

TUSB926x Flash Burner

User's Guide



Literature Number: SLLU125D
August 2010–Revised October 2013

1	Introduction	5
1.1	Reference Material	5
1.2	Definitions	5
1.3	Required Equipment	5
2	Installing the Flash Burner Software	6
2.1	Running the Setup Program	6
2.2	Connecting the TUSB926x Based Hardware	8
3	Flash Burner GUI description	12
3.1	HID Interface	12
3.2	SCSI Interface	17
4	Using the Flash Burner Software	20
4.1	Opening the Flash Burner Software	20
4.2	Flash Burner GUI options	21
4.3	Editing USB Descriptors	23
4.4	Selecting a Compatible Device	24
4.5	Selecting a Compatible Firmware Binary File	25
4.6	Burning a Firmware Binary File Into the SPI Flash	26
4.7	Erasing Firmware in the SPI Flash	27
4.8	Exporting the Firmware Data to a File	28
5	Troubleshooting	30
5.1	Re-Installing Flash Burner Driver Instance Manually	30
5.2	TUSB926x Not Detected Due to Corrupted Firmware	33
5.3	TUSB926x Unrecognized by the Flash Burner GUI and Device Manager	33
5.4	GUI Succeeded, But Device is Not Properly Programmed	34
5.5	TUSB926x is Properly Programmed But has Yellow Alert in Device Manager	34

List of Figures

2-1.	.NET Framework 3.5 Installation	6
2-2.	Driver Installation Caution Examples	7
2-3.	Security Example for Windows™	7
2-4.	Software Installation Completed	8
2-5.	TUSB926x Device Instances With Blank or SPI Flash Not Connected	9
2-6.	TUSB926x Device Instances on HID-Enabled Firmware	10
2-7.	TUSB926x Device Instances on Mass-Storage-Only (No HID) Firmware	11
3-1.	Descriptors and Firmware Location	12
3-2.	Get HID Devices Flow Diagram	15
3-3.	GUI Burner or Re-Programmer Flow Diagram	16
3-4.	Listing Compatible Mass Storage Devices	18
3-5.	Erasing the Device's Flash Through SCSI Commands	19
4-1.	Flash Burner Software Locations	20
4-2.	Flash Burner GUI	21
4-3.	Flash Burner Software Version	21
4-4.	Selecting Options on the Flash Burner GUI	22
4-5.	Getting Current Serial Number	23
4-6.	Descriptors Info Group-Box	23
4-7.	Editing USB Descriptors	24
4-8.	List for Selecting a Compatible TUSB926x Device	25
4-9.	Selecting a Valid Firmware File	25
4-10.	Burning a Valid Firmware File into the SPI Flash	26
4-11.	SPI Flash Programming Succeeded	26
4-12.	Operating System Asking for a Reboot	26
4-13.	Erasing SPI Flash With HID-Enabled Firmware	27
4-14.	Erase Flash Command Succeeded	27
4-15.	Erasing the Device's Flash from USB Mass Storage Instance	28
4-16.	Export Button	28
4-17.	Saving the EEPROM Image	29
5-1.	TUSB926x Boot Loader Instance Not Properly Recognized	31
5-2.	Manual Flash Burner Driver Installation	32
5-3.	Skipping Windows™ Update for Driver Installation	32
5-4.	Automatic Driver Search	33
5-5.	Data Transmission During Flash Programming	34
5-6.	TUSB926x Based Device in Conflict	35
5-7.	Registry Cleanup Button	35
5-8.	User-Defined VID/PID Cleanup	36

List of Tables

3-1.	HID Reports	14
3-2.	SCSI Commands	17
5-1.	OP-Codes	34

Introduction

This document provides details on using the Flash Burner software with TUSB926x based devices. The software is a Windows-based tool allowing in-circuit programming of the SPI Flash via the universal serial bus (USB). A valid firmware file is necessary to properly use the TUSB926x device as a USB 3.0 to SATA bridge.

The term Flash Burner is equivalent to TUSB926x Flash Burner or Re-Programmer.

1.1 Reference Material

All materials referenced by this user's guide are on the TUSB926x Program Management and Design SharePoint sites, if not available from any external public source.

1.2 Definitions

The following are useful definitions for reference:

- BOT – Bulk-only transfer
- FW – Firmware
- GUI – Graphical user interface
- HID – Human interface device
- SCSI – Small computer system interface
- SPI – Serial peripheral interface
- UAS – USB-attached SCSI
- VID – Vendor ID
- PID – Product ID

1.3 Required Equipment

The following equipment is required to use the Flash Burner software:

- TUSB926x EVM board
- SPI Flash device (see [Figure 5-1](#) to verify supported Op-codes)
- USB interface Cable (Type A connector to Type B connector)
- PC that runs Windows XP™, or higher (32-bit or 64-bit) operating system (OS)
- TUSB926x firmware version 0.95 or greater

Installing the Flash Burner Software

2.1 Running the Setup Program

Extract the setup program on the preferred directory. Run the setup.exe file. The InstallShield Wizard guides the user through the installation process.

The Flash Burner program requires Microsoft .NET Framework 3.5™. The installer downloads and installs this framework version, if it is not present on the system. Ensure a proper internet connection because the installation is performed by a web download directly from the Microsoft™ website (see [Figure 2-1](#)).

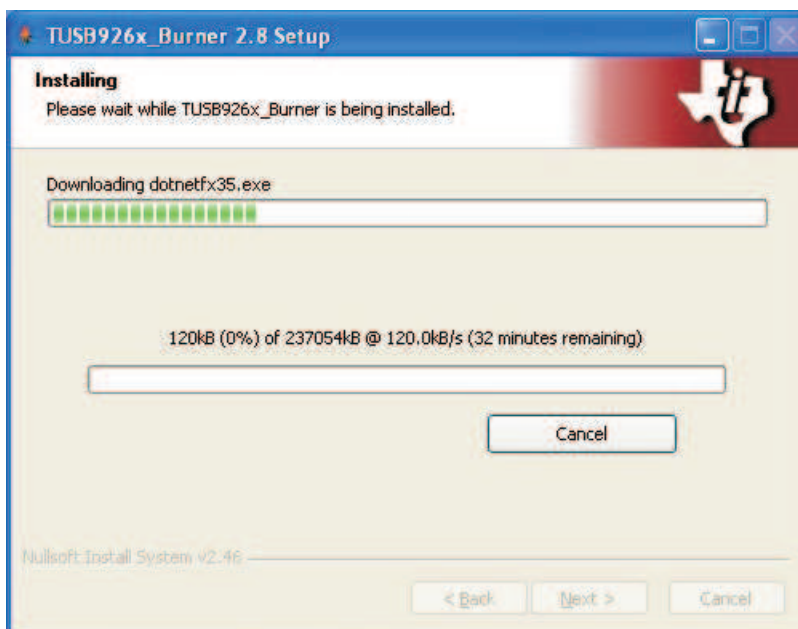


Figure 2-1. .NET Framework 3.5 Installation

After the .NET validation finishes, the installer copies the necessary files to the local disk and executes a driver co-installer, which installs the "FlashBurner" driver in the system. Depending on the system settings, a caution message or security window may appear during the driver installation process. When prompted, accept the driver installation as shown in [Figure 2-2](#) and [Figure 2-3](#).

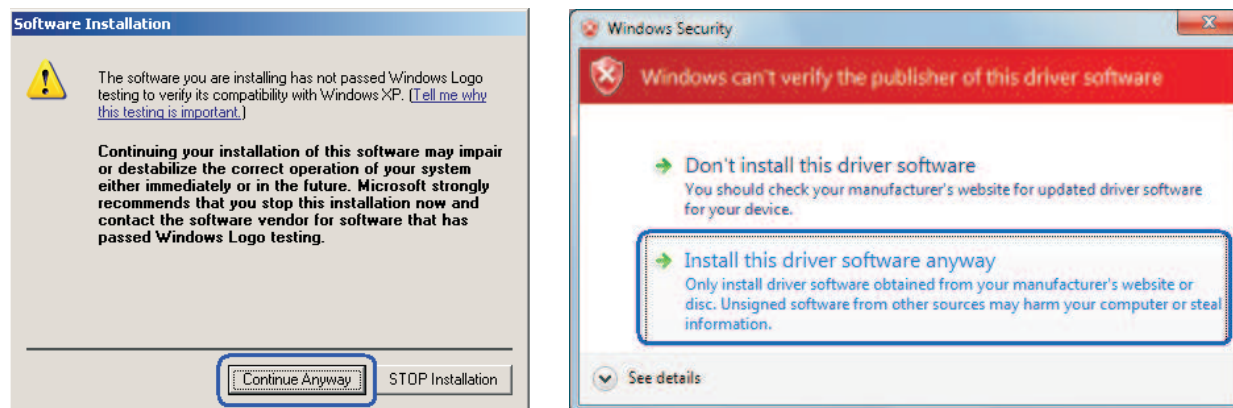


Figure 2-2. Driver Installation Caution Examples

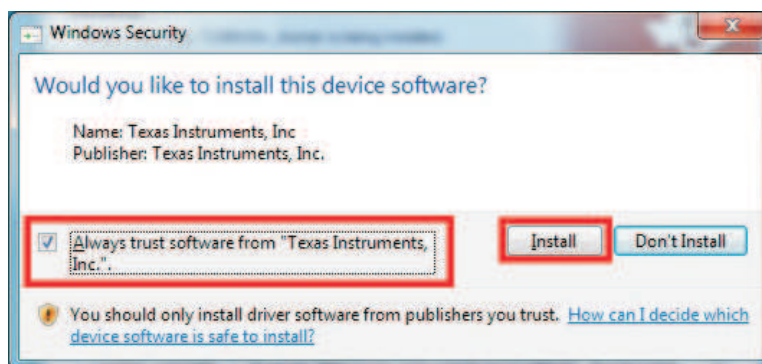


Figure 2-3. Security Example for Windows™

After all the necessary files have been copied into the system, the installer gives the option to restart the system to properly update the files.

