# NVIDIA JETSON TX1 USB2.0 TUNING GUIDE

DA-08619-001_v1.6 | March 2018

**Application Note**

# DOCUMENT CHANGE HISTORY

DA-08619-001_v1.6

| Version | Date | Description of Change |
|---|---|---|
| 1.0 | June 12, 2017 | Initial Release |
| 1.5 | July 24, 2017 | Added USB 2.0 Compliance Test tool |
| 1.6 | March 15, 2018 | •General editing throughout the application note<br>•Updated xUSB registers in Table 3<br>•Updated steps for USB test mode |

# TABLE OF CONTENTS

# LIST OF FIGURES



# LIST OF TABLES

# OVERVIEW

This application note describes the registers and steps needed to tune the USB2.0 high speed eye diagram for NVIDIA® Jetson™ TX1. USB-IF provides complete test specification and instructions on their website for high speed host and device mode testing. NVIDIA typically uses Tektronix oscilloscopes for USB characterization. The following test procedures are for Tektronix oscilloscopes.

- Host Mode Testing
  - Procedure for using Tektronix TDS694C: http://www.usb.org/developers/docs/Host_HS_Test.pdf
  - Procedure for using other Tektronix scopes: http://www.usb.org/developers/docs/Host_test_procedure.pdf
- Device Mode Testing
  - Procedure for using Tektronix TDS694C: http://www.usb.org/developers/compliance/Device_HS_Test.pdf
  - Procedure for using other Tektronix scopes: http://www.usb.org/developers/docs/DeviceTestProcedure.pdf

Customers are free to use oscilloscopes from other vendors to do USB characterization.

# REQUIRED EQUIPMENT

- Tektronix TDS694C or faster digital sampling oscilloscope
- Tektronix P6247 or P6248 or equivalent differential probe * 1
- High-speed USB Electrical Test Fixture
- Oscilloscope USB test Software
- Tool to access register/memory space in Tegra or build a special image to force USB Test mode enabled

# ATTACHMENTS

The following file is attached to this application note:

*USB 2.0 Compliance Test Tool.nvzip*

To access the attached file(s), click the **Attachment** icon on the left hand toolbar on this PDF (using Adobe Acrobat Reader or Adobe Acrobat). Select the file and use the Tool Bar options (**Open, Save**) to retrieve the document(s). Files with the .nvzip extension (if applicable) can be extracted using 7-Zip file archive software, or may be renamed to .zip and extracted with other archive software.

Figure 1 shows an example of how to obtain access to the attachment(s) of this PDF file.

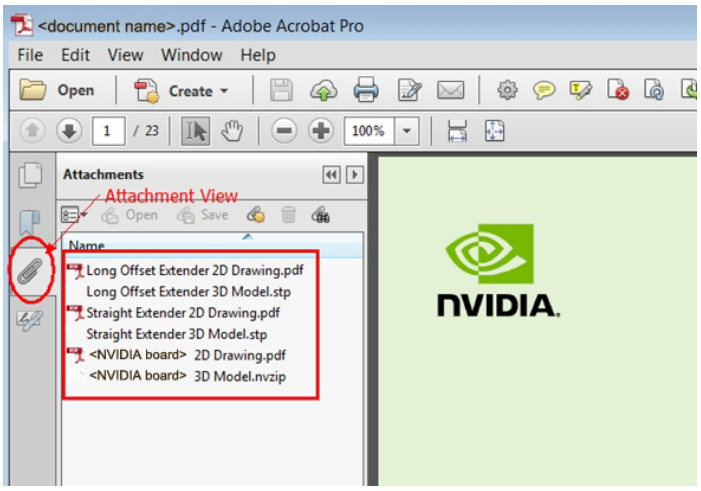


Figure 1. Access to PDF Attachments Example

# REGISTERS FOR HOST MODE TESTING

Figure 2 shows the USB2.0 PHY and HSIC multiplexing options.

