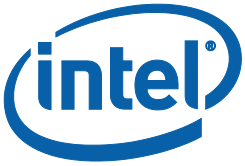


EHCI Removal from Braswell SoC

Technical White Paper

July 2014

Revision 0.5

Intel Confidential

Document: IBP# 549688

By using this document, in addition to any agreements you have with Intel, you accept the terms set forth below.

You may not use or facilitate the use of this document in connection with any infringement or other legal analysis concerning Intel products described herein. You agree to grant Intel a non-exclusive, royalty-free license to any patent claim thereafter drafted which includes subject matter disclosed herein.

INFORMATION IN THIS DOCUMENT IS PROVIDED IN CONNECTION WITH INTEL PRODUCTS. NO LICENSE, EXPRESS OR IMPLIED, BY ESTOPPEL OR OTHERWISE, TO ANY INTELLECTUAL PROPERTY RIGHTS IS GRANTED BY THIS DOCUMENT. EXCEPT AS PROVIDED IN INTEL'S TERMS AND CONDITIONS OF SALE FOR SUCH PRODUCTS, INTEL ASSUMES NO LIABILITY WHATSOEVER AND INTEL DISCLAIMS ANY EXPRESS OR IMPLIED WARRANTY, RELATING TO SALE AND/OR USE OF INTEL PRODUCTS INCLUDING LIABILITY OR WARRANTIES RELATING TO FITNESS FOR A PARTICULAR PURPOSE, MERCHANTABILITY, OR INFRINGEMENT OF ANY PATENT, COPYRIGHT OR OTHER INTELLECTUAL PROPERTY RIGHT.

A "Mission Critical Application" is any application in which failure of the Intel Product could result, directly or indirectly, in personal injury or death. SHOULD YOU PURCHASE OR USE INTEL'S PRODUCTS FOR ANY SUCH MISSION CRITICAL APPLICATION, YOU SHALL INDEMNIFY AND HOLD INTEL AND ITS SUBSIDIARIES, SUBCONTRACTORS AND AFFILIATES, AND THE DIRECTORS, OFFICERS, AND EMPLOYEES OF EACH, HARMLESS AGAINST ALL CLAIMS COSTS, DAMAGES, AND EXPENSES AND REASONABLE ATTORNEYS' FEES ARISING OUT OF, DIRECTLY OR INDIRECTLY, ANY CLAIM OF PRODUCT LIABILITY, PERSONAL INJURY, OR DEATH ARISING IN ANY WAY OUT OF SUCH MISSION CRITICAL APPLICATION, WHETHER OR NOT INTEL OR ITS SUBCONTRACTOR WAS NEGLIGENT IN THE DESIGN, MANUFACTURE, OR WARNING OF THE INTEL PRODUCT OR ANY OF ITS PARTS.

Intel may make changes to specifications and product descriptions at any time, without notice. Designers must not rely on the absence or characteristics of any features or instructions marked "reserved" or "undefined". Intel reserves these for future definition and shall have no responsibility whatsoever for conflicts or incompatibilities arising from future changes to them. The information here is subject to change without notice. Do not finalize a design with this information.

The products described in this document may contain design defects or errors known as errata which may cause the product to deviate from published specifications. Current characterized errata are available on request.

Contact your local Intel sales office or your distributor to obtain the latest specifications and before placing your product order.

Copies of documents which have an order number and are referenced in this document, or other Intel literature, may be obtained by calling 1-800-548-4725, or go to: http://www.intel.com/design/literature.htm%20

All products, computer systems, dates, and figures specified are preliminary based on current expectations, and are subject to change without notice.

This document contains information on products in the design phase of development. Do not finalize a design with this information. Revised information will be published when the product is available. Verify with your local sales office that you have the latest datasheet before finalizing a design.

Intel processor numbers are not a measure of performance. Processor numbers differentiate features within each processor family, not across different processor families. See www.intel.com/products/processor_number for details.

Code names featured are used internally within Intel to identify products that are in development and not yet publicly announced for release. Customers, licensees and other third parties are not authorized by Intel to use code names in advertising, promotion or marketing of any product or services and any such use of Intel's internal code names is at the sole risk of the user.

Intel and the Intel logo are trademarks of Intel Corporation in the U.S. and/or other countries.

*Other names and brands may be claimed as the property of others.