TI recommends to choose “Reboot now” and click Finish when the installer wizard indicates the Flash Burner software installation is complete (see [Figure 2-4](#)).

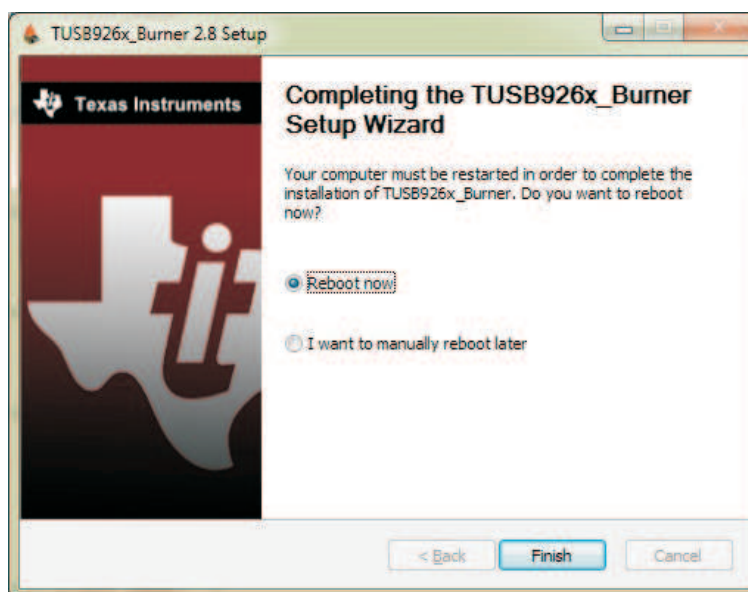


Figure 2-4. Software Installation Completed

2.2 Connecting the TUSB926x Based Hardware

Connect the TUSB926x EVM board to any USB port available on the PC.

The OS enumerates the TUSB926x EVM board in different ways, which depend on the content of the SPI Flash of the device:

1. Blank or SPI Flash not connected (see [Figure 2-5](#))
2. HID-enabled firmware (see [Figure 2-6](#))
3. Mass-storage-only (no HID) (see [Figure 2-7](#))