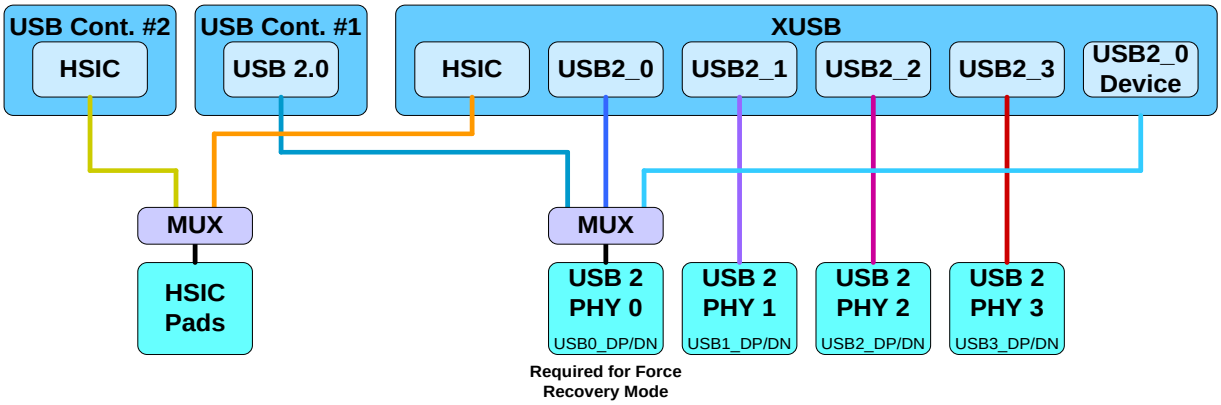


Figure 2. Tegra X1 Controllers and Interfaces Routing Map

**Note:** In R28.1 and later L4T Driver Package, USB0_DP/DN routing comes from xUSB Controller instead of USB2.0 Controller #1.

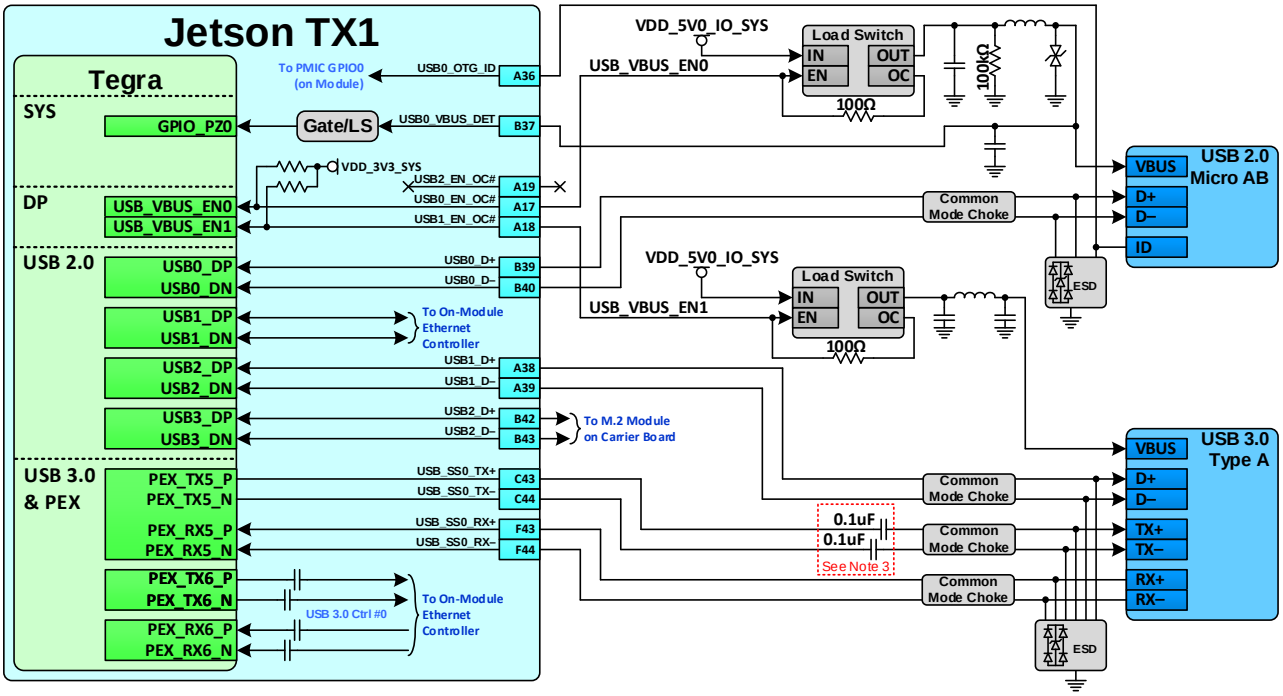


Figure 3. Jetson TX1 Connection Example

Toggle the Jetson TX1 USB registers listed in Table 1 to force Test J, Test K, Test SE0 NAK, and Test Packet on the USB2.0 port from USB2.0 Controller #1. For USB2.0 port from xUSB Controller, "tegra_hc_port_test_v3_static" tool can be used to enable test packets. The tool can be found attached to this document. Refer to section "Test Mode Programming Sequence" for additional information.