Contents

Revision History

Document Number	Revision Number	Description	Revision Date
549688	0.5	Initial release.	July 2014

§ §

1 Introduction

1.1 Planned SoC Changes

For Braswell SoC, Intel plans to remove both EHCI controllers along with their integrated rate matching hubs (RMH). Intel will continue to provide USB functionality through the xHCI controller for USB 2.0 and USB 3.0 connectivity.

1.2 Document Scope

This document discusses the implications of removing EHCI from the Braswell and discusses specific solutions for OS installation from an USB storage device with an OS that does not natively support xHCI in both the OEM and Channel Environments.

1.3 Overview

In prior generations, systems had both xHCI and EHCI controllers. In the case where a new OS install from a USB device was required the hardware would have the legacy EHCI drivers natively available in the OS. This native EHCI support allows the USB ports to function during a clean OS install thereby providing an end-user the ability to use a USB based storage or HID device to manually install the xHCI driver provided by Intel.

With the removal of the EHCI Controller, Intel does not anticipate any issues with only exposing an xHCI controller for Windows* 8/8.1 or Linux* (2.6.38 or later kernel) due to native xHCI support on the OS. However, Windows 7 does not natively support xHCI controller and relies on manual installation of the Intel's xHCI driver. While there are two types of USB pre-OS support for storage and HID (Human Interface Devices) devices, the concern with the lack of in-the-box support for the xHCI controller involves cases where the Windows 7 operating system needs to be installed from a USB storage device.

In addition to OEM considerations, the channel provides some unique challenges as end-users can purchase motherboards and build their own systems. As a general case, SATA based optical interface, OS and OS recovery capabilities are not included with the motherboard.

1.4 USB HID Device Support

Intel does not foresee any issue concerning USB HID support with an OS lacking native xHCI support. This is because HID class USB devices are emulated by the BIOS as PS/2 devices behind a legacy super/IO or Embedded Controller using the I/O 60/64h I/O addresses. SMI interrupts are used to provide the emulation of PS/2 devices from the BIOS. This PS/2 emulation is not affected by the removal of the EHCI hardware.

It should be noted that the emulation of the keyboards and mice are generally limited to 'compatible' functionality, meaning that vendor-specific key mappings, hotkeys, and other functionality beyond the standard PS/2 keyboard and mouse likely are not supported. The intent of providing HID class emulation for at least keyboard and mouse even in OS run-time is primarily to enable the end-user to navigate through the policies to install an xHCI driver, where the standard OS drivers will then provide the full USB experience. While HID emulation support is not a major concern it may be prudent for Intel and OEMs to work together to ensure main keyboards and mice have baseline functional support.

1.5 USB Storage Device Support

For the purpose of this document, USB Storage class devices are categorized into thumb drives and optical drives and represent the largest obstacle to the seamless removal of the legacy EHCI USB controllers. Windows 7 installations with xHCI hardware concerns become visible during OS installation and installation of the xHCI driver into a fresh OS.

1.6 xHCI-based Windows* 7 Install Flow

The high-level flow of Windows* 7 installations on a 'new' system from USB-attached storage consist of the following:

1. BIOS Discovery of Bootable Images - The bootable image is on a USB interface attached to the Intel xHCI controller.
2. BIOS Control Transferred to the Boot loader - The boot-loader uses an INT-13 DOS style handler (Legacy BIOS) or the Block IO Interface (UEFI BIOS) to access the USB device storage subsystem. During this phase, a virtual RAM drive is created and all fundamental OS information, including hardware drivers and WinPE kernel, are loaded into this virtual drive.
3. WinPE Kernel Starts - The USB specific issues start here as the xHCI driver is not available in the WinPE kernel. The BIOS is continuing to emulate storage services so that the boot loader is unaware that the physical storage device relies on the USB controller stack. When all of the drivers are loaded into the RAM drive, execution is handed off to the WinPE kernel. At this point in time, the WinPE kernel ceases to utilize the BIOS routines, and instead uses self-contained services within the RAM drive. The WinPE kernel will not be able to use the xHCI controller hardware as there is no xHCI driver in the RAM drive and access to the OS source files is severed. At this point, the install process is stuck, and a WinPE kernel message similar to the following will be displayed: "A required CD/DVD device driver is missing. If you have a driver floppy disk, CD, DVD, or USB flash drive, please insert now."
4. In addition to the primary issue of OS installation is the secondary issue of installing the xHCI driver, as the xHCI driver is not part of the Windows* 7 installation package. Therefore, the xHCI drivers will not be copied onto the OS target drive, either HDD or SDD, etc. If the driver can be acquired via a network (wired or wireless), or some other non-USB interface, then the problem is resolved, and the xHCI controller will now be visible. However, if the platform is dependent on getting the xHCI driver from the OS install source path (USB-attached ODD or Key) then there is an issue.