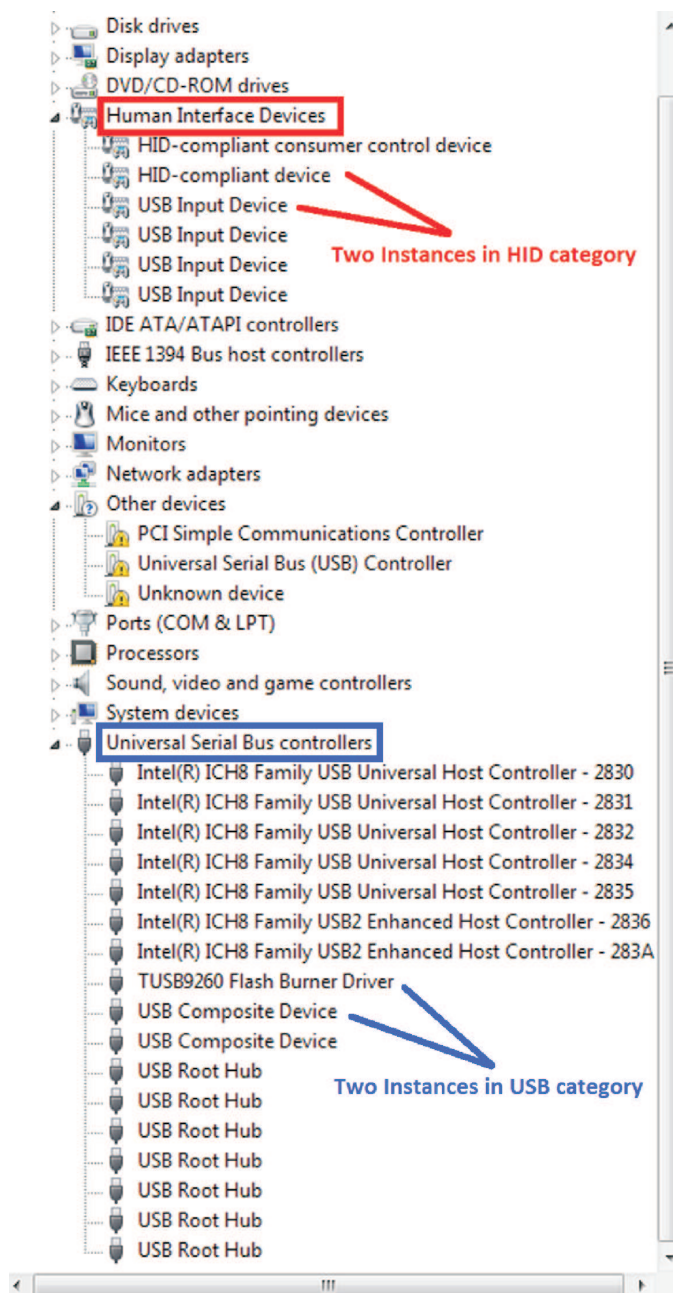


Figure 2-5. TUSB926x Device Instances With Blank or SPI Flash Not Connected

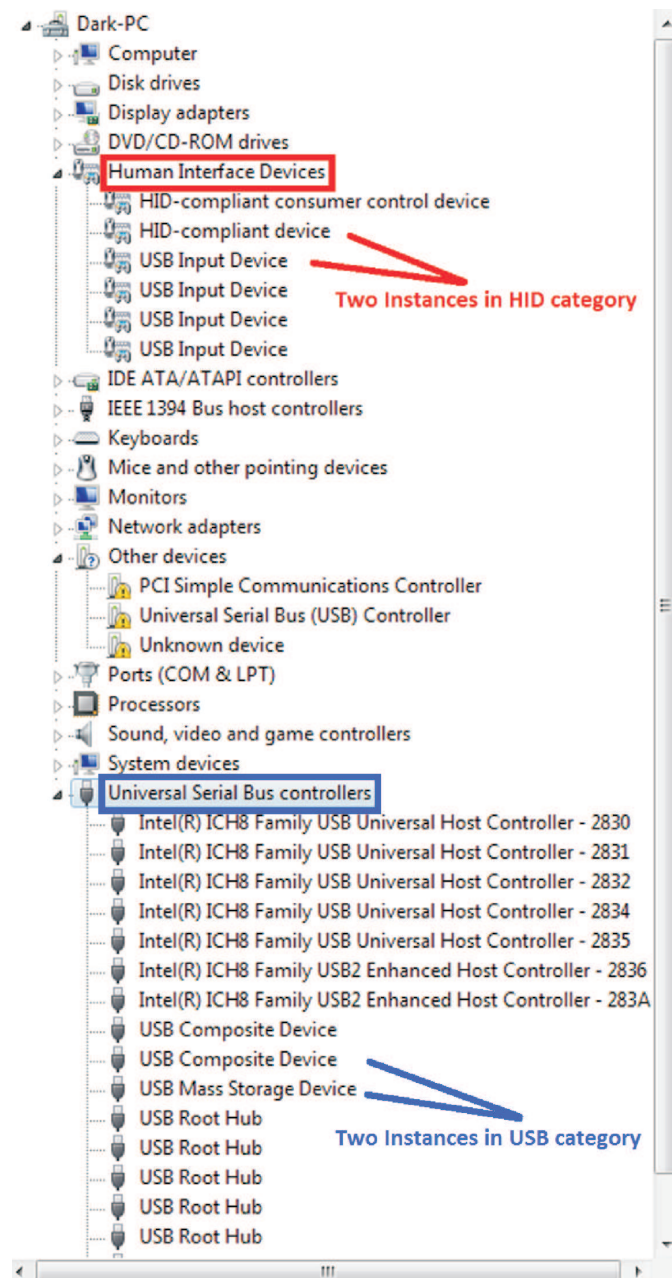


Figure 2-6. TUSB926x Device Instances on HID-Enabled Firmware

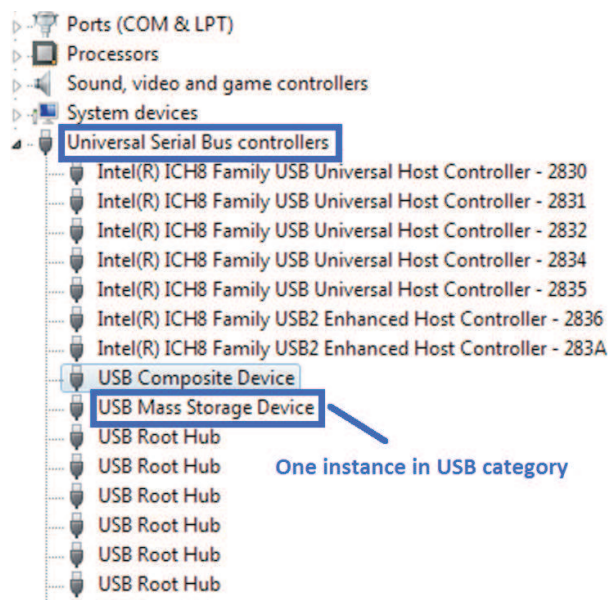


Figure 2-7. TUSB926x Device Instances on Mass-Storage-Only (No HID) Firmware

3.1 HID Interface

- **USB_HID_SETUP_DOWNLOAD_DATA**

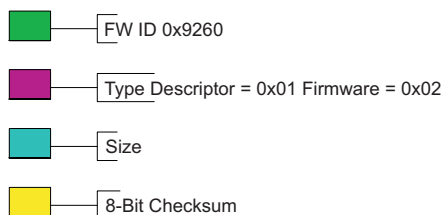
[illegible]

Figure 3-1. Descriptors and Firmware Location

Only the boot loader device supports this HID report.

- **USB_HID_RESET_FLASH_BURNER_DEVICE**

This report instructs the device to perform a soft reset. Both the Boot Loader and the firmware support this HID report.

- **USB_HID_POISON_FLASH**

This report instructs the boot loader to erase the content of the SPI Flash. Only the firmware supports this HID report.

- **USB_HID_ENABLE_REPROGRAM**

This report prepares a device that already has firmware to be re-programmed. Only the firmware supports this HID report.

- **USB_HID_IS_FLASH_PRESENT**

This report helps the user determine whether the device has an SPI Flash or not. Only the boot loader supports this HID call.

- **USB_HID_GET_FIRMWARE_VERSION**

Through this report, the user can determine the current firmware version operating in our device. Only the firmware supports this HID report.

[Table 3-1](#) describes the data contained on these reports.

Table 3-1. HID Reports

HID REPORT	BYTE 0	BYTE 1	BYTE 2	BYTE 3	BYTE 4	BYTE 5	BYTE 6	BYTE 7	BYTE 8
USB_HID_SETUP_DOWNLOAD_DATA	Opcode = 0x01	Should be set to 0x00	Not used; value should be set to 0x00	Not used; value should be set to 0x00	Not used; value should be set to 0x00	Not used; value should be set to 0x00	Not used; value should be set to 0x00	Not used; value should be set to 0x00	Not used; value should be set to 0x00
USB_HID_RESET_FLASH_BURNER_DEVICE	Opcode = 0x02	Should be set to 0x00	Not used; value should be set to 0x00	Not used; value should be set to 0x00	Not used; value should be set to 0x00	Not used; value should be set to 0x00	Not used; value should be set to 0x00	Not used; value should be set to 0x00	Not used; value should be set to 0x00
USB_HID_POISON_FLASH	Opcode = 0x03	Not used Value should be set to 0x00	Not used; value should be set to 0x00	Not used; value should be set to 0x00	Not used; value should be set to 0x00	Not used; value should be set to 0x00	Not used; value should be set to 0x00	Not used; value should be set to 0x00	Not used; value should be set to 0x00
USB_HID_ENABLE_REPROGRAM	Opcode = 0x07	Valid values: 0x00 or 0x01. 0x00 = Re-programming disabled 0x01 = Re-programming enabled	Not used; value should be set to 0x00	Not used; value should be set to 0x00	Not used; value should be set to 0x00	Not used; value should be set to 0x00	Not used; value should be set to 0x00	Not used; value should be set to 0x00	Not used; value should be set to 0x00
USB_HID_IS_FLASH_PRESENT	Opcode = 0x08	Valid values: 0x00 or 0x01. 0x00 = Flash 0x01 = No flash	Not used; value should be set to 0x00	Not used; value should be set to 0x00	Not used; value should be set to 0x00	Not used; value should be set to 0x00	Not used; value should be set to 0x00	Not used; value should be set to 0x00	Not used; value should be set to 0x00
USB_HID_GET_FIRMWARE_VERSION	Opcode = 0x09	Not used; value should be set to 0x00	Not used; value should be set to 0x00	Not used; value should be set to 0x00	Not used; value should be set to 0x00	Not used; value should be set to 0x00	Not used; value should be set to 0x00	Firmware minor version	Firmware major version

To recognize an HID device as TUSB926x compatible, the device must respond with a 0x9260 to an HID Feature report request. [Figure 3-2](#) describes this process in detail.

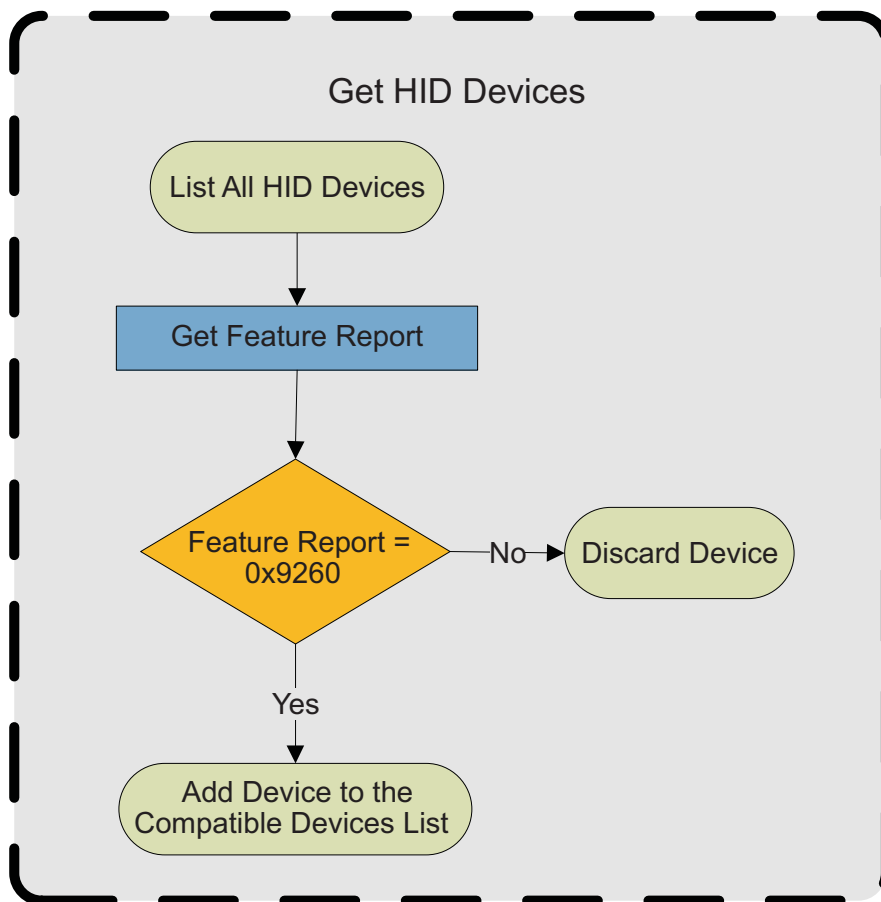


Figure 3-2. Get HID Devices Flow Diagram

After the GUI has successfully detected at least one TUSB926x HID compatible device, the user needs to determine whether the device has already been programmed or not. The user can achieve this by getting the VID and PID of the device.

If a device has VID = 0x0451 and PID = 0x926B, the user can assume the device is not programmed and is ready to be programmed. Otherwise, the user needs to take the device to its original configuration (by erasing or poisoning the SPI Flash of the device) before sending the new data to be programmed.