Table 1. USB2.0 Controller #1 Port Test Control Registers

| Descriptions | Register Name and Setting |
|---|---|
| Normal Operations (default) | USB2_CONTROLLER_USB2D_PORTSC1_0 [19:16] = 0000b |
| Test J | USB2_CONTROLLER_USB2D_PORTSC1_0 [19:16] = 0001b |
| Test K | USB2_CONTROLLER_USB2D_PORTSC1_0 [19:16] = 0010b |
| Test SE0 NAK | USB2_CONTROLLER_USB2D_PORTSC1_0 [19:16] = 0011b |
| Test Packet | USB2_CONTROLLER_USB2D_PORTSC1_0 [19:16] = 0100b |

**USB2.0 Controller #1 Port Registers Address:**

Port 0: 0x7D000174: USB2_CONTROLLER_USB2D_PORTSC1_0

# TEST MODE PROGRAMMING SEQUENCE

Changes have been made to the USB driver such that all USB2.0 ports are now from xUSB controller. Therefore, to test all USB2.0 ports based on R28.1 and later L4T Driver Package, proceed to the steps listed in the "Alternate Test Mode Programming Sequence" section.

The programming sequence for enabling USB2.0 test mode is as follows:

1. For USB2.0 port from USB2.0 Controller #1:
   a) Connect any USB device to the port (this will prevent the controller from entering power down mode).
   b) Disable the auto-suspend for the controllers:

      echo on > /sys/bus/usb/devices/usb1/power/control

      echo on > /sys/bus/usb/devices/usb2/power/control

      echo on > /sys/bus/usb/devices/usb3/power/control

   c) Set USB2_CONTROLLER_USB2D_PORTSC1_0[19:16] = 0100b to enable test packets.
   d) Unplug the USB device and plug in the test fixture to start USB2.0 eye diagram test.
2. For USB2.0 port from xUSB Controller:
   a) Connect any USB device to the port (this will prevent the controller from entering power down mode).
   b) Disable the auto-suspend for the controllers:

      echo on > /sys/bus/usb/devices/usb1/power/control

      echo on > /sys/bus/usb/devices/usb2/power/control

      echo on > /sys/bus/usb/devices/usb3/power/control

c) To enable test packets, use the "tegra_hc_port_test_v3_static" tool attached to this application note as shown:

```
sudo ./tegra_hc_port_test_v3_static -x -y
```

where y is the port number that a USB device is enumerated to and can be found by "cat /proc/kmsg". For USB1_D+/D- (module pin A38/A39), it will show "USB1-3: new H/F/L speed USB device…", here 1 is the bus number and 3 is the port number.

d) Unplug the USB device and plug in the test fixture to start USB2.0 eye diagram test.

**Note:** "Any USB device" listed in the sub-steps (a) refers to any real device, like HS uDISK or LS mouse.

# REGISTERS TO ADJUST HIGH SPEED USB2.0 EYE DIAGRAM

The following are Jetson TX1 USB registers that may be needed to tune the USB2.0 eye diagram. Section "Tuning Procedure" below discusses how to use these registers during characterization.

Table 2. USB2.0 Controller #1 Registers

| Register Name | Bit Fields | Description |
|---|---|---|
| **USB1_UTMIP_XCVR_CFG0_0: Address 0x7D000808** | | |
| UTMIP_XCVR_HSSLEW_MSB | 31:25 | HS slew rate control (MSB) |
| UTMIP_XCVR_SETUP_MSB | 24:22 | HS driver output setup control (MSB) |
| UTMIP_XCVR_HSSLEW | 5:4 | HS slew rate control (LSB) |
| UTMIP_XCVR_SETUP | 3:0 | HS driver output setup control (LSB) |
| Note: 0x0 is strongest, 0xF is weakest | | |
| **USB1_UTMIP_XCVR_CFG1_0: Address 0x7D000838** | | |
| UTMIP_XCVR_TERM_RANGE_ADJ | 21:18 | Range adjustment on terminations |
| **USB1_UTMIP_SPARE_CFG0_0: Address 0x7D000834** | | |
| FUSE_RPD_CTRL | 8 | 15K Host Pull-down Override<br>1 = fuse default<br>0 = override |
| FUSE_TERM_RANGE_ADJ_SEL | 4 | ATERM Override<br>1 = fuse default<br>0 = override |
| FUSE_SETUP_SEL | 3 | SETUP Override<br>1 = fuse default<br>0 = override |