§§

2 xHCI USB Operating System Install Solutions

The following solutions are proposed for proper OS installation on xHCI only based systems:

1. Use Windows* 8/8.1 and not Windows 7 - Windows 8/8.1 is a natively xHCI aware OS so the pre-install and post-install phases of the OS will have drivers for the xHCI hardware.
    a. Note that this option does not presume any OEM specific customization of the OS installation media. Windows 8/8.1 will allow installation from a device and any post-install OEM driver bundling would function as intended.

5. Use OEM provided DVD, special HD partition (E.g. Prebuilt Linux with native xHCI support); Network, or SATA-attached optical drives for OS installation. This option bypasses the issue, and allows the OS image to be installed from an interface that is natively supported by Windows 7. Once the baseline OS image is installed, the OS will be able to load the xHCI driver, where thereafter full USB functionality will be supported.
    b. Prior to the xHCI driver installation, only keyboard and mouse HID class devices would be supported via PS/2 and Block IO Interface BIOS emulation.
    c. The xHCI driver would not be installable from USB-attached storage media
    d. A script based or GUI-based sequence could be used to format the destination USB media, copy the xHCI driver and Windows* 7 OS to the targeted drive for direct install – However this has not been proven
    e. Guidance for creating a Windows 7 Installation with xHCI Driver is provided in Adding USB 3.0 Drivers to Win 7 Install
2. Install Windows 7 in virtual machine on physical disk and configure it to be bootable directly from physical machine as described in Install Win 7 Virtual Machine as Bootable
3. Create or provide tools or instructions to create a custom Windows 7 WinPE image with the xHCI driver included – This is described in Boot From Customer WinPE with USB 3.0 Driver

2.1 Adding USB 3.0 Drivers to Windows* 7 Install

1. Make ISO image from Windows 7 DVD
2. Save install.wim and boot.wim to root of C drive
   E.g. c:\wim
3. Download Windows Automated Installation Kit for your specific OS version
4. Start AIK Deployment Tools Command Prompt from Start menu

5. Use the Command prompt to move to install.wim
   cd c:\wim
6. Create a directory under c:\wim called mount
   c:\wim\mount
7. Mount the WIM image with write permissions:
   E.g. imagex /mountrw install.wim 5 c:\wim\mount
   Imagex will tell which number is which Windows* edition if it's not found.
   Number 5 is in this case for Windows 7 Ultimate Edition 32 bit.
8. Add USB3.0 drivers to the install.wim image using DISM
   a. Navigate to the location of the USB3.0 is located
      E.g. C:\USB3\x86
   b. Add the USB3.0 Drivers (.inf) file to the install.wim image
   c. E.g. C:\USB3\x86>dism /image:C:\wim\mount /add-driver /driver:. /recurse

      *Note: If DISM says that files are essential for booting but drivers are unsigned, then add /forceunsigned parameter.

      C:\USB3\x86>dism /image:C:\wim\mount /add-driver /driver:. /recurse */forceunsigned*
9. Save the WIM image with new drivers by unmounting and committing

   E.g. C:\USB3\x86>imagex /unmount /commit C:\wim\mount
10. Add USB3.0 driver to two installations of boot.wim with the following steps, so the drivers will be accessible during installation.
    a. Add driver to boot.wim 1
       i. Mount the boot.wim 1 image with write permissions
          E.g. C:\wim>imagex /mountrw boot.wim 1 c:\wim\mount
       ii. Change to the directory where the USB3.0 drivers are located.
          E.g. C:\wim>cd C:\USB3\x86
       iii. Add the USB3.0 Drivers (.inf) file to the base boot.wim 1 image
          E.g. C:\USB3\x86>dism /image:C:\wim\mount /add-driver /driver:. /recurse
       iv. Unmount the boot.wim 1 image
          E.g. C:\USB3\x86>imagex /unmount /commit C:\wim\mount
    b. Add driver to boot.wim 2
       i. Mount the boot.wim 2 image with write permissions
          E.g. C:\wim>imagex /mountrw boot.wim 1 c:\wim\mount
       ii. Change to the directory where the USB3.0 drivers are located.
          E.g. C:\wim>cd C:\USB3\x86