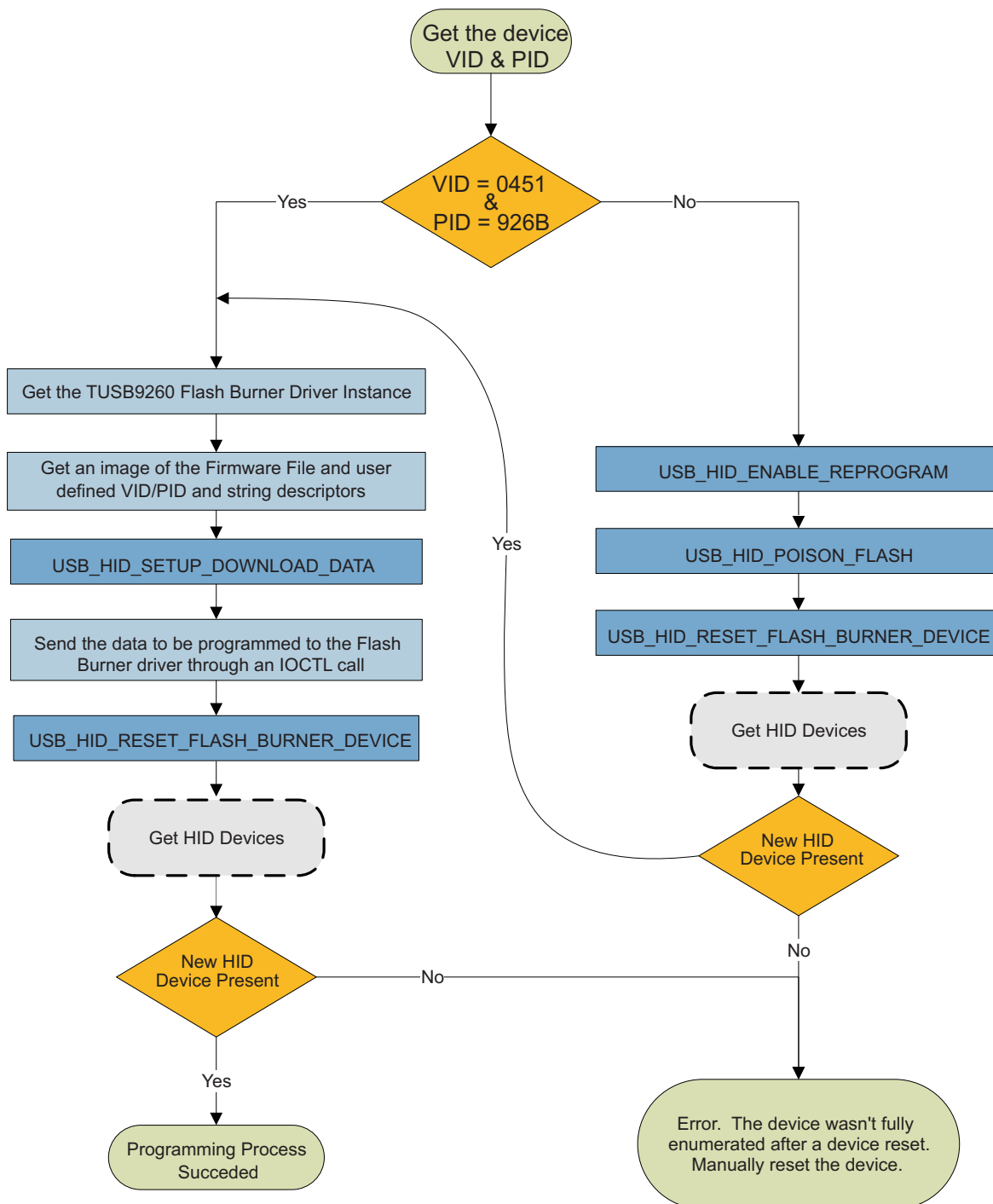


Figure 3-3. GUI Burner or Re-Programmer Flow Diagram

3.2 SCSI Interface

When a previously-burned firmware in the SPI Flash of the device contains a mass-storage-only interface, it must have additional means to erase the flash of the device when updating the firmware content. Beginning on Firmware version 0.95, a set of TI vendor-specific SCSI commands were implemented to provide the necessary means to communicate specific commands to the device whenever the HID interface is missing. The currently implanted SCSI commands are:

- **SCSI_TI_FLASH_UNLOCK**

Similar to the USB_HID_ENABLE_REPROGRAM, this command prepares the device to be re-programmed. This command serves as a lock to ensure no accidental flash erasing is performed.

- **SCSI_TI_FLASH_ERASE**

As with the USB_HID_POISON_FLASH report, this command instructs the device to erase the content of the SPI Flash.

- **SCSI_TI_GET_PID**

This SCSI command returns a constant data (0x9260) that helps to identify the device among other storage devices in the system, similar to the HID Feature report implementation

- **SCSI_TI_GET_FIRMWARE_VERSION**

Through this command, the user can determine the current firmware version operating in the device.

- **SCSI_TI_GET_USB_SPEED**

Through this command, the user receives an indicator representing the current USB connection speed.

- **SCSI_TI_DEVICE_RESET**

Similar to the USB_HID_RESET_FLASH_BURNER_DEVICE report, this command instructs the device to perform a soft reset. This command is specially required after erasing the flash of the device, so the BootLoader instance can re-initialize.

- **SCSI_TI_READ_FLASH**

This command reads a block of data from the SPI Flash. This command dumps the content of the SPI Flash into a file.

Table 3-2 describes the data contained on the above detailed commands.

Table 3-2. SCSI Commands

SCSI COMMAND	COMMAND DATA	RETURN DATA
SCSI_TI_FLASH_UNLOCK	0xE1	No data returned
SCSI_TI_FLASH_ERASE	0xE2	No data returned
SCSI_TI_GET_PID	0xE3	2 bytes – 0x9260
SCSI_TI_GET_FIRMWARE_VERSION	0xE4	2 bytes
		Byte 0 – Firmware minor version
		Byte 1 – Firmware major version
SCSI_TI_GET_USB_SPEED	0xE5	1 byte
		USB_LOW_SPEED = 0
		USB_FULL_SPEED = 1
		USB_HIGH_SPEED = 2
		USB_SUPER_SPEED = 3
		USB_SPEED_UNKNOWN = 4
SCSI_TI_DEVICE_RESET	0xE6	No data returned
SCSI_TI_READ_FLASH	0xE7	Up to 255 bytes of data read from the SPI Flash

Figure 3-4 describes the TI vendor-specific SCSI commands usage while listing compatible TUSB926x based disks over the Flash Burner GUI.

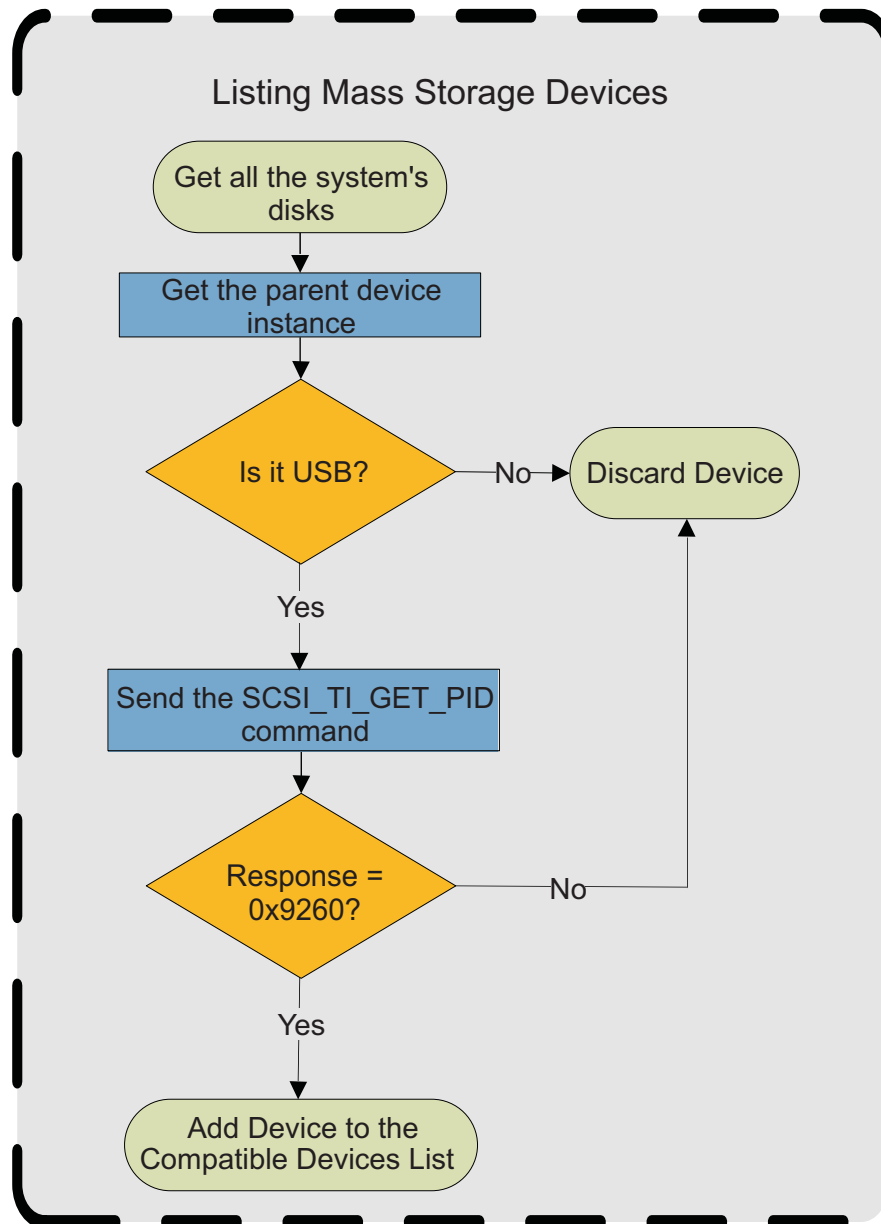


Figure 3-4. Listing Compatible Mass Storage Devices

During the flash erasing process, TI vendor-specific SCSI commands are issued in the sequence described in [Figure 3-5](#).