Table 3. xUSB Registers

| Register Name | Bit Fields | Description |
|---|---|---|
| **XUSB_PADCTL_USB2_OTG_PADx_CTL_0_0: Address 0x7009F088 for USB0, 0x7009F108 for USB1, and 0x7009F148 for USB2** | | |
| HS_SLEW | 8:6 | HS slew rate control |
| HS_CURR_LEVEL | 5:0 | HS driver output setup control |
| **XUSB_PADCTL_USB2_OTG_PADx_CTL_1_0: Address 0x7009F08C for USB0, 0x7009F10C for USB1, and 0x7009F14C for USB2** | | |
| RPD_CTRL | 30:26 | RPD_CTRL (15K host pull down) |
| TERM_RANGE_ADJ | 6:3 | HS termination control |
| **XUSB_PADCTL_USB2_BIAS_PAD_CTL_0_0 (device RX testing): Address 0x7009F284** | | |
| HS_SQUELCH_LEVEL | 2:0 | HS SQUELCH control |

**Note:** The USBx is based on the module pin name, so USB0 is USB0_D+/D- (B39/B40), USB1 is USB1_D+/D- (A38/A39), and USB2 is USB2_D+/D1 (B42/B43).

# TUNING PROCEDURE

During manufacturing chip production, each NVIDIA Tegra device is calibrated and the fuses corresponding to USB drive strength (SETUP or HS_CURR_LEVEL), HS termination (ATERM or TERM_RANGE_ADJ), and 15K host pull down (RPD_CTRL) are burnt on each chip.

Before making any USB measurements on USB2.0 port from USB2.0 Controller #1 (that is, USB0_D+/D-), ensure that USB1_UTMIP_SPARE_CFG0_0[4:3] are set to 11b and USB1_UTMIP_SPARE_CFG0_0[8] is set to 1b. Doing so will load the values programmed by ATE for SETUP, ATERM and RPD_CTRL into the pad configuration inputs. Do note that when USB1_UTMIP_SPARE_CFG0_0[4:3] bits are set, register USB1_UTMIP_XCVR_CFG0_0 bits [24:22] and [3:0] does not reflect actual drive strengths and USB1_UTMIP_XCVR_CFG1_0 bit [21:18] does not reflect actual termination values.

To find out the default drive strength and HS termination value, read from FUSE_USB_CALIB fuse. USB_CALIB[5:0] represents the fused SETUP value for USB0 and USB_CALIB[10:7] represents the ATERM value for all the ports. USB_CALIB[22:17] is for USB1, and USB_CALIB[28:23] is for USB2. FUSE_USB_CALIB_EXT[4:0] represents the RPD_CTRL (host pull-down) for all ports.

During the characterization stage, manually adjusting the SETUP value should be enough to fulfill compliance requirements. To do this, first set USB1_UTMIP_SPARE_CFG0_0[3] = 0 and then write a new SETUP value in USB1_UTMIP_XCVR_CFG0_0 bits [24:22] and [3:0]. Only adjust the HS termination value as a last resort. To adjust the HS termination value, set USB1_UTMIP_SPARE_CFG0_0[4] = 0 and then write a new value in USB1_UTMIP_XCVR_CFG1_0 bits [21:18].

**Note:** NVIDIA does not recommend customers adjusting termination values. Do note that if the TERM_RANGE_ADJ needs to be adjusted, it may result in an impedance mismatch on the board and further attention may be needed.

To tune the eye diagram for USB2.0 ports from xUSB, adjusting the HS_CURR_LEVEL can be done by modifying the xUSB registers directly.

