Please note that Cypress is an Infineon Technologies Company.
The document following this cover page is marked as “Cypress” document as this is the company that originally developed the product. Please note that Infineon will continue to offer the product to new and existing customers as part of the Infineon product portfolio.

Continuity of document content
The fact that Infineon offers the following product as part of the Infineon product portfolio does not lead to any changes to this document. Future revisions will occur when appropriate, and any changes will be set out on the document history page.

Continuity of ordering part numbers
Infineon continues to support existing part numbers. Please continue to use the ordering part numbers listed in the datasheet for ordering.

THIS SPEC IS OBSOLETE

Spec No: 001-52970

Spec Title: WINDOWS HARDWARE QUALITY LABS (WHQL) SIGNING PROCEDURE FOR CUSTOMER MODIFIED CYPRESS USB DRIVER FILES - AN52970

Sunset Owner: HASIB MANNIL (HBM)

Replaced by: NONE

AN52970

Windows Hardware Quality Labs (WHQL) Signing Procedure for Customer Modified Cypress USB Driver Files

Author: Anand Srinivasan
Associated Project: No
Associated Part Family: None
Software Version: DTM Studio 1.6.8367.000
Related Application Notes: None

If you have a question, or need help with this application note, contact the author at aasi@cypress.com

Abstract

AN52970 shows you how to obtain a digital signature by passing Microsoft's Windows® Hardware Quality Labs (WHQL) testing for customer-modified Cypress USB driver files (*CyUSB.sys* and *CyUSB.inf*). Cypress supplies a digitally signed driver with its reference designs and development kits, whereas that signature is broken when you add customer-specific information (VID, PID, strings, and so on) to the driver files.

Contents

Introduction

The 'Certified for Windows' logo is the digital signature used by Microsoft® to indicate that third-party hardware and drivers are compatible with Microsoft operating systems. This signature is obtained by running and passing a set of Microsoft-defined tests to ensure the quality of the hardware and driver. Driver Test Manager (DTM) is the platform that runs the Microsoft Windows Driver Kit (WDK) test framework with driver and hardware device for the Windows Logo Program. Windows 7 or Vista 64-bit operating systems do not allow running unsigned driver in normal mode. (64-bit Windows 7 or Vista is run in test mode while running tests on the drivers). This digital signature can be obtained using a third-party CA certificate or from Microsoft using the process mentioned in this application note.

CyUSB.sys is WHQL-certifiable (we run tests to make sure the driver can be certified). This application note introduces DTM setup used for the Windows Logo Program and the procedure to run these tests for customer-modified Cypress USB driver files (*CyUSB.sys* and *CyUSB.inf*). We recommend that customers first check the Microsoft website for any updates in tests or procedures and for any information beyond the scope of this application note. We also recommend the frequently asked questions (FAQ on WHQL Signing for Cypress USB Driver files) section of this application note.

Logo Programs

Microsoft defines these logo programs:

- Device logo (Device must pass tests that verify reliability and performance of the device).
- Driver quality signature (Driver must pass tests that verify the criteria and design requirements defined by Microsoft for drivers that run on Windows).
- System logo (Devices and drivers in the system must have their respective device logos or driver quality signatures and the system must pass an additional series of tests).

Device and system logos allow the manufacturer to provide a logo that specifies the device and system is Windows certified. Driver quality signature allows the driver to load in Windows with compatibility and reliability requirements defined by Microsoft.

The application note illustrates the procedure using as example, DTM test performed for Cypress FX2LP™ development board using Cypress driver files (*CyUSB.sys and CyUSB.inf)* on a XP 32-bit system.

DTM Setup

The DTM setup consists of the following,

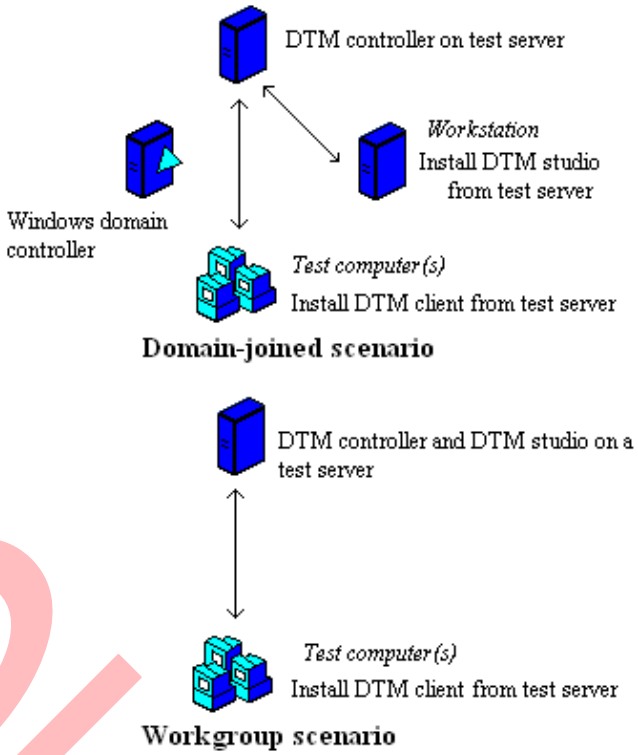
- DTM clients: Computers on which DTM tests are run.
- DTM controller: Controls and schedules the tests on DTM clients. It communicates with DTM client through Windows Test Technology (WTT).
- DTM studio provides the visual interface through which the controller and clients are managed.

DTM controller is installed on the controller computer, which is the test server. The installation of DTM studio is based on the network scenario in which the DTM setup is deployed.

There are two scenarios in which DTM setup is deployed: Domain-joined scenario and Workgroup scenario.

In domain-joined scenario, the networked computers have a definite client server model. In a workgroup scenario, the computers are all peers. In this scenario, DTM studio must be installed on the test server on which DTM controller is installed. In domain-joined scenario, it can be installed on a separate workstation.

Figure 1. Network Scenario of DTM Setup



The main difference between both scenarios is that the domain-joined scenario requires a Windows domain controller. This means one additional computer. The Windows domain controller must have Microsoft Active Directory configured and running. Microsoft Active Directory is used to store information and settings in a central database. Deployment of domain-joined scenario increases the security and ease of setting changes through the use of Windows domain controller. This comes at the cost of using one more computer (Windows domain controller) in the DTM setup.

The DTM setup has a few constraints. They are:

- DTM controller must be installed on a test server running Windows server 2003.
- DTM controller is not supported on a computer that is already set up as a domain controller. User account with administrator permissions is required to install DTM controller.
- DTM controller, DTM studio, and DTM client are not supported on Virtual PC environment.
- DTM client on which the tests are scheduled should be running the OS for which the device and driver is being tested.