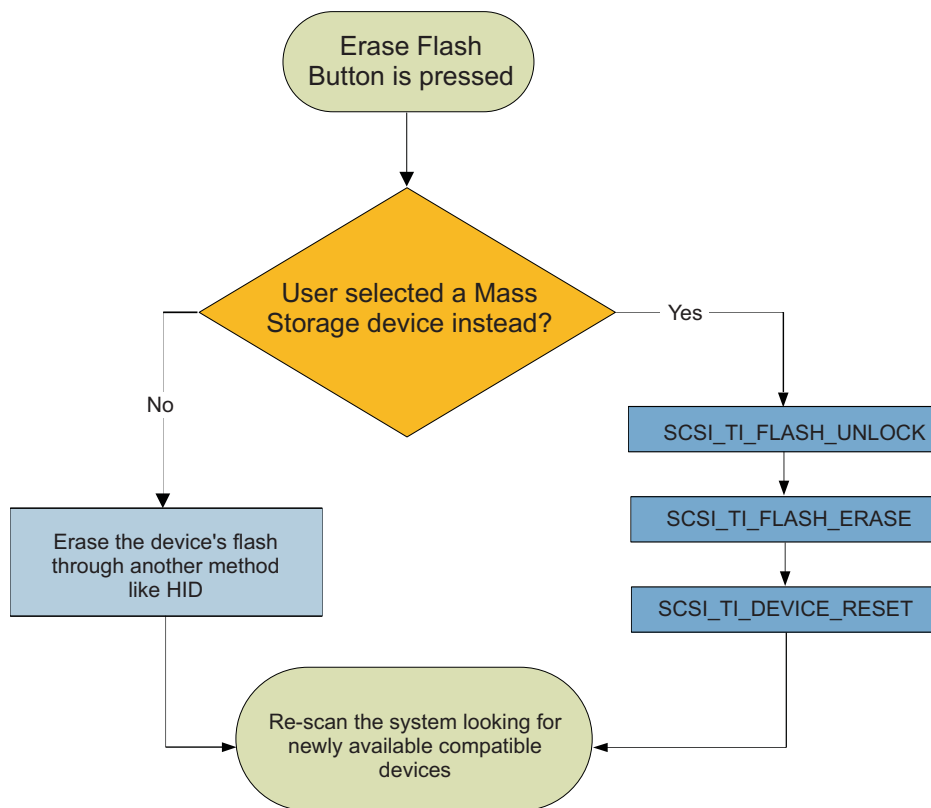


Figure 3-5. Erasing the Device's Flash Through SCSI Commands

Using the Flash Burner Software

4.1 Opening the Flash Burner Software

After checking that no missing instances of the TUSB926x EVM are present at the Device Manager, the user can access the Flash Burner utility by clicking on the “TUSB926x Flash Burner” shortcut added to desktop, or by going to “Start → Texas Instruments Inc → TUSB926_FlashBurner → TUSB926x Flash Burner” (see [Figure 4-1](#)).

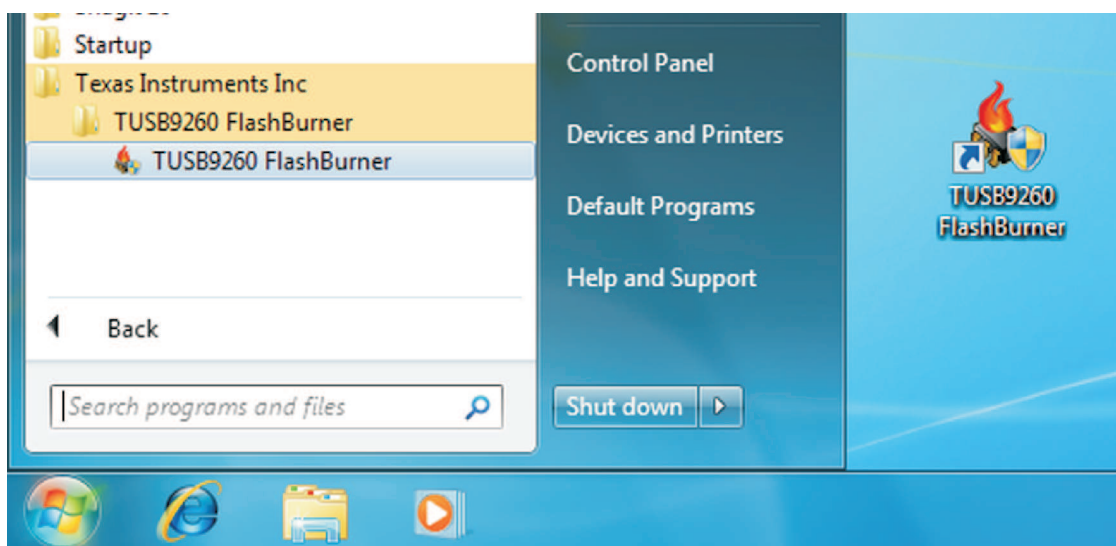


Figure 4-1. Flash Burner Software Locations

NOTE: Administrator rights are required under Windows Vista and higher operating systems to execute this application.

After executing the TUSB926x Flash Burner application, the following user interface appears:

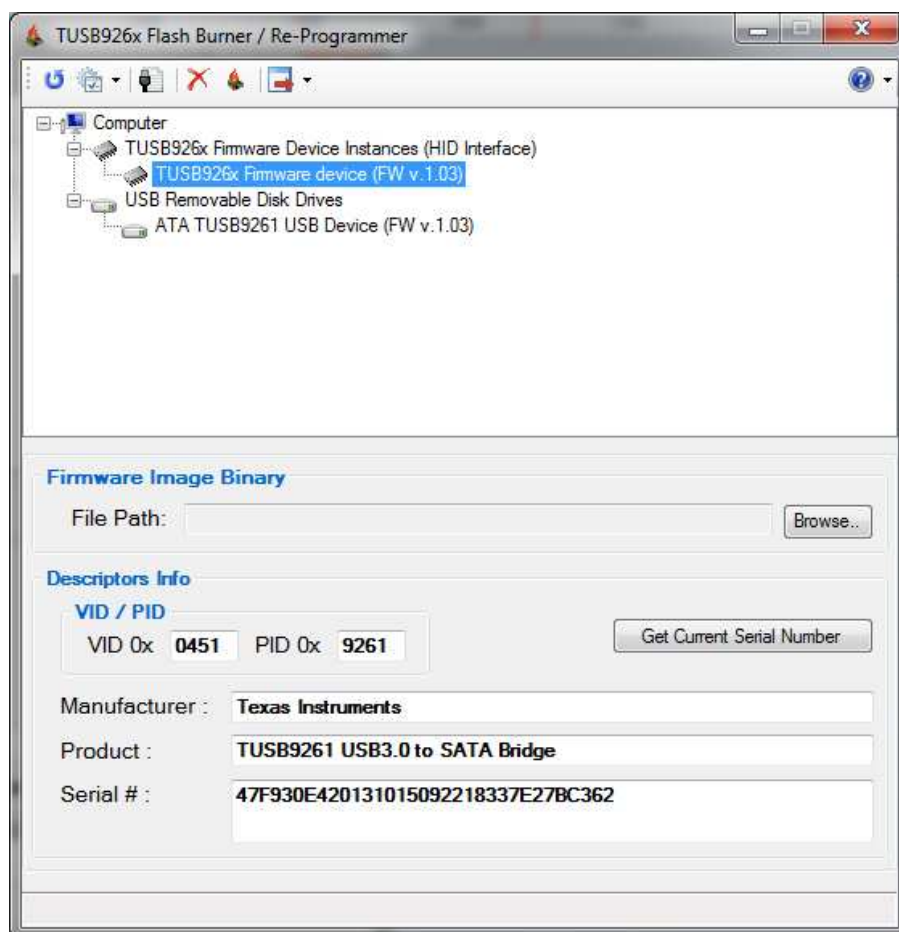


Figure 4-2. Flash Burner GUI

To check for the GUI version in use, click on the Help drop-down menu and choose “About” to display the application information (see Figure 4-3).