It must be emphasized that if any SETUP/HS_CURR_LEVEL modification is needed, it must be done as an offset to the default fused value in order to take into account silicon process differences. Never apply a global overwrite SETUP/HS_CURR_LEVEL value for all silicon. There is a mechanism provided in software to read fused USB drive strength and add an offset to it. Consult an NVIDIA CE for additional information. For USB2.0 Controller #1, if an offset is used, USB1_UTMIP_SPARE_CFG0_0[3] should be set to 0 for the override to take into effect.

To change high speed slew rate, write directly to HS_SLEW bits. There is no override bit for high speed slew. For USB2.0 Controller #1, both MSB and LSB HS slew are to control the rise/fall rate but LSB[5:4] adjust the pre-driver strength and it has least efficiency.

# SOFTWARE VERIFICATION

A functional check is recommended. Connect the DUT to USB hosts and devices to perform a check on functionality.

To check if software implements the tuned offset step properly, load new software with offset included into another DUT and check to ensure:

For USB2.0 Controller #1:

USB1_UTMIP_SPARE_CFG0_0[3] = 0

USB1_UTMIP_XCVR_CFG0_0 bits [24:22] & [3:0] = USB_CALIB + tuned offset steps

For xUSB:

HS_CURR_LEVEL = USB_CALIB + tuned offset steps

# ALTERNATE TEST MODE PROGRAMMING SEQUENCE

Toggle Jetson TX1 USB registers listed in Table 4 to force Test J, Test K, Test SE0 NAK, and test packet on USB2.0 ports from xUSB controller.

Table 4. xUSB USB2.0 Port Test Control Registers

| Descriptions | Register Name and Setting |
|---|---|
| Normal operations | XUSB_XHCI_OP_PORTPMSCHS_x_TM [31:28] =0000b |
| Test J | XUSB_XHCI_OP_PORTPMSCHS_x_TM [31:28] =0001b |
| Test K | XUSB_XHCI_OP_PORTPMSCHS_x_TM [31:28] =0010b |
| Test SE0 NAK | XUSB_XHCI_OP_PORTPMSCHS_x_TM [31:28] =0011b |
| HS Test packet | XUSB_XHCI_OP_PORTPMSCHS_x_TM [31:28] =0100b |
| Force enable | XUSB_XHCI_OP_PORTPMSCHS_x_TM [31:28] =0101b |

The following steps provide an alternative way to place USB2.0 ports from xUSB controller into Test Mode through direct register accesses:

1. Connect any USB device to the port (this will prevent the controller from entering power down mode).

**Note:** "Any USB device" refers to any real device, such as HS uDISK or LS mouse.

2. Disable the auto-suspend for the controllers:

```
echo on > /sys/bus/usb/devices/usb1/power/control
echo on > /sys/bus/usb/devices/usb2/power/control
```

echo on > /sys/bus/usb/devices/usb3/power/control

3. Set PP (Port Power) in Disabled state by XUSB_XHCI_OP_PORTSC* bit [9] = 0.

   USB0: 0x70090460: XUSB_XHCI_OP_PORTSC_4

   USB1: 0x70090480: XUSB_XHCI_OP_PORTSC_6

   USB2: 0x70090490: XUSB_XHCI_OP_PORTSC_7

4. Set RS (Run/Stop) bit in the XUSB_XHCI_OP_USBCMD bit [0] = 0.

   0x70090020: XUSB_XHCI_OP_USBCMD

5. Wait for HCHalted (HCH) bit in the XUSB_XHCI_OP_USBSTS bit [0] = 1.

   0x70090024: XUSB_XHCI_OP_USBSTS

6. Set the xUSB USB2.0 Port Test Control Registers in PORTPMSCHS register to enable test patterns (refer to table “xUSB USB2.0 Port Test Control Registers” for additional information).

   USB0: 0x70090464: XUSB_XHCI_OP_PORTPMSCHS_4

   USB1: 0x70090484: XUSB_XHCI_OP_PORTPMSCHS_6

   USB2: 0x70090494: XUSB_XHCI_OP_PORTPMSCHS_7

**Note:** Per USB2.0 Specification, only a single downstream facing port can be in test_mode at a given time.

7. Disable Pad PD (Power Down) by clearing the XUSB_PADCTL_USB2_OTG_PADx_CTL_0_0 bit [27:26] = 0b’00.

   USB0: 0x7009F088: XUSB_PADCTL_USB2_OTG_PAD0_CTL_0_0

   USB1: 0x7009F108: XUSB_PADCTL_USB2_OTG_PAD2_CTL_0_0

   USB2: 0x7009F148: XUSB_PADCTL_USB2_OTG_PAD3_CTL_0_0

8. Plug in the test fixture to start USB2.0 eye diagram test.

**Note:** In Steps 3, 6, and 7, USB0 is USB0_D+/D- (B39/B40), USB1 is USB1_D+/D- (A38/A39), and USB2 is USB2_D+/D- (B42/B43).

