name** box, type a name for your test case.
 - b. In the **Catalog** list, select the catalog to add the test to. By default, Windows Embedded Test Catalog is selected.
 - c. In the **Category** list, select a category for the test case.
 - d. In the **Run type** list, select a type of run to use for the test case: **Fully Automated**, **Partially Automated**, or **Manual**.

If you select **Automated**, in the **Test harness** list, select the test harness to use for the automated run: **Execution Engine**, **TUX**, or **Tux.net**.
 - e. In the **Command line** box, type a new command if you do not want to use the default command.
 - f. Under **Supported architectures**, select the check box for each CPU architecture you want your test case to support.

Each time you select a check box, an **Add/Remove Test Files** dialog box opens. In the **Add/Remove Test Files** dialog box, select the main binary file for the selected architecture and add support files.
4. Click the **Advanced** button, in the **Advanced Test Information** dialog box, and then do the following:
 - a. In the **Test type** list, select a test type for your test case.
 - b. In the **Run sub-type** list, select a run sub-type.
 - c. In the **Required peripherals** list, select required peripherals to use in peripheral detection.
 - d. In the **Cancel command line** box, type a command line to use to stop the test.
 - e. If the test is automated or semi-automated and run from the CTK UI, in the **List command line** box, type a command line to use for running the test from the Test Pass list.
 - f. If the test requires a server to run on the desktop computer, in the **Server command** box, type a server application to run.
 - g. (Optional) In the **Server command line** box, type a server command line command.
 - h. Select the **Enable connection heartbeat** check box to enable continuous check to see if the test device is connected.
 - i. In the **Description** box, type descriptive notes about the test case.
5. Click **OK**.
6. Click **Save** to add the test case to the catalog you selected.

Windows Embedded Test Catalog

The Windows Embedded test catalog is the master list of all of the test cases included in the CTK. Each test area, such as Display or Networking, contains a number of individual tests that can be selected and run. The current Windows Embedded test catalog includes the following tests:

- Accelerometer
- Audio
- Backlight
- Battery
- Bootloader
- Communication Bus
- Display
- File System
- Input
- Kernel
- Multimedia
- Networking
- NLED (Notification light-emitting diode)
- OAL (OEM Abstraction Layer)
- Power Manager
- Security
- Shell
- Smart Card
- Storage Media
- USB

Detecting Peripherals and Packages

In the Windows Embedded Compact Test Kit (CTK) you can enable or disable peripheral detection. If peripheral detection is enabled, when connecting to a device the CTK checks the device for peripherals that each test case requires.

In the **Test pass display** window a “not detected” icon (a red circle with a line through it) is displayed on each test case folder where the tests require peripherals that the attached device does not have. For more information, see [Test Pass Icons](#).

The peripherals are specified in **Required peripherals** for a test case when using the **Test Manager** to add a new or edit an existing test case. The required peripherals for a test case can be viewed in the **Needed Peripherals** property. For more information, see [Adding a Test Case to a Test Catalog](#).

To enable peripheral detection

- In Windows Embedded Compact Test Kit, on the **Connection** menu, click **Peripheral Detection**.

Package detection is the same as peripheral detection, except it checks for required software packages instead of peripherals. Package detection is not currently used much in the test cases in the Windows Embedded Test Catalog. The required packages for a test case can be viewed in the **Needed Packages** property.

Using the Graph Tool

You can use the Graph Tool to graphically display the results of key performance tests such as the Winsock, USB, and Bluetooth performance tests. The Graph Tool can be run from within the Windows Embedded Compact Test Kit (CTK) or from the console command line. You can use the console version of the Graph Tool if you are running the supported performance test through Platform Builder and not the Windows Embedded Compact Test Kit (CTK). Use this method to process the perf results log and generate the graph. For more information on the console option, see [Running the Graph Tool as a Console Application](#).

The Graph Tool can be run from within the Windows Embedded CTK in two different ways: integrated as part of viewing the results for a test or run directly from the **Tools** menu.

Viewing Performance Test Results Graphically

The Graph Tool can be run from within the Windows Embedded CTK as part of the process of viewing test results for the Winsock, USB, and Bluetooth performance tests.