iii. Add the USB3.0 Drivers (.inf) file to the base boot.wim 2 image
E.g. C:\USB3\x86>dism /image:C:\wim\mount /add-driver /driver:. /recurse

iv. Unmount the boot.wim 2 image
E.g. C:\USB3\PPT\x86>imagex /unmount /commit C:\wim\mount

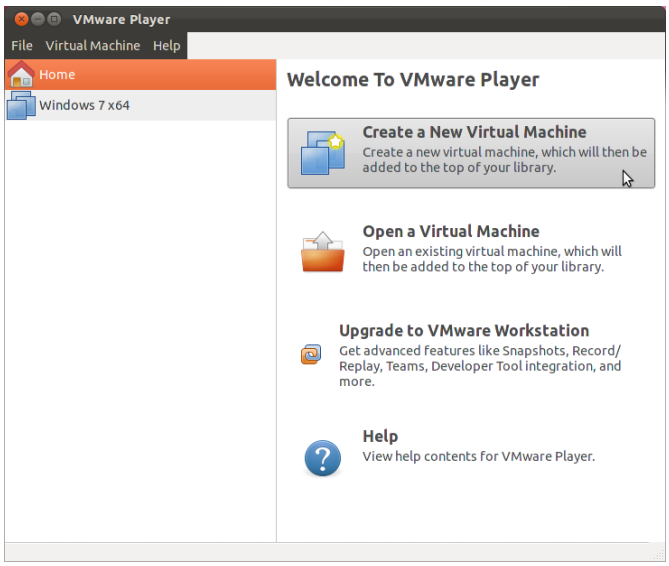
11. Add the install.wim and boot.wim back to the ISO image with your favorite ISO handling software (E.g. UltraISO)
12. Burn ISO as new DVD (Make sure that the DVD is bootable) with favorite burning software (E.g. CDBurnerXP) or create image to a UFD.
13. Install the Windows 7 as usual. After installation, Windows 7 will have the correct and working USB3.0 drivers.

References:

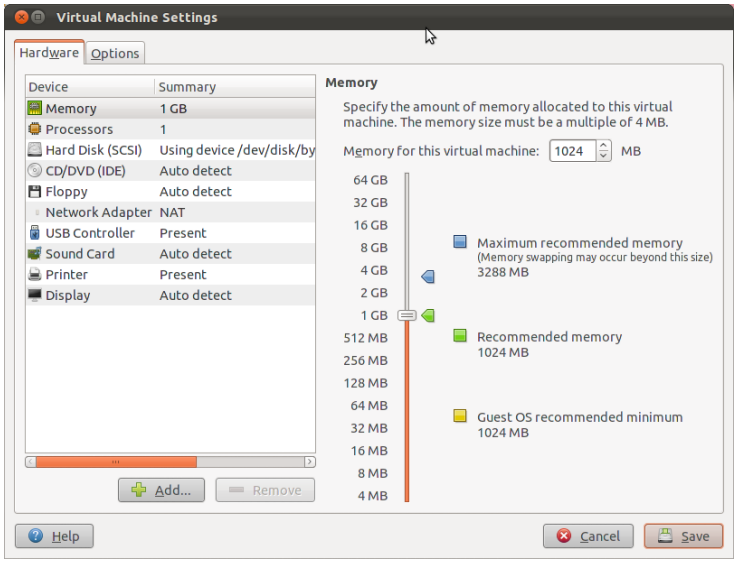
http://superuser.com/questions/63773/how-to-add-drivers-to-windows-7-installation-dvd

2.2 Install Windows* 7 Virtual Machine as Bootable

1. **Install OS other than Win* 7 (E.g. Ubuntu 11.10 - The Linux OS must support xHCI / or Windows* 8.1).**
2. Install VMware* Player 4 or newer and run with root privilege
This document will not cover how to install VMWare program.
    a. Install Win7 using VMware* Player – Use the Wizard to create a new Virtual Machine and select option to install the operating system later.