## Notice

The information provided in this specification is believed to be accurate and reliable as of the date provided. However, NVIDIA Corporation ("NVIDIA") does not give any representations or warranties, expressed or implied, as to the accuracy or completeness of such information. NVIDIA shall have no liability for the consequences or use of such information or for any infringement of patents or other rights of third parties that may result from its use. This publication supersedes and replaces all other specifications for the product that may have been previously supplied.

NVIDIA reserves the right to make corrections, modifications, enhancements, improvements, and other changes to this specification, at any time and/or to discontinue any product or service without notice. Customer should obtain the latest relevant specification before placing orders and should verify that such information is current and complete.

NVIDIA products are sold subject to the NVIDIA standard terms and conditions of sale supplied at the time of order acknowledgement, unless otherwise agreed in an individual sales agreement signed by authorized representatives of NVIDIA and customer. NVIDIA hereby expressly objects to applying any customer general terms and conditions with regard to the purchase of the NVIDIA product referenced in this specification.

NVIDIA products are not designed, authorized or warranted to be suitable for use in medical, military, aircraft, space or life support equipment, nor in applications where failure or malfunction of the NVIDIA product can reasonably be expected to result in personal injury, death or property or environmental damage. NVIDIA accepts no liability for inclusion and/or use of NVIDIA products in such equipment or applications and therefore such inclusion and/or use is at customer's own risk.

NVIDIA makes no representation or warranty that products based on these specifications will be suitable for any specified use without further testing or modification. Testing of all parameters of each product is not necessarily performed by NVIDIA. It is customer's sole responsibility to ensure the product is suitable and fit for the application planned by customer and to do the necessary testing for the application in order to avoid a default of the application or the product. Weaknesses in customer's product designs may affect the quality and reliability of the NVIDIA product and may result in additional or different conditions and/or requirements beyond those contained in this specification. NVIDIA does not accept any liability related to any default, damage, costs or problem which may be based on or attributable to: (i) the use of the NVIDIA product in any manner that is contrary to this specification, or (ii) customer product designs.

No license, either expressed or implied, is granted under any NVIDIA patent right, copyright, or other NVIDIA intellectual property right under this specification. Information published by NVIDIA regarding third-party products or services does not constitute a license from NVIDIA to use such products or services or a warranty or endorsement thereof. Use of such information may require a license from a third party under the patents or other intellectual property rights of the third party, or a license from NVIDIA under the patents or other intellectual property rights of NVIDIA. Reproduction of information in this specification is permissible only if reproduction is approved by NVIDIA in writing, is reproduced without alteration, and is accompanied by all associated conditions, limitations, and notices.

ALL NVIDIA DESIGN SPECIFICATIONS, REFERENCE BOARDS, FILES, DRAWINGS, DIAGNOSTICS, LISTS, AND OTHER DOCUMENTS (TOGETHER AND SEPARATELY, "MATERIALS") ARE BEING PROVIDED "AS IS." NVIDIA MAKES NO WARRANTIES, EXPRESSED, IMPLIED, STATUTORY, OR OTHERWISE WITH RESPECT TO THE MATERIALS, AND EXPRESSLY DISCLAIMS ALL IMPLIED WARRANTIES OF NONINFRINGEMENT, MERCHANTABILITY, AND FITNESS FOR A PARTICULAR PURPOSE. Notwithstanding any damages that customer might incur for any reason whatsoever, NVIDIA's aggregate and cumulative liability towards customer for the products described herein shall be limited in accordance with the NVIDIA terms and conditions of sale for the product.

## Trademarks

NVIDIA, the NVIDIA logo, Jetson, and Tegra are trademarks and/or registered trademarks of NVIDIA Corporation in the U.S. and other countries. Other company and product names may be trademarks of the respective companies with which they are associated.

## Copyright