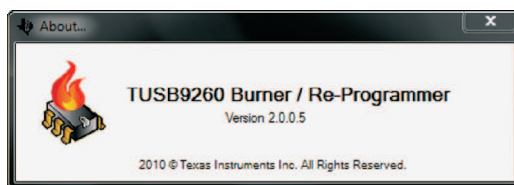


Figure 4-3. Flash Burner Software Version

4.2 Flash Burner GUI options

The Options menu enables the user to change different aspects of the Flash Burner GUI configuration.

To access the options menu items, simply click the Options button (see Figure 4-4).

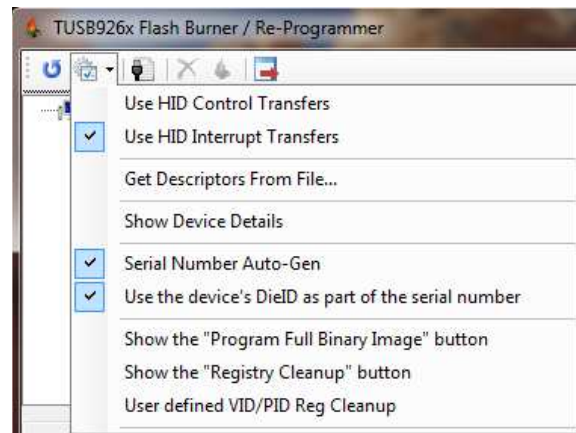


Figure 4-4. Selecting Options on the Flash Burner GUI

The following configuration aspects can be changed from the Options menu:

- (a) **HID transfer type:** Users can select between using either “Control Transfers” or “Interrupt Transfers” (default).

NOTE: If “HID Control Transfers” is used, make sure the device is PG2.5 and firmware v0.83, or better; otherwise, an error message displays and the device might become unresponsive, until the device is reset.

- (b) **Get Descriptors from File:** By setting this option, the user can get a set of descriptor settings from a descriptors file (*.desc), previously generated with the “Advance Descriptors Editor” (refer to [Section 4.3](#) for more details).

NOTE: By un-checking this option, the application uses the default descriptor settings.

- (c) **Show Device Details:** By setting this option, the application shows a panel displaying additional information about the selected device.
- (d) **Serial Number Auto-Generator:** By setting this option, the Flash Burner GUI automatically generates a unique serial number for every programmed device. The auto-generated serial number is made of:
- Random number (8 chars)
 - Date and time (Year – 4 chars, Month – 2 chars, Day – 2 chars, Hour – 2 chars, Minutes – 2 chars, Seconds – 2 chars, and Milliseconds – 3 chars)
 - Serial number of the HDD running Windows™ (8 chars)
- (e) **Use DieID for serial number:** When this option is set, part of the serial number reported by the TUSB926x string descriptors is overwritten with the device Die ID (a unique ID number which is generated during the device manufacturing process). This option is set by default.
- (f) **Show the Program Full Binary Image button:** When setting this option, a new button is shown on the toolbar at the top of the utility; that button allows the user to program the selected binary image without adding the formatting and descriptors data. This can be used to program a firmware file previously generated using the “Export” button. (Please refer to section 4.8 for additional details)
- (g) **Show the Registry Cleanup button and User-defined VID/PID Registry Cleanup:** Allows the user to clean up the Windows™ registry from possible conflicts originated when programming multiple devices on the same system with different configurations (observed only on development environments). Please refer to [Section 5.5](#) for additional details on how to use these options.

Additionally to the serial number auto-generation feature, users can also get the serial number from its current device just in case they need to keep it for upcoming firmware updates.

To achieve this, after selecting a compatible device from the Flash Burner GUI device list, the Get Current Serial Number button is displayed within the Descriptors Info group-box. By clicking this button (see [Figure 4-5](#)) the Flash Burner GUI gets the current serial number on the device and automatically disables the serial number auto-generation feature if it is enabled.



Figure 4-5. Getting Current Serial Number

4.3 Editing USB Descriptors

USB descriptors provide the host with all the necessary information to describe the USB device, so any change to these values must be done carefully.

Basic descriptor information can be changed from the TUSB926x Flash Burner GUI main form. Within the Descriptors Info group-box there is a series of text box controls (see [Figure 4-6](#)) that lets the user enter customized information about the device such as:

- **VID** – 4 characters long (assigned by USB-IF)
- **PID** – 4 characters long (assigned by the manufacturer)
- **Manufacturer String Descriptor** – Maximum 30 characters long
- **Product String Descriptor** – Maximum 30 characters long
- **Serial Number String Descriptor** – Must be unique for each device, maximum 64 characters long

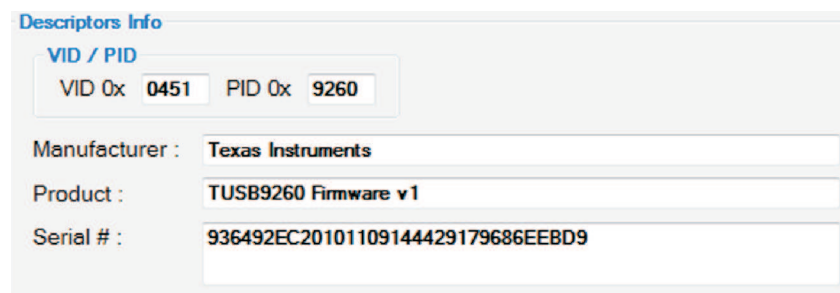


Figure 4-6. Descriptors Info Group-Box

The Flash Burner GUI also provides the means to edit additional descriptors information through the Advanced Descriptors Editor tool.

To open the “Advanced Descriptors editor” tool interface, click on the editor button located on the tool bar menu at the top of the Flash Burner GUI.

Once the descriptors editor is open (see [Figure 4-7](#)), the user can check the value of each individual descriptor to be used, edit additional descriptors data, and enable or disable some of the interfaces used by the device.

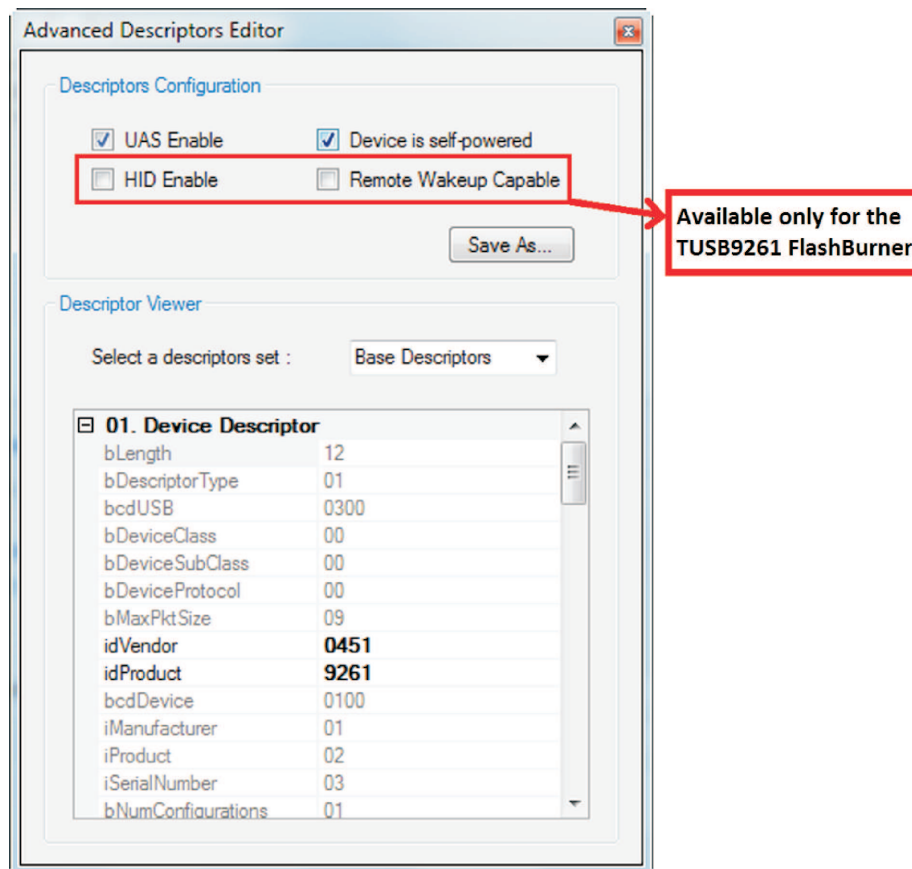


Figure 4-7. Editing USB Descriptors

Through this tool, users can also save the configuration for the descriptors to a descriptors (*.desc) file for future use. That file can later be loaded from the Options menu (refer to [Section 4.2](#)).