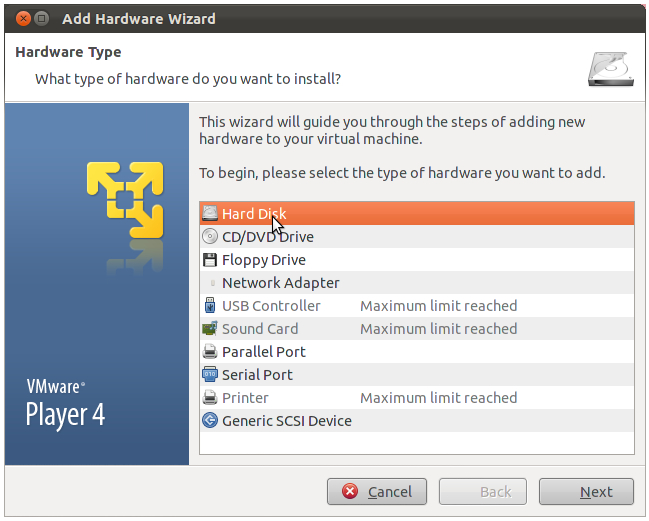


b. Highlight recent created “Windows 7” virtual Machine and click to “Virtual Machine Player Settings”

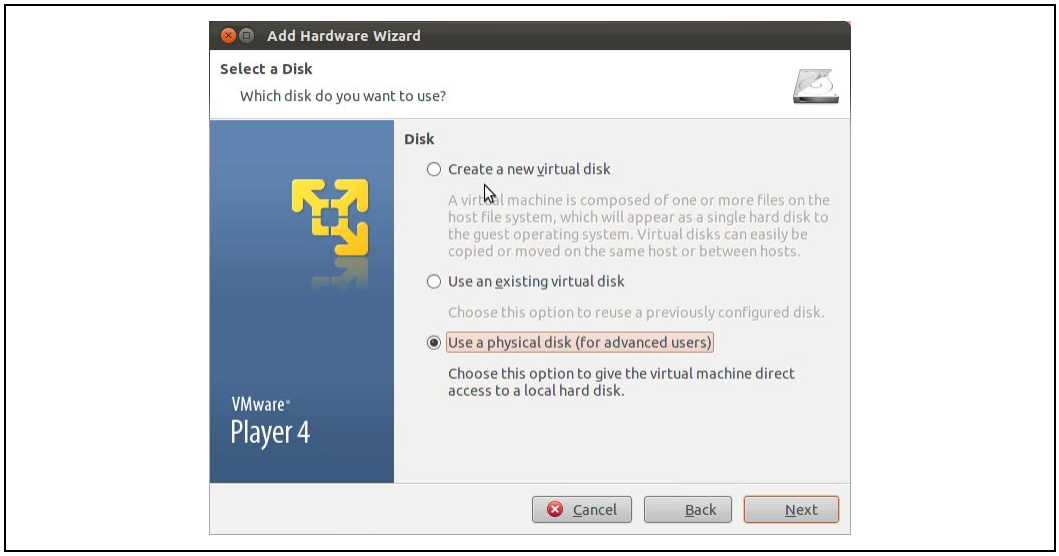


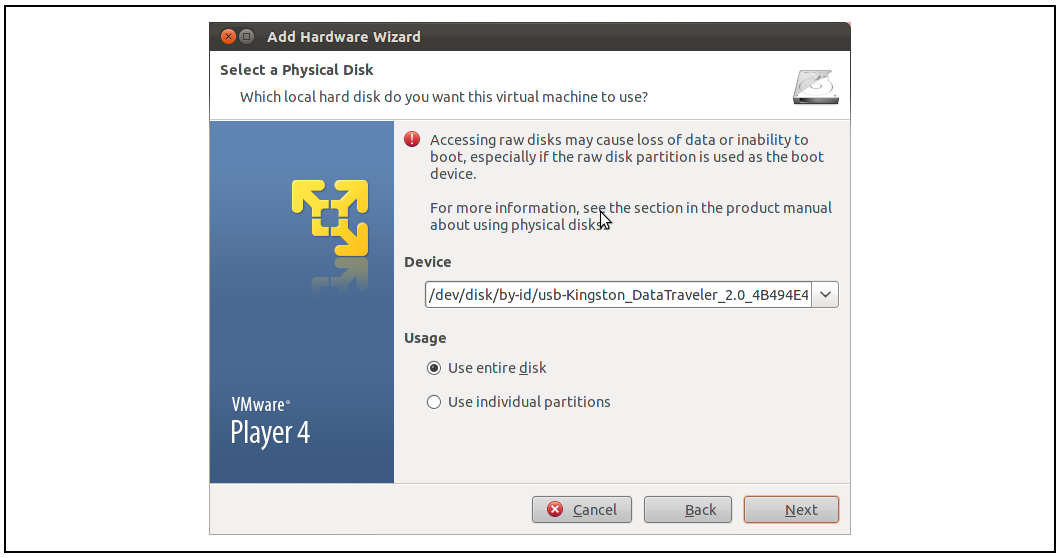
i. Click “Add”