- DTM controller is connected to the network with atleast100 Mbps network connection. Tests with time constraints may fail if the client does not respond on time.
- A unique name with 15 or fewer characters is used to name DTM clients.
- Client running Windows 2000 can be used for testing purposes only. There is no logo program for Windows 2000.
- The 64-bit version of Windows Vista and 7 will not allow unsigned drivers to be used in normal mode. If the testing involves unsigned drivers, it must be test-signed and tested with the OS in test-mode.

For more information on the constraints and setting up of DTM setup, see the step-by-step guide available at http://msdn.microsoft.com/en-us/library/windows/hardware/br259125.aspx. It also provides details of a sample test.

Recommended Practices

The following are best practices can be followed for the DTM tests.

- Install the OS anew on the clients before running the tests. Previous tests may have made changes to the OS, which can affect the current test results.
- Update all software components of clients and controllers before starting the test.
- The controller requires a user account to run tests on clients. Some tests require administrator rights. Create user accounts in the clients with administrator rights for the controller.
- The controller reboots the clients for certain tests. Disable the password of the controller's user account in the client. This reduces the requirement of user intervention during tests.
- During tests, the username and password of clients are sent unencrypted on the network by the controller for log files. Therefore, make sure the DTM setup is on a private network without connection to other networks or internet.
- Connect the USB device (device under test) to the host through a self-powered USB 2.0 certified compliant hub. The device should be in tier 1 of the hierarchy. Tests such as the device framework test fails if the USB device is directly connected to the host.
- Right click My Computer and then click Properties. In the Advanced tab, click the Settings button under Startup and Recovery. This opens the Startup and Recovery window. In this window set the following parameters.

Table 1. Parameters for Start and Recovery

Parameter	Value
Default OS	OS for which device and driver is tested
Write debugging information	Kernel memory dump

- In the Desktop, right click and select Properties. In the Screensaver tab:
  - Set screensaver to none.
  - Click **Power** button; in the window that appears set never to turn-off monitor, turn-off hard disks, system standby, and system hibernates.
- The controller communicates with the client often and tests may fail if communication fails. Turn-off Windows firewall.
- For machine names, file names and other credentials use words that describe their use. Though some names may be long, it aids in debugging.

The status display of DTM monitor panes is static. Refresh the panes to get the latest status.

Test-signing the Driver

The 64-bit version of Windows Vista and 7 will not allow unsigned drivers to be used in normal mode. If the testing involves unsigned drivers, it must be test-signed and tested with the OS in test-mode. Tools to test-sign the driver files are available as part of Windows Driver development kit (WinDDK). Following are steps involved in test signing the driver and

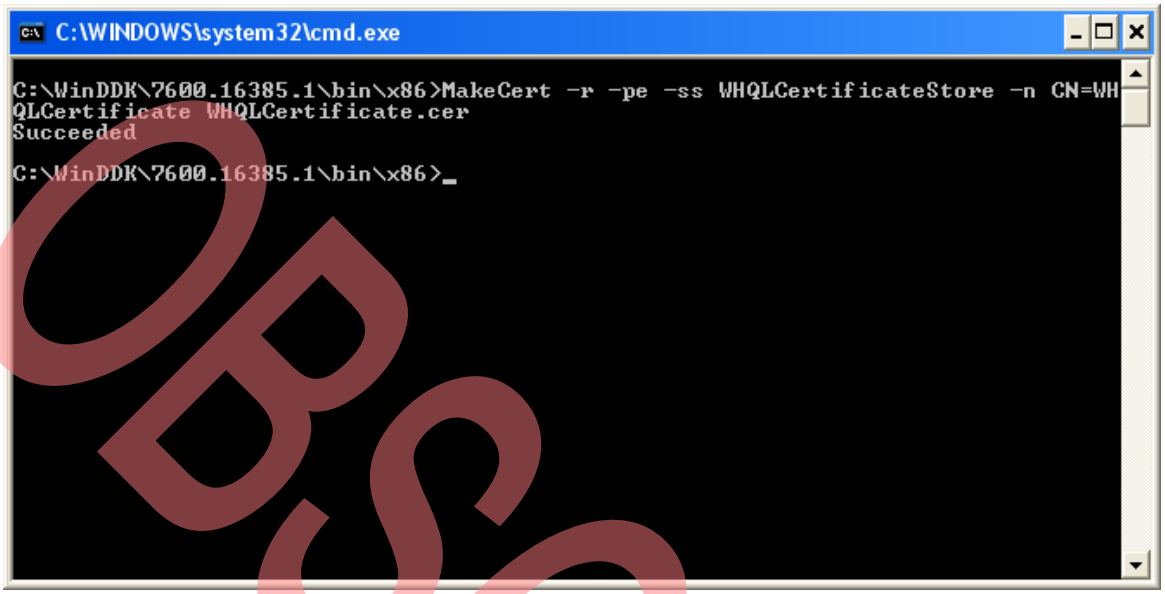
Creating certificate

1. MakeCert.exe available as part of Windows SDK is used to create certificate.
2. Navigate to the path of the MakeCert.exe and use the following command as shown in Figure 2 to create the certificate

   MakeCert -r -pe -ss CertificateStore -n CN=CertificateName outputCertificateFile.cer

   Here **CertificateStore** is the certificate store that stores the certificate (a custom name), **CertificateName** is the certificate name and **outputCertificateFile** is the name of the output certificate file where the test X.509 certificate will be written.