To save the current descriptors to a file, simply click on the Save As button, select a file name, and location from the resulting dialog and click OK.

NOTE: Make sure that all the enabled interfaces are supported by the firmware that is to be burned in the SPI Flash.

4.4 Selecting a Compatible Device

The Flash Burner GUI automatically identifies all the compatible devices already connected on the system. These compatible devices are listed on the top of the Flash Burner application instance. Through this list, the user can choose among all the TUSB926x based devices detected (see [Figure 4-8](#)). For this example, a TUSB926x EVM board was connected with a valid firmware image. Depending on the device selected, the toolbar buttons are enabled, so the user can perform any of the available tasks.

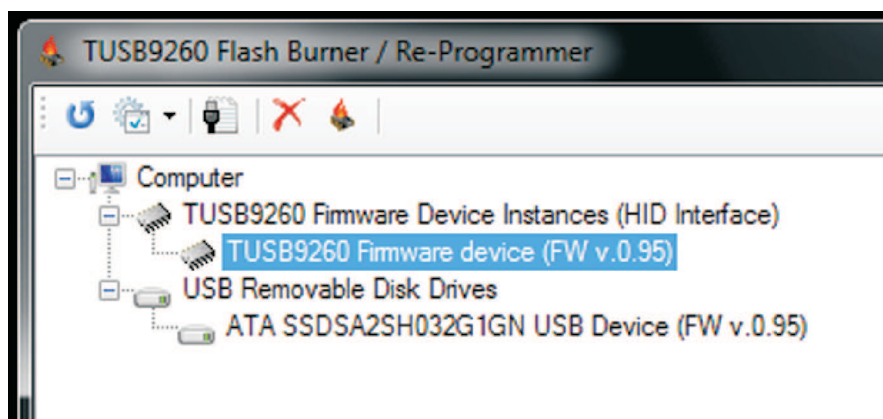


Figure 4-8. List for Selecting a Compatible TUSB926x Device

4.5 Selecting a Compatible Firmware Binary File

In the middle of the Flash Burner GUI, there is a group-box identified as Firmware Image Binary, which has a browse button that lets the user choose the *.BIN or *.HEX file to be burned into the SPI Flash. Click on this Browse button and select the appropriate firmware file located in the system. Click on Open after (see [Figure 4-9](#)).

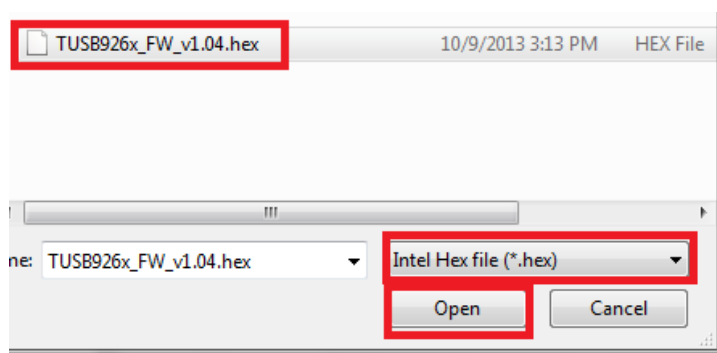


Figure 4-9. Selecting a Valid Firmware File

4.6 Burning a Firmware Binary File Into the SPI Flash

After selecting a valid firmware file, the user can now click on the Program button (see [Figure 4-10](#)).

The status bar indicates that the programming process has started by displaying a message stating that “Programming is in progress...”.

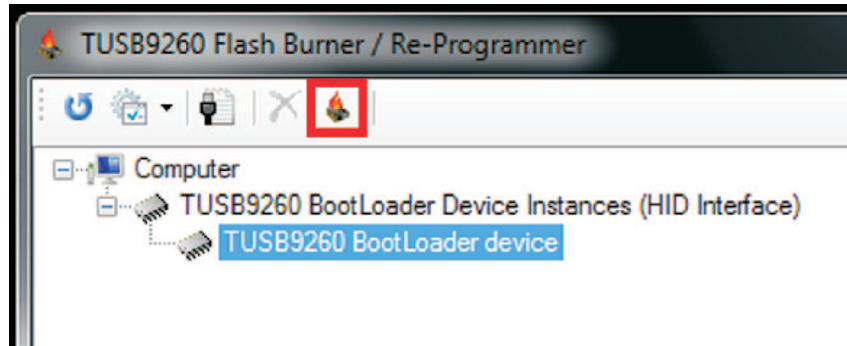


Figure 4-10. Burning a Valid Firmware File into the SPI Flash

When the firmware file has been properly burned into the SPI Flash, a message box showing “Flash Programming Succeeded” appears. Click OK to continue (see [Figure 4-11](#)).

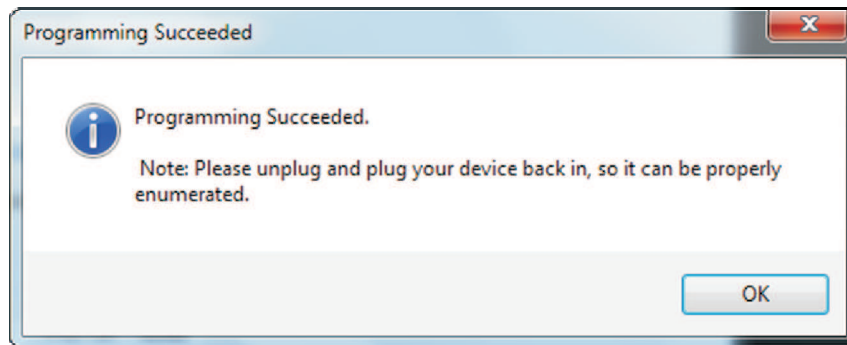


Figure 4-11. SPI Flash Programming Succeeded

TI recommends a reboot after successfully programming a firmware file into the SPI Flash device. Although not mandatory, the OS sometimes asks the user to perform this action.

When prompted by the system, click Yes to restart the computer (see [Figure 4-12](#)).

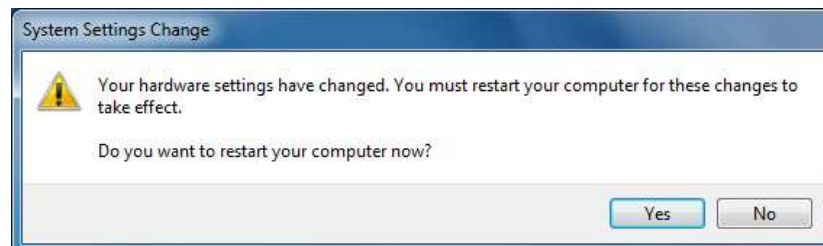


Figure 4-12. Operating System Asking for a Reboot

4.7 Erasing Firmware in the SPI Flash

To erase the content of the SPI Flash, click the Erase Flash button to issue the erase flash command on the device.

There are two different ways to erase the SPI Flash, depending on the firmware already present:

1. HID-enabled firmware:

- In this case, select the preferred HID interface instance that is shown in the compatible device list, and then click the Erase Flash button (see [Figure 4-13](#)).



Figure 4-13. Erasing SPI Flash With HID-Enabled Firmware

- Click OK after the erase flash command succeeds (see [Figure 4-14](#)).

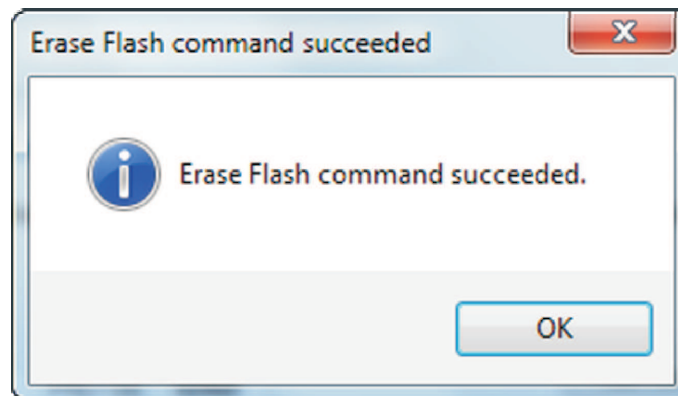


Figure 4-14. Erase Flash Command Succeeded

- After the SPI Flash is erased, the TUSB926x board re-enumerates with its Boot Loader instance (see [Figure 2-7](#)).
- ### 2. Mass-storage-only firmware (no HID)
- When having a firmware without an HID instance, the Erase Flash button can also be accessed when selecting the "USB Mass Storage" instance of our device under the "USB Removable Disk Drives" category.
- From a user perspective, this method is exactly the same as with the HID instance of the device, that is simply select the Mass Storage device instance and click the Erase Flash Button (see [Figure 4-15](#)). This time the GUI sends the erase flash command through TI vendor-specific SCSI commands.