ii. Select “Hard Disk”, then click “Next”



iii. Select “Use a physical disk (for advanced users)”





iv. Click “Next” the “Finish” to complete the setting

c. At “Virtual Machine Setting->CD/DVD”, select your standard Win7 installation ISO Image

d. Click “Play virtual machine” and install Win7 normally with the modified Win 7 image (see Adding USB 3.0 Drivers to Win 7 Install).

3. Preparing the HDD for boot

a. Apply Microsoft* Fix it 50470 or edit the registry manually.

i. Locate and then click one of the following registry sub keys:

- HKEY_LOCAL_MACHINE\System\CurrentControlSet\Services\Msahci

- HKEY_LOCAL_MACHINE\System\CurrentControlSet\Services\IastorV

ii. In the right pane, right-click Start in the Name column, and then click Modify.

iii. In the Value data box, type 0, and then click OK.

iv. Reference: http://support.microsoft.com/kb/922976

4. Shut down the VM and the host PC.
5. Restart and boot Win7 from the HDD.
6. Done

2.3 Boot from Customer WinPE with USB 3.0 Driver

1. Create a custom WinPE image
   a. Start AIK Deployment Tools Command Prompt from Start menu
   b. Copy the WinPE default to a working local directory path:

      copype.cmd <arch> <destination>

      Where <arch> can be x86, amd64, or ia64 and <destination> is a path to local directory. For example,

      E.g. copype.cmd x86 c:\winpe_x86

      The following directory structure will be created and all files copied
      \Winpe_x86
      \Winpe_x86\ISO
      \Winpe_x86\Mount

   c. Mount the WinPE image with write permissions
      E.g. c:\winpe_x86>imagex /mountrw winpe.wim 1 C:\winpe_x86\mount

   d. Change to the directory where the USB3.0 drivers are located.
      E.g. c:\winpe_x86>cd C:\USB3\x86

   e. Add the USB3.0 Drivers (.inf) file to the base WinPE image
      E.g. C:\USB3\x86>dism /image:C:\winpe_x86\mount /add-driver /driver:. /recurse

   f. Unmount the WinPE image
      C:\USB3\x86>imagex /unmount /commit C:\winpe_x86\mount

g. Copy the base image (Winpe.wim) into the \Winpe_x86\ISO\sources folder, and rename the file to Boot.wim
C:\USB3\x86>copy c:\winpe_x86\winpe.wim c:\winpe_x86\ISO\sources\boot.wim

2. Prepare the UFD
   a. Open command prompt, use Diskpart to format the UFD as FAT32
   E.g.
      i. Diskpart
      ii. select disk 2
      iii. clean
      iv. create partition primary
      v. select partition 1
      vi. active
      vii. format quick fs=fat32
      viii. assign
      ix. exit

   This example above assumes Disk 2 is the UFD

3. Copy customized WinPE image to the UFD using command prompt
   xcopy c:\winpe_x86\iso*.* /e f:\
   Where f is the letter of the UFD

4. Copy USB driver to the same UFD where the customized WinPE is located
5. Boot from customized WinPE with USB3 driver added.
6. Navigate to the Win7 installation disk.
   E.g. E:
7. Start Win7 installation (USB ODD, USB Key or anywhere).
   E.g. Setup.exe
8. Choose custom installation
9. Choose load driver to select the following driver to be installed:
   "Intel(R) USB3.0 eXtensible Host Controller"
   "Intel(R) USb3.0 Root Hub"
10. Continue the installation as usual.
11. Newly installed Win7 will have the USB driver installed.

§§