To view performance test results graphically

1. In the Windows Embedded CTK, connect to a device. For more information, see [Connecting to a Device](#).
2. Open a performance test, such as the Winsock Performance Test. For more information, see [Running the Winsock Performance Test](#).
3. On the **File** menu, point to **Open**, and then click **Test Pass**.
4. In the **Open** dialog box, select a test pass file and then click **Open**.
Note If a test pass was already in the **Test Pass** display window, you'll be asked if you want to save the current test pass before a new test pass is created.
5. In the **Test Pass** display window, select the checkboxes of the test cases or test case nodes you want to run.
6. On the **Test** menu, click **Run Test Pass**, or press the **F5** key.
7. Once the test pass has completed running, click on the **View Results for the Selected Test Pass** icon in the toolbar or right-click on the individual test and click **View Results**.
8. On the **Common Results View** window tab, in the **Test Pass Results** table, double-click the test results row for the successfully complete performance test.
9. On the **Graphical Results View** window tab, in the **Test Case Results History** table, in the **Graph View** column, click the document icon.
10. In the window that opens, view the test results data in graphical format.

Running the Graph Tool from the Tools Menu

In addition to the Graph Tool running within the Windows Embedded CTK as part of the process of viewing test results, the Graph Tool can be run directly from the **Tools** menu.

To run the Graph Tool from the Tools menu

1. In Windows Embedded Compact Test Kit, on the **Tools** menu, click **Graphical Results View**.
2. In the **Graphical Performance View** dialog box, on the **Simple** tab, do the following:
 - a. In the **Test Dll Name** list, select the configuration file for the type of performance test results to graph. For example, if you are graphing the Winsock Performance Test results, select **perf_winsock2**.
 - b. Click the **Open Notebook** icon to view the selected configuration file.
 - c. Next to **Input File**, click the **Open Folder** icon.
 - d. In the **Add File** dialog box, select the input log file to use for graphing data.
 - e. Change the folder path in **Output** or keep the default path.
3. (Optional) Click the **Advanced** tab to add the following additional features to your graph:

To add this optional feature	Do the following
To graph how performance varies over time	<ol style="list-style-type: none"> a. Select Trend Analysis to graph multiple logs. b. Next to Input File(s) click the open folder icon. c. In the Add File dialog box, select an input log file to use. d. Click Add more logs to add more input logs for the trend analysis. You can use a maximum of 10 input log files.
To graph multiple logs using one test log as a benchmark	<ol style="list-style-type: none"> a. Next to Input File(s) click the open folder icon. b. In the Add file dialog box, select an input log file to use. c. Click Add more logs to add more input logs. d. Next to Benchmark File click the open folder icon. e. In the Add file dialog box, select an input log file to use as a benchmark.
To run Microsoft Excel once the graphs are drawn	<ul style="list-style-type: none"> • Select Launch Excel. This option is selected by default.

To output graphs in an MIME HTML (MHTML) web page archive format	<ul style="list-style-type: none"> • Select mht.
To output graphs as JPEG image files	<ul style="list-style-type: none"> • Select Images.
To select a line, plotted point, or bar graph	<ul style="list-style-type: none"> • Select Graph Type.
To set the font size	<ul style="list-style-type: none"> • Select a Font Size.

4. Do one of the following:
 - Click the **Run** to generate graphs.
 - Click **Reset** to reset to the default setting.
 - Click **Close** to exit.

Appendices

The appendices contain additional information about the Windows Embedded Compact Test Kit (CTK). The following sections provide instructions for running test programs using a command line and instructions for running a Winsock performance test.

- [How to Run CTK Tests Manually Through Platform Builder](#)
- [Running the Graph Tool as a Console Application](#)
- [Running the Winsock Performance Test](#)

How to Run CTK Tests Manually Through Platform Builder

The Windows Embedded Compact Test Kit (CTK) can be run using Platform Builder.

To run the Windows Embedded CTK tests manually using the command line in Platform Builder

1. To use necessary test binary files, do one of the following:
 - Copy the specific test binary files that you need to run a particular test to your OS release directory.
 - a. From **%CTK_InstallRoot%\Windows Embedded Compact Test Kit\tests\target\, copy the Tux test harness (TUX) test DLL and any dependent binaries to your working release directory.**
 - b. From **%CTK_InstallRoot%\Windows Embedded Compact Test Kit\harnesses\target\, copy the binary files for the TUX harness.**
 - In Platform Builder set the **Alternate Release Directories** to point to the directories that contain test binary files you need to run a particular test.
 - a. In Platform Builder on the **Target** menu, click **Alternate Release Directories**.
 - b. In **Alternate Release Directories** window, click the **Add New Directory** folder icon.

- c. In the **Browse for Folder** dialog box, select the folders listed below and then click **OK**.
 - **%CTK_InstallRoot%\Windows Embedded Compact Test Kit\tests\target**
 - **%CTK_InstallRoot%\Windows Embedded Compact Test Kit\harnesses\target**
2. Copy the default command lines for the tests from one of the locations below:
 - The test kit help documentation TestKit.chm located at **%CTK_InstallRoot%\Program Files\Windows Embedded Compact Test Kit\Help**.
 - The CTK **Properties** window.
3. In Platform Builder on the **Target** menu, click **Target Control**.
4. In the **Windows CE** command prompt window, paste the command line.
5. Run the test and view debug output in the Platform Builder's **Output** window.