Figure 2. Creating Certificate



3. More information on MakeCert can be found in the Microsoft webpage on MakeCert.

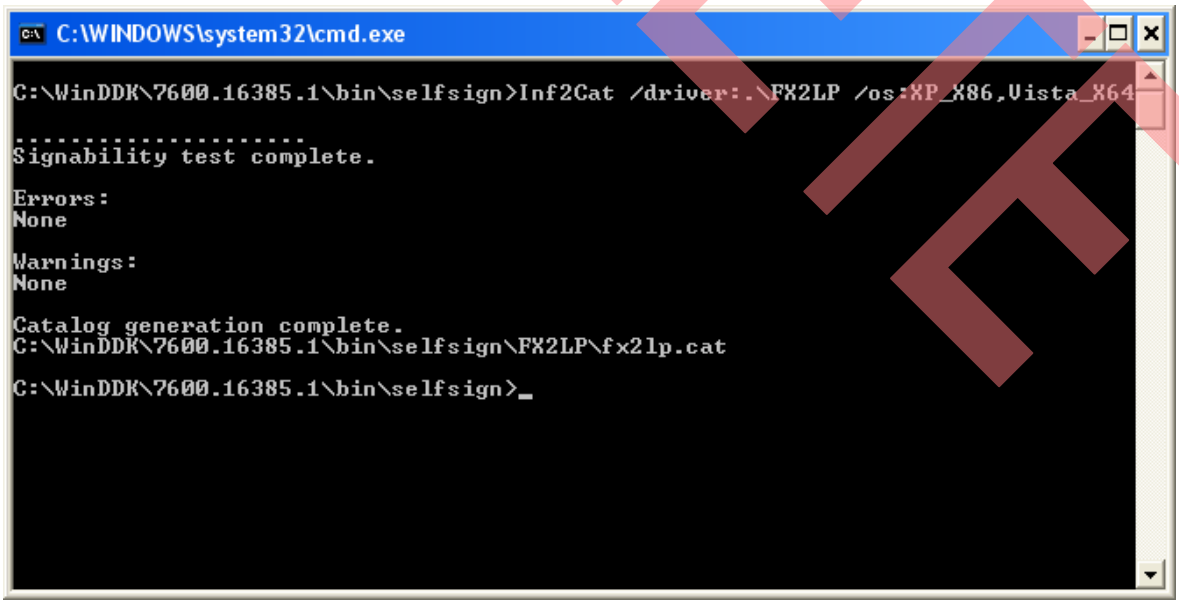
Generating Catalog File

1. To create the catalog file (**.cat*) Inf2Cat.exe is used.
2. Inf2Cat.exe takes input from the inf file and creates the catalog file based on input from the inf file hence it is easier to use.
3. Navigate to the path of Inf2Cat.exe and use the following command as shown in Figure 3.

   Inf2Cat /driver:**Packagepath** /os:**WindowsVersionsList**

   Here **Packagepath** is the path of the driver files and **WindowsVersionsList** is the list of Windows OS and CPU platform to be signed for separated by, (Comma).

Figure 3. Creating Catalog File



4. Inf2Cat takes the name of the catalogfile from the CatalogFile name specified in the inf file. More information on Inf2Cat can be found in Microsoft webpage on Inf2Cat.

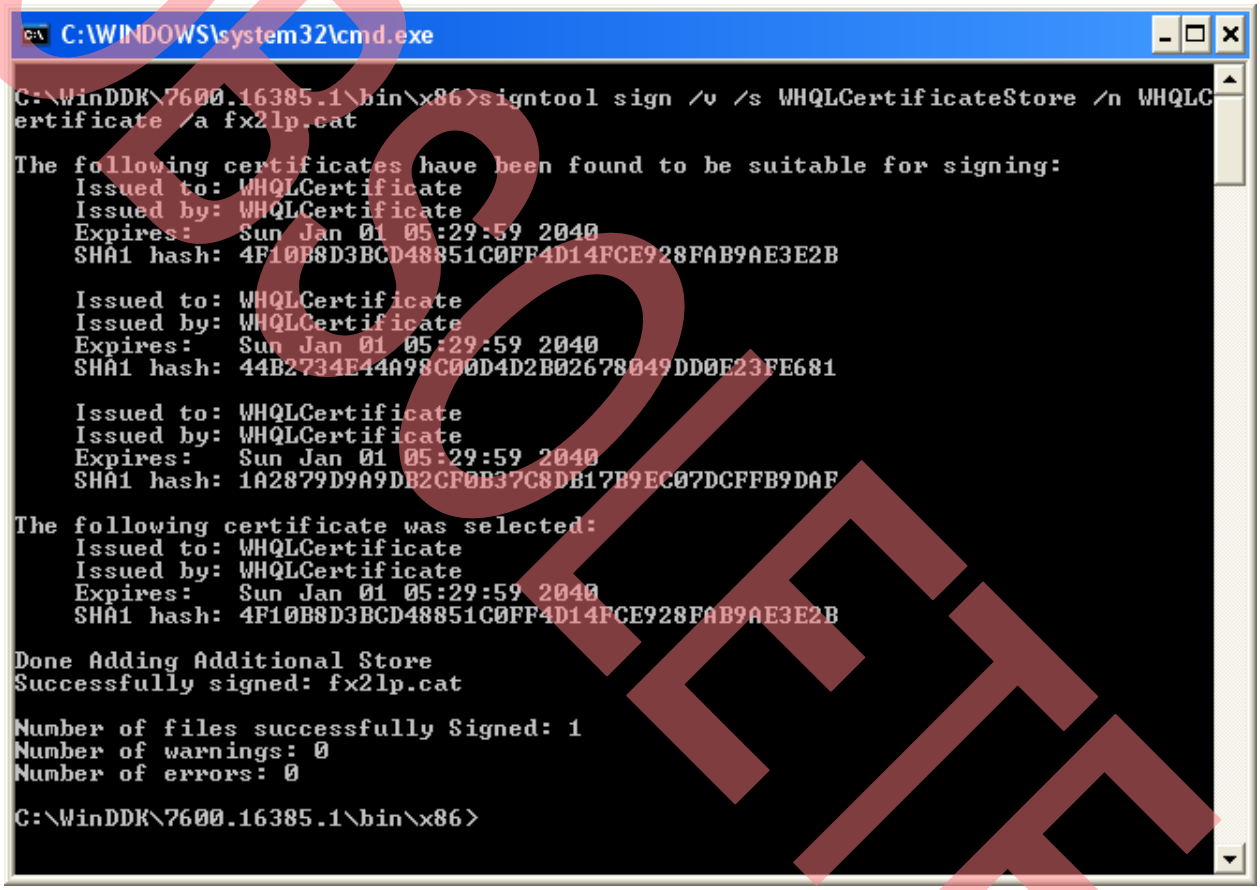
Test-signing the Catalog File

1. SignTool is used to sign the catalog file.
2. Navigate to the path of SignTool.exe and use the following command as shown in Figure 4

   SignTool sign /v /s **CertificateStore** /n **CertificateName** /a **CatalogFileName.cat**

   Here **CatalogFileName** is the name of the catalogfile.