Running the Graph Tool as a Console Application

In addition to running the Graph Tool from within the Windows Embedded Compact Test Kit (CTK), you can run the Graph Tool as a console application.

To run the Graph Tool from the console command line

1. Open a command prompt window.
2. Navigate to the folder where the Graph Tool is installed. The default location is **C:\Program Files\Windows Embedded Compact Test Kit\GraphTool\bin**.
 - Note** By default the Graph Tool configuration files are located in **C:\Program Files\Windows Embedded Compact Test Kit\GraphTool\ConfigFiles**.
3. Type the following command, replacing placeholder text in with your specific test information:

```
GraphToolConsole.exe -c <config_file> -i <input_log_file> -t  
<test_dll_name>
```

Graph Tool Command-Line Parameters

This section describes the usage syntax and command-line parameters for the Graph Tool console application.

```
GraphToolConsole -c <Graph Tool Config File> -i <Input File> -t <Test name>  
[-o <Output File(xlsx)>] [-b <Benchmark file>][-trend][-image <path>][-ct  
<Chart Type>][-XlLaunch][-mht <Output File(mht)>][-font <Font Type>]
```

-c <Graph Tool Config File> : (Input) Path to the graph tool configuration file(xml)

-t <Test name> : (Input) Test DLL name from used tp generate logs

-i <Input file> : (Input) Input log file(s)

[-o <Output excel file>] : (Output)Path to the output graph file(.xlsx) (optional)(By default saves as Graphs.xlsx in base directory)

[-b <Benchmark file>] : (Input) Benchmark file (optional)

[-trend] : (Input) Enables trend analysis (optional)

[-image <Path>] : (Output) Saves graphs as JPEG images at specified path (optional)(By default images are saved in base directory)

[-mht <Output mht file>] : (Output) Saves graphs as web page(mht) at specified path (optional) (By default saves as Graphs.mht in base directory)

[-ct <Chart Type>] : Chart type to be plotted (optional)

- 1-Smooth curve with markers
- 2-Smooth curve

[-font] : Font Type (optional)

- 1-Small font type
- 2-Medium font type
- 3-Large font type

[-XlLaunch] : Opens the output excel file(optional)

Note: If path for the input file(s) not specified, the base directory will be treated as the default path.

Examples:

```
>GraphToolConsole -c ConfigFile.xml -t perf_winsock2 -i perf.LOG -image  
>GraphToolConsole -c ConfigFile.xml -o graph.xlsx -i perf.LOG -t perf_winsock2  
>GraphToolConsole -c ConfigFile.xml -o graph.xlsx -t perf_winsock2 -i perf.LOG  
>GraphToolConsole -c ConfigFile.xml -i perf.LOG -t perf_winsock2 -image -ct 2 -  
XlLaunch  
  
>GraphToolCo -t perf_winsock2 -c ConfigFile.xml -i perf.LOG -image c:\images\ -ct  
2  
  
>GraphTool -c ConfigFile.xml -i perf.LOG -mht graphs.mht -t perf_winsock2 -ct 2 -  
font 1 -XlLaunch
```

Running the Winsock Performance Test

This section guides you through running the Winsock Performance Test. The Graph Tool can graph the results of this test. For more information, see [Using the Graph Tool](#). The following steps guide you through the process of running the Winsock Performance Test.

- [Executing the Winsock Performance Test Command in Platform Builder](#)
- [Creating a Winsock Performance Test Template](#)
- [Creating a Winsock Performance Test Pass](#)
- [Running the Winsock Performance Test](#)

Executing the Winsock Performance Test Command in Platform Builder

A device must be running and attached to Platform Builder before you can connect to the device from the Windows Embedded Compact Test Kit (CTK). For more information, see [Connecting to a Device](#).

When running the Winsock Performance Test, you must execute a command in Platform Builder to run the server part of the Winsock Performance Test before starting the Winsock Performance Test in the Windows Embedded CTK.

Before executing the command to run the server part of the Winsock Performance Test, Platform Builder must be attached to the device that you are going to connect to from the CTK to run the Winsock Performance Test.

To execute the Winsock Performance Test Command in Platform Builder

1. In Platform Builder, On the **Target** menu, click **Target Control**.
2. At the **Window CE>** command prompt, type `s perf_winsockd2 -debug` and press **Enter**.

You are now ready to run the Winsock Performance Test in the Windows Embedded CTK.

Creating a Winsock Performance Test Template

Before creating a test pass you must create a template to base the test pass on. For more information, see [Creating a Test Pass Template](#).