Figure 4. Test-signing the catalog file



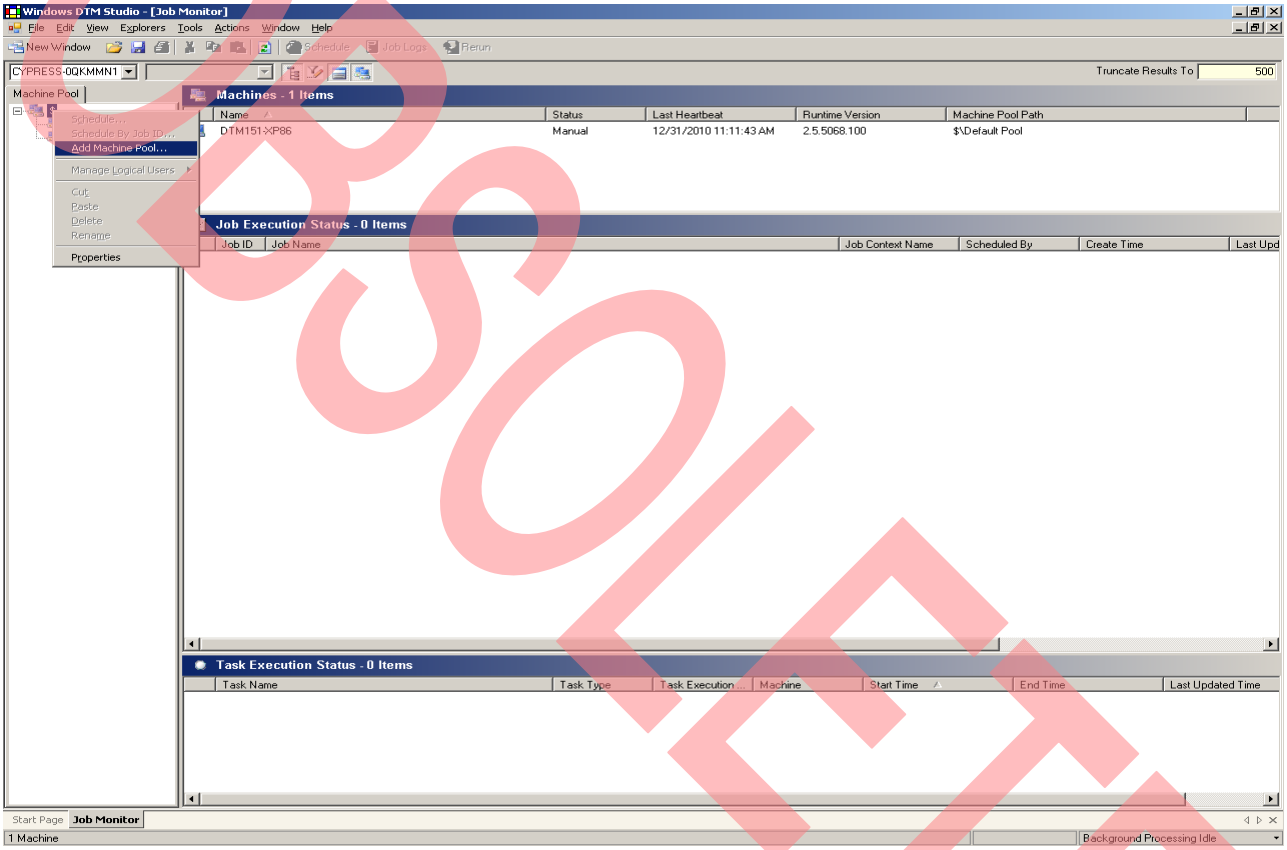
3. More details about the tool and the command-line arguments can be found in the signtool webpage. More information on test-signing the driver can be found in Test-Signing a Driver File webpage. Information on how to boot the 64-bit OS in test-mode can be found in TESTSIGNING boot configuration option webpage.

DTM Procedure

The procedure for DTM testing is as follows:

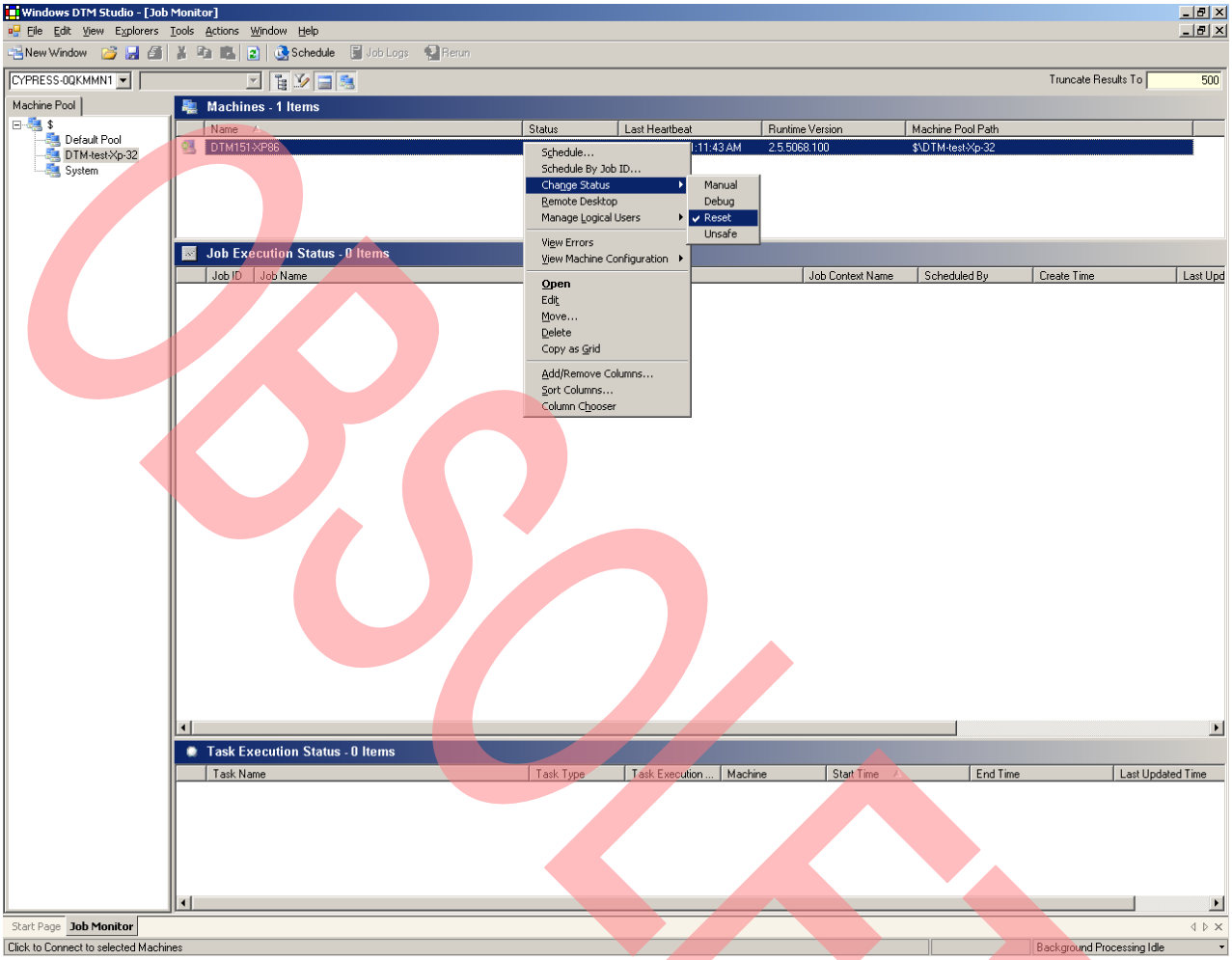
1. Configure the DTM setup network. Install DTM controller on the controllers. Install DTM studio on the controller or a separate computer based on the network scenario. Run the setup of DTM studio from the controller.
2. Install the OS anew on the clients. Install DTM client on the clients by running the setup from the controller.
3. In DTM studio, click **Job Monitor** in the Explorer menu to open the job monitor.
4. In the Machine Pool tab, right click the **$** symbol, and then click **Add Machine Pool**.