To create a Winsock Performance test pass template

1. In the Windows Embedded Compact Test Kit, on the **File** menu, point to **New**, and then click **Test Pass Template**.
2. On the **Test Manager** window, on the **Test Pass Templates** tab on the right, under **Edit Templates**, click the **New** button.
3. In the new template text box that appears, type **Winsock Performance** to rename the template from **New Template**.
4. In the **Test Case Explorer** tree view display area, double-click **Windows Embedded Test Catalog** to expand the catalog tree.
5. Double-click the **Networking** folder.
6. Double-click **Ethernet**.

7. Click **Winsock Performance Test**, and then click the **Add >** button to add the selected test to the new **Winsock Performance** template.
8. Click **Done** to exit **Test Manager**.

You are now ready to create a test pass from the template you just created and saved.

Creating a Winsock Performance Test Pass

Once you have created the test pass template, you can create a test pass.

You must first be connected to a device before you can create a test pass. Connect to the device that you have attached to Platform Builder running the **Window CE** `Window CE>s perf_winsockd2 -debug` command as described in [Executing the Winsock Performance Test Command in Platform Builder](#), if you have not already done so.

To create a Winsock Performance test pass from the template

1. On the **View** menu, click **Test Pass Templates View**.
2. In the **Test Case Explorer** window, on the **Test Pass Templates View** tab, right-click the **Winsock Performance** template, and then click **Use as Current Test Pass**.

Note If **Use as Current Test Pass** is not available on the pop-up menu, you most likely are not connected to a device.

3. In the **Test Pass** display window, click the new test pass **Windows Embedded** that is highlighted and type **Winsock Performance Pass** to rename it.

Note If a test pass was already in the **Test Pass** display window, you'll be asked if you want to save the current test pass before a new test pass is created. Running the Winsock Performance Test

After the Winsock Performance Test has been added to a test pass, you can run the test.

To run the Winsock Performance test

1. In the **Test pass display** window, click the plus sign (+) preceding **Winsock Performance Pass** to expand the test pass tree.
2. Click on **Winsock Performance Test**.
3. If the **Properties** window is not visible, on the **View** menu, click **Properties Window**.
4. Select the **Current Command Line** text box.
5. Change **server_ip** to **localhost** and **ip_version** to **4** in the command `"-o -d perf_winsock2 -c"-s server_ip -i ip_version -l true"`.
6. Right-click on **Winsock Performance Test** and click **Run Test**.
7. In the **Test pass display** window, right-click on **Winsock Performance Test** and click **Start Test**.
8. When the **Winsock Performance Test [In Progress]** changes to **Winsock Performance Test [Passed]**, the test run is complete and you can view the results.

For information about reviewing the test results, see [Using the Graph Tool](#).

The green status bar at the bottom of the Windows Embedded CTK main window displays the progress of the test.

Conclusion

The Windows Embedded Compact Test Kit (CTK) is a new application that provides significant improvements to the user interface and the overall feature set when compared to the previous release known as Windows Embedded CE Test Kit (CETK).

The Windows Embedded CTK is a tool that you can use to test the functionality and performance of device drivers and related hardware for a Windows Embedded Compact device. The Windows Embedded CTK has the following key features:

- Improved user interface that resembles Visual Studio
- Ability to group user-selected test cases into test passes
- Improved test results viewer lets you store and review test results per test pass
- Upon connection to a device, detection of peripherals and drivers needed for tests
- Integration with the new Graph Tool to graph performance test results
- Users can add custom Tux test harness (TUX)-based tests to the CTK
- Support for x86, MIPSII, MIPSII_FP, ARMv5, ARMv6, and ARMv7 processors

The Windows Embedded Test Catalog is the master list of all of the test cases in the Windows Embedded CTK, which currently includes the following tests:

- Accelerometer
- Audio
- Backlight
- Battery
- Bootloader
- Cellular
- Communication Bus
- Display
- File System
- Input
- Kernel
- Multimedia
- Networking
- NLED (Notification light-emitting diode)
- OAL (OEM Abstraction Layer)
- Power Manager
- Security
- Shell
- Smart Card
- Storage Media

- USB

Additional Resources

- [Windows Embedded website](http://go.microsoft.com/fwlink/?LinkId=183524) (http://go.microsoft.com/fwlink/?LinkId=183524)

Copyright

This document is provided “as-is.” Information and views expressed in this document, including URL and other Internet Web site references, may change without notice. You bear the risk of using it.

This document does not provide you with any legal rights to any intellectual property in any Microsoft product. You may copy and use this document for your internal, reference purposes.

© 2011 Microsoft. All rights reserved.