Figure 5. Adding Machine Pool in DTM Studio



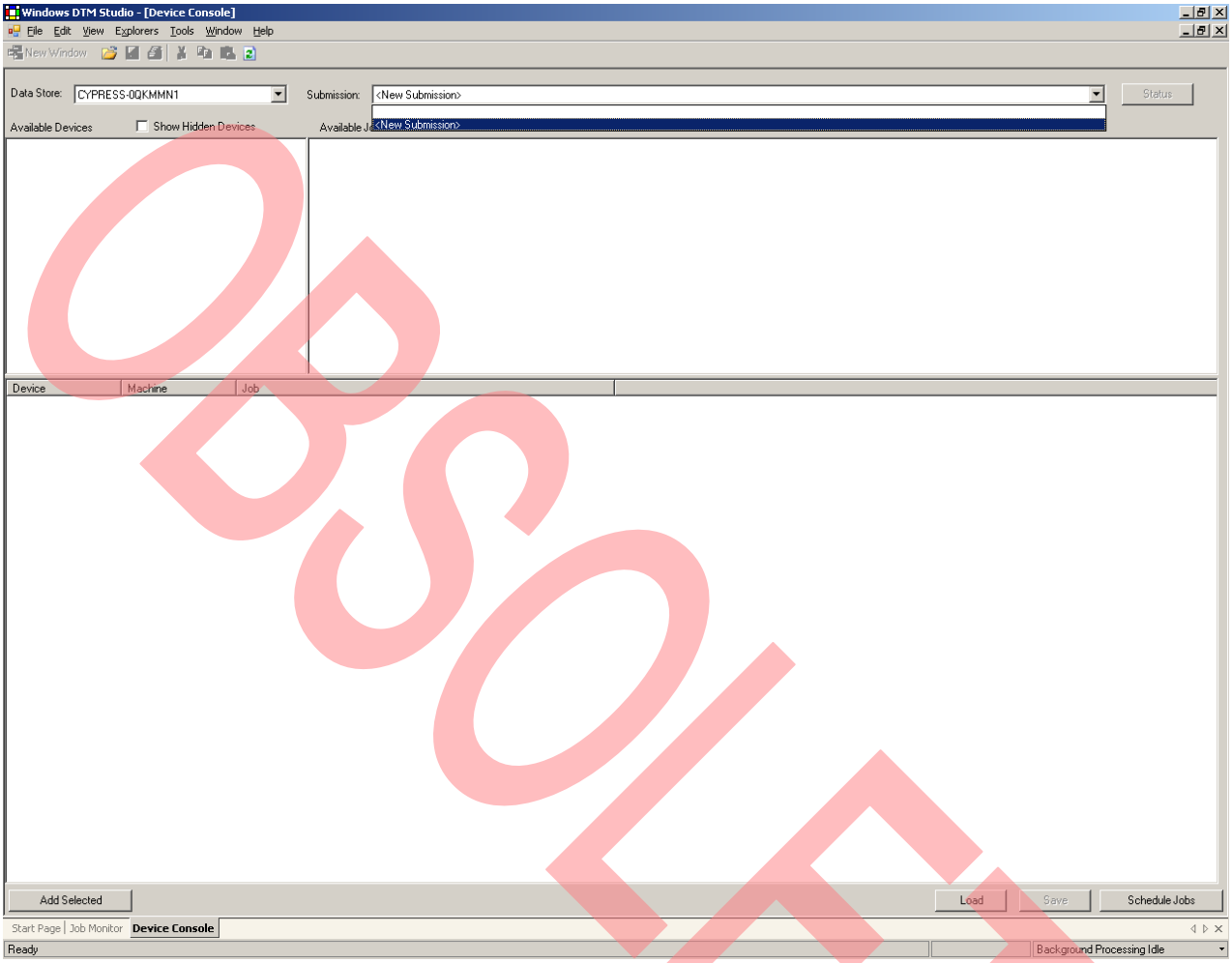
5. Select the controller as the job delivery agent for the machine pool.
6. Move clients on which tests are to be run to the machine pool. The status of the clients is manual.
7. Right click the client; in Change Status, click **Reset**. After sometime, the status changes to Ready. This must be the client status to run the tests.

Figure 6. Resetting the Client



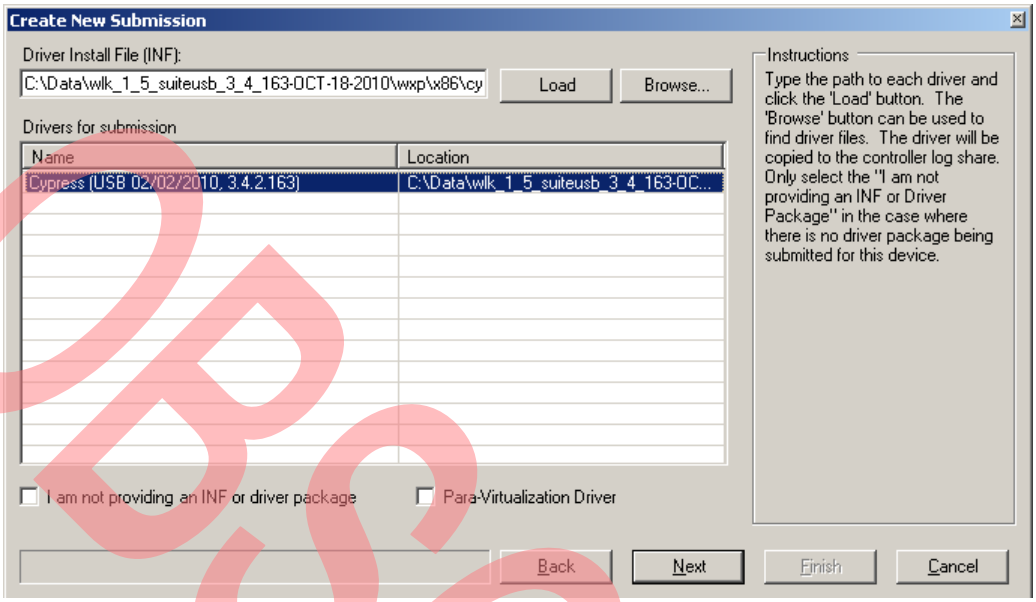
8. In the device console window, select **New Submission** in the submission field.

Figure 7. New Submission



9. The ‘Select OS’ dialog box is displayed. Select the OS type for which the driver and device is to be signed and click **Next**.
10. In the ‘Select Category’ dialog box, select the logo program for which the submission is done, and click **Next**.
11. In the ‘Logo Program’ dialog box, select the test suites and click **Next**.
12. In the ‘Qualification Level’ dialog box, select the qualification level and click **Next**.
13. In ‘Create New Submission’ window, enter the name of the Submission Package. Then select the machine pool that will be used to run the tests.
14. In Driver Install File (*.inf*) dialog box, select the appropriate *.inf* files to load the drivers required for testing and click **Next**. For more details on inf file used with *CyUSB.sys*, see AN61465 - Working With inf File of a Device Using CyUSB.sys.

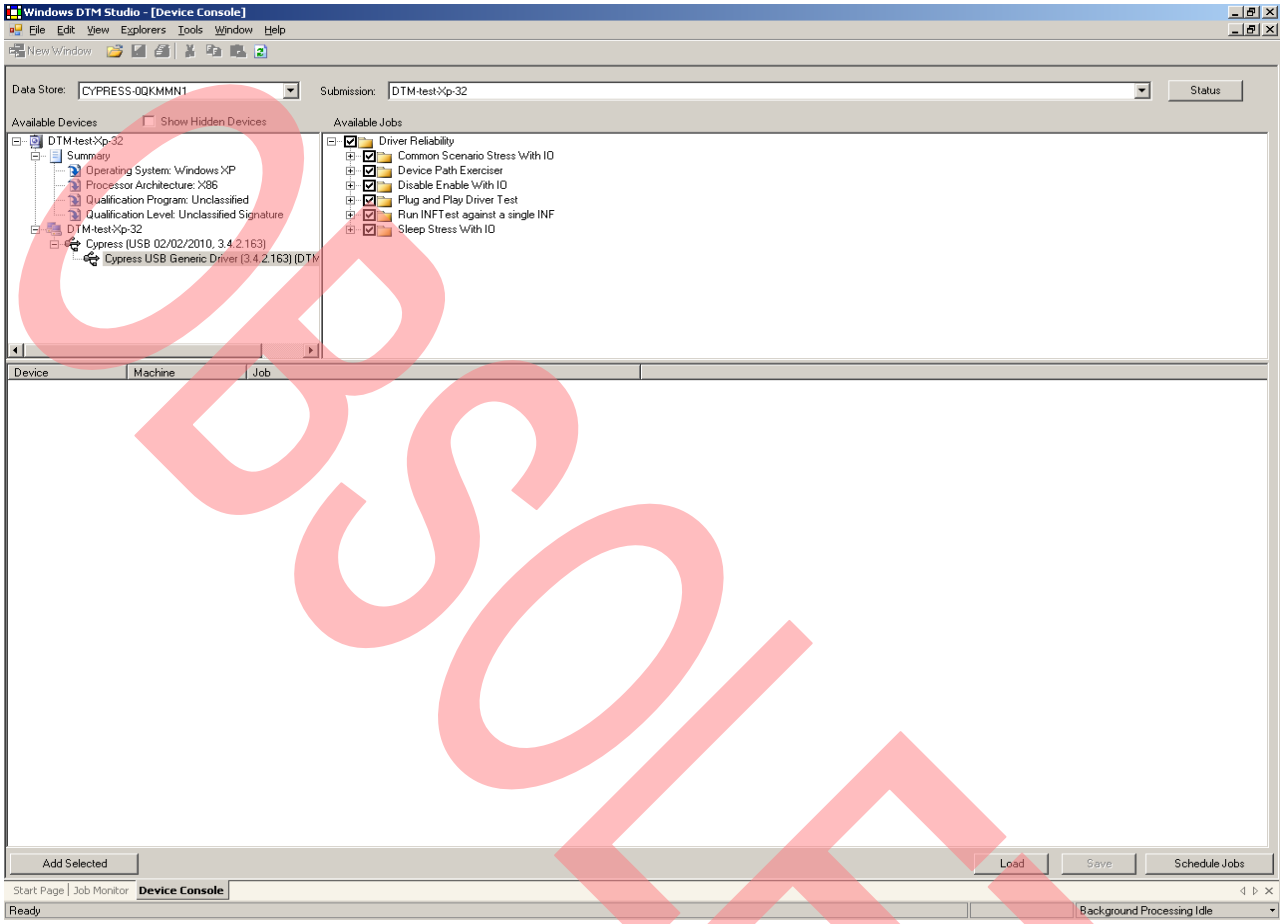
Figure 8. Selecting the *.inf* File



15. In the Device List dialog box, select the device to be tested from the tree of devices displayed.
16. The details of the test are displayed to be checked for correctness. Click **Finish**; the available jobs are displayed in the device console.
17. In **Connectivity_USB** test right click **USB-IF Test Certification ID Check** test and click **Edit Parameters…**. In the window that appears enter the USB-IF Test ID against the **TESTID** field. Enter the path of the Chapter test log against the **PATH_TO_LOGS** field. These test logs can be generated by running USB Chapter 9 tests using **USB Command Verifier** tool available at usb.org. Enter whether the device passes USB Interoperability compliance test against INTEROP_TESTS_PASSED field (TRUE for pass and FALSE for fail) and press OK.

    Note: Device should have passed the above mentioned tests for this test to succeed.
18. For certification, run all the tests. Check all the available jobs and click **Add Selected**. Now click **Schedule Jobs**.

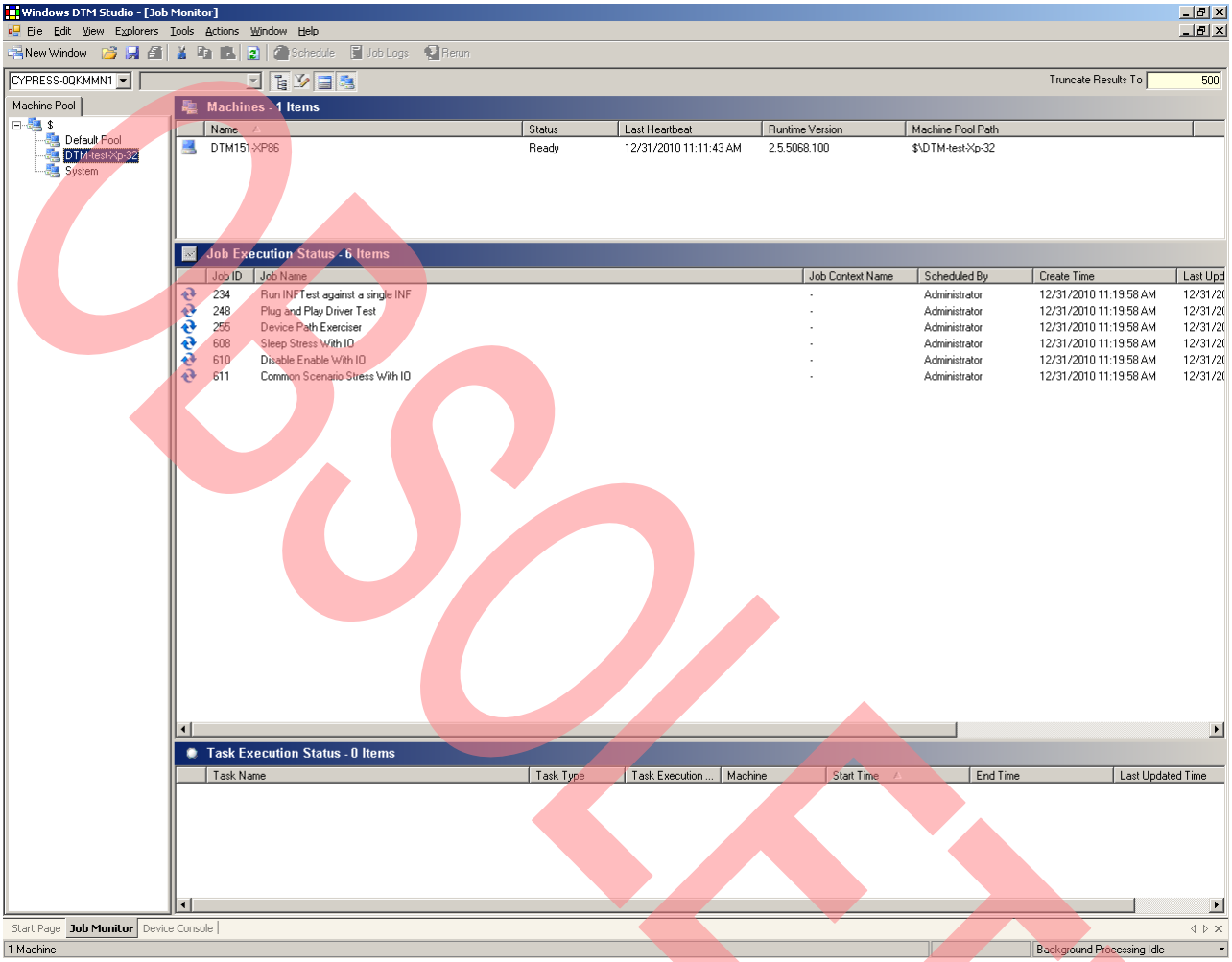
Figure 9. Selecting Tests



19. The job is scheduled on the clients by the controller. The controller usually reboots the client before starting the test.

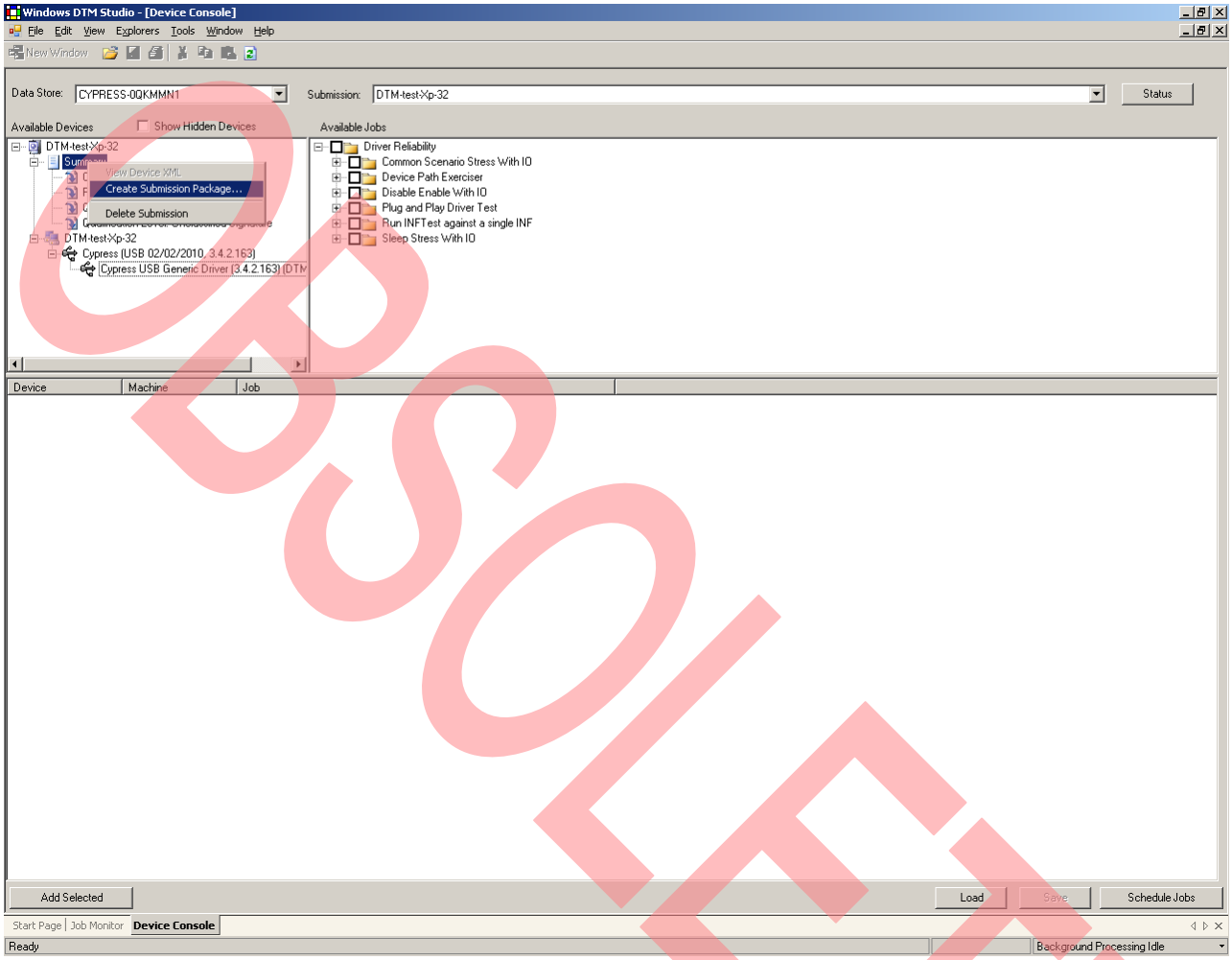
20. The status of the jobs, tasks, and clients are monitored using the Job Execution Status, Task Execution Status, and Machines panes respectively.

Figure 10. Status of Client Machine, Jobs, and Tasks in DTM Studio



21. The job status is known from the symbol near it. Each job consists of tasks. There are certain tasks which roll out the result. A failure of one of these tasks fails the job.
22. The WDK File Signature Verification task checks for the presence of digital signature in the files used during the test. This test always fails because the device and driver under test is not signed. It is an example of a task which does not roll out the result.
23. After all the jobs are complete, click **Status** in the device console. This starts Microsoft contingency and errata filtering to convert Microsoft accepted test case failures to pass. Now the submission status is displayed. All the tasks which roll out the result must be passed for the device and driver to be eligible for logo program.
24. In the Available Devices pane, right click the submission name and select **Create Submission Package**.
25. In the Save As dialog box, select the file location and name, and click **OK**. A *.cpk* file is generated. This is the result log of the DTM test.

Figure 11. Creating Submission Package



Submission for Logo Program

A Winqual account with necessary permissions is required to do the submission. More details on creating the Winqual account, permissions, billings, and constraints are found in https://sysdev.microsoft.com/ (**Note:** Open the link using Internet Explorer).

The steps to submit a logo program are as follows:

1. In the home page of the user account click **Hardware Logo** > **Create Logo** in the Windows Logo Programs section on the left.
2. In the right side of the page, the links for the setup of Winqual submission tool (WST) and Winqual uploader tool (WUT) are displayed. WST is used to create the submission package. WUT is a client application required for logo submission package uploads in winqual.
3. Install WST and WUT in the system.
4. Open WST and using the Add button, add the result log (*.cpk* file) of the DTM test.
5. If the driver used is not inbox (windows generic driver), then the driver, locales, and symbols (optional) must be added. Symbol files must be provided when making an "unclassified submission".
6. Save the list for later use. This is saved as an *.xml* file containing all the submission information.
7. Click the **Create Package** button to create the submission package.
8. In the home page of the user account click **Hardware Logo** > **Create Logo** in the Windows Logo Programs section of the left navigation.

9. Use the page that appears to upload the submission file (*.xml* file).
10. Provide the credentials requested and go through the concerned legal agreements. The credentials requested and legal agreements may vary based on the submission. The credentials are needed to get access to the system.
11. Use the next page that appears to upload the submission package.
12. Some credentials are requested at this stage to confirm the identity. Enter these to complete the submission process.

Common Issues and Resolution

- DTM setup not being up to date may cause tests to fail. Installing the OS anew and keeping the DTM installations up to date resolves these issues.
- Hardware interfaces not being up to specification may cause tests to fail. Make sure that the USB port, hub and the device are USB 2.0 compliant to avoid these issues.

FAQ on WHQL Signing for Cypress USB Driver files

1. **Question:** I get the following error while binding my device to *CyUSB.sys* in Windows 7/Vista 64-bit environment, "Windows encountered a problem installing the driver software for your device". What does this error mean? How can it be resolved?

   Answer: *CyUSB.sys* downloaded through our website is an unsigned driver. This error reported while an unsigned driver used in 64-bit operating systems in normal mode.
   Following are the steps to disable driver signature enforcement in 64-bit operating system:
   a) During boot-up press F8.
   b) In the list of options that appear select "Disable driver signature enforcement".
   This should resolve the issue.

2. **Question:** Usage of *CyUSB.sys* in Vista 64-bit operating system gives Code 39 error (Code 52 in the case of Windows 7). What does this error mean?

   Answer: *CyUSB.sys* downloaded through our website is an unsigned driver. This error reported while an unsigned driver used in 64-bit operating systems in normal mode.
   Following are the steps to disable driver signature enforcement in 64-bit operating system:
   a) During boot-up press F8.
   b) In the list of options that appear select "Disable driver signature enforcement".
   This should resolve the issue.
   Note: In the case of Windows Vista 64-bit operating system the error message is "Windows cannot load the device driver for this hardware. The driver may be corrupted or missing. (Code 39)".
   In the case of Windows 7 64-bit operating system it is "Windows cannot verify the digital signature for the drivers required for this device. A recent hardware or software change might have installed a file that is signed incorrectly or damaged, or that might be malicious software from an unknown source. (Code 52)".

3. **Question:** I need more information on WHQL procedure including cost details, debug procedure and so on.

   Answer: More details on Windows Logo Program are available in Microsoft website.

Summary

This document introduces DTM setup and describes the procedure to obtain the 'Certified for Windows' logo for customer-modified Cypress USB driver files (*CyUSB.sys* and *CyUSB.inf*). It discusses the best practices that help to create the submission package for Windows logo program. Microsoft provides the logo after reviewing the submission package.

About the Author

Name:	Anand Srinivasan
Title:	Applications Engineer Sr
Contact:	aasi@cypress.com

Document History

Document Title: Windows Hardware Quality Labs (WHQL) Signing Procedure for Customer Modified Cypress USB Driver Files – AN52970

Document Number: 001-52970

Revision	ECN	Orig. of Change	Submission Date	Description of Change
**	2695279	AASI	04/20/2009	New application note.
*A	3129959	AASI	01/06/2011	Added section on test-signing the driver Uploaded better resolution pictures.
*B	3148701	AASI	01/20/2011	Changed document title from 'Windows Hardware Quality Lab (WHQL) Signing Procedure' to 'Windows Hardware Quality Labs (WHQL) Signing Procedure for CyUSB.sys'. Moved the 'Recommended Practices' section to the start of the document to maintain the operation flow.
*C	3261401	AASI	05/19/2011	Changed title to "Windows Hardware Quality Lab (WHQL) Signing Procedure for Cypress Semiconductor USB Product Driver (*CyUSB.sys)*". Added FAQ section based on common issues reported by customers.
*D	3309783	AASI	07/12/2011	Removed step 8 since it is not valid in DTM studio 1.6.8367.000 Modified step 9 to 14 to add more clarity Added step 17 to describe the USB-IF Test Certification ID Check test
*E	3560013	AASI	03/24/2012	Minor text edits. Added content on test-signing driver. Updated template.
*F	3645889	AASI	06/14/2012	Updated Introduction section and added reference to AN61465 in step 14.
*G	4356725	HBM	04/22/2014	Obsolete document.

Worldwide Sales and Design Support

Cypress maintains a worldwide network of offices, solution centers, manufacturer's representatives, and distributors. To find the office closest to you, visit us at Cypress Locations.

Products

Automotive	cypress.com/go/automotive
Clocks & Buffers	cypress.com/go/clocks
Interface	cypress.com/go/interface
Lighting & Power Control	cypress.com/go/powerpsoc cypress.com/go/plc
Memory	cypress.com/go/memory
Optical Navigation Sensors	cypress.com/go/ons
PSoC	cypress.com/go/psoc
Touch Sensing	cypress.com/go/touch
USB Controllers	cypress.com/go/usb
Wireless/RF	cypress.com/go/wireless

PSoC® Solutions

psoc.cypress.com/solutions

PSoC 1 | PSoC 3 | PSoC 5

Cypress Developer Community

Community | Forums | Blogs | Video | Training

Technical Support

cypress.com/go/support

Cypress Semiconductor
198 Champion Court
San Jose, CA 95134-1709

Phone : 408-943-2600
Fax : 408-943-4730
Website : www.cypress.com