Figure 4-15. Erasing the Device's Flash from USB Mass Storage Instance

- Click OK after the erase flash command succeeds (see [Figure 4-14](#)).
- After the SPI Flash is erased, the TUSB926x board re-enumerates with its Boot Loader instance.

NOTE: Only firmware version 0.95 and above support the TI (vendor) specific SCSI commands required by this function. Please make sure to use the appropriate firmware version.

4.8 Exporting the Firmware Data to a File

The TUSB926x Flash Burner GUI is also helpful when a user wants to burn the SPI Flash data using a method other than the FlashBurner GUI, as it can export all the required data such as the USB descriptors, checksums, and firmware in the appropriate format (as shown in [Figure 3-1](#)).

To generate and export the formatted binary image to a file, follow the steps described in [Section 4.3](#) and [Section 4.5](#) and click the Export Formatted Binary button within the Export Menu button as shown in [Figure 4-16](#).

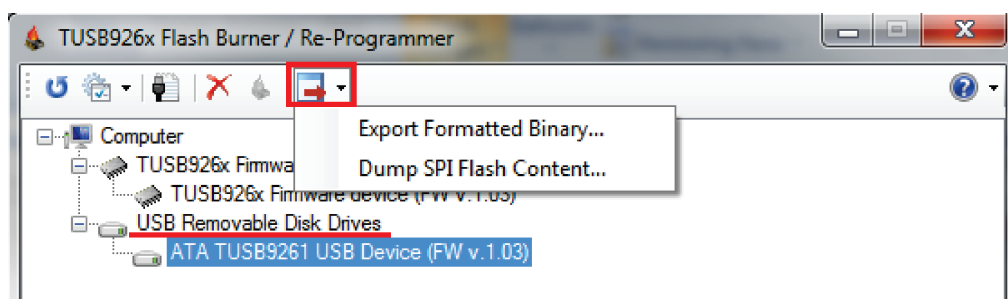


Figure 4-16. Export Button

When prompted, select a folder to store the EEPROM image, and input the desired name and format. Finally, click the Save button as shown in [Figure 4-17](#).

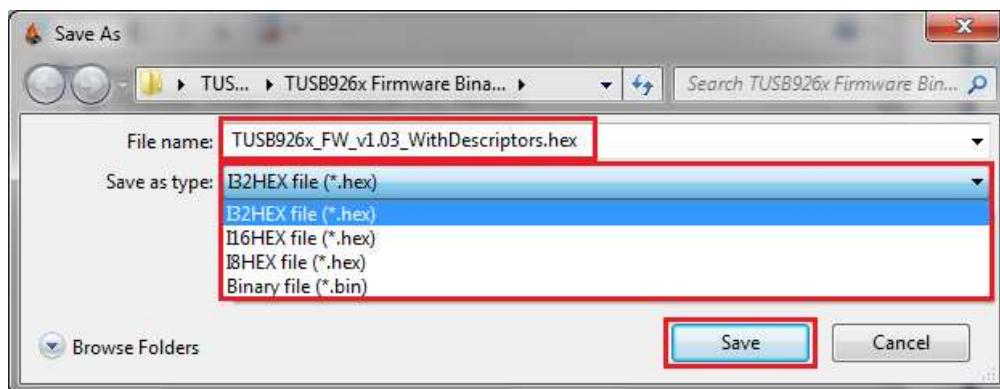


Figure 4-17. Saving the EEPROM Image

Additionally, by using the “Dump SPI Flash Content...” users can extract the data currently stored on the SPI flash and save it into a file. Please note that this function is only supported through vendor-specific SCSI commands, so the button to use is only enabled when selecting a device instance under the “USB Removable Disk Drives” category.

Troubleshooting

During the TUSB926x EVM enumeration process, there might be problems that do not allow the user to run the Flash Burner software in an adequate way. All instances should be present in order to burn or reprogram the SPI Flash.

The following are the most common cases that may arise when using the Flash Burner software:

5.1 Re-Installing Flash Burner Driver Instance Manually

This section details how to manually install the Flash Burner driver in the system, although it is not necessary because the Flash Burner GUI automatically installs the driver when required.

In case the TUSB926x EVM board initializes with a “TUSB926x Boot Loader” yellow alert instance (see [Figure 5-1](#)), implement the following steps to perform a manual driver installation:

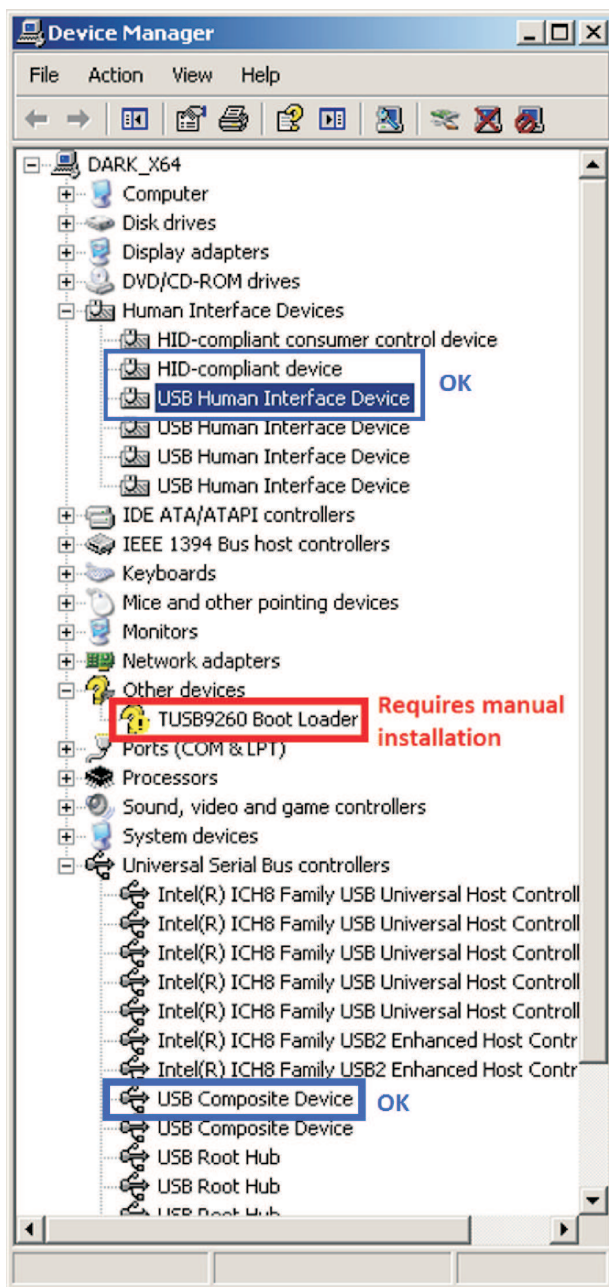


Figure 5-1. TUSB926x Boot Loader Instance Not Properly Recognized

Right-click on the “TUSB926x Boot Loader” instance and select the “Update Driver...” option (see [Figure 5-2](#)).

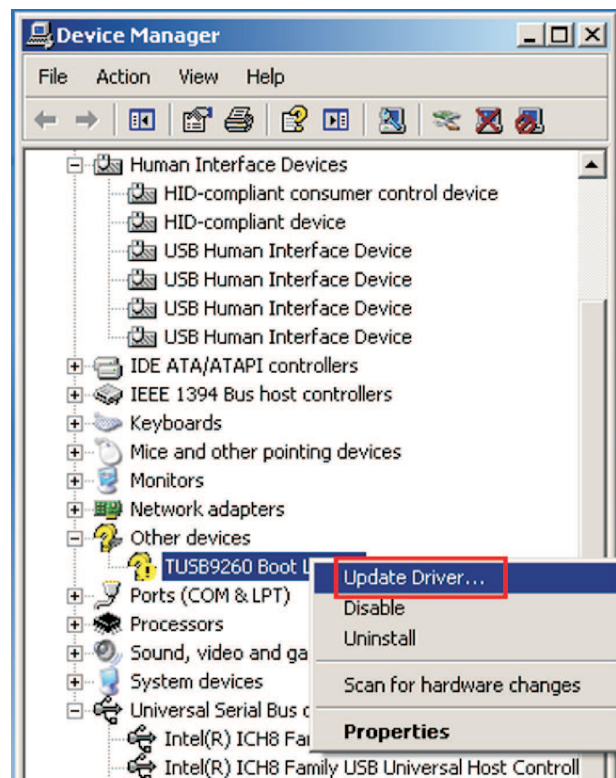


Figure 5-2. Manual Flash Burner Driver Installation

After some seconds, a Hardware Update Wizard window appears. Select “No, not this time” and click Next (see Figure 5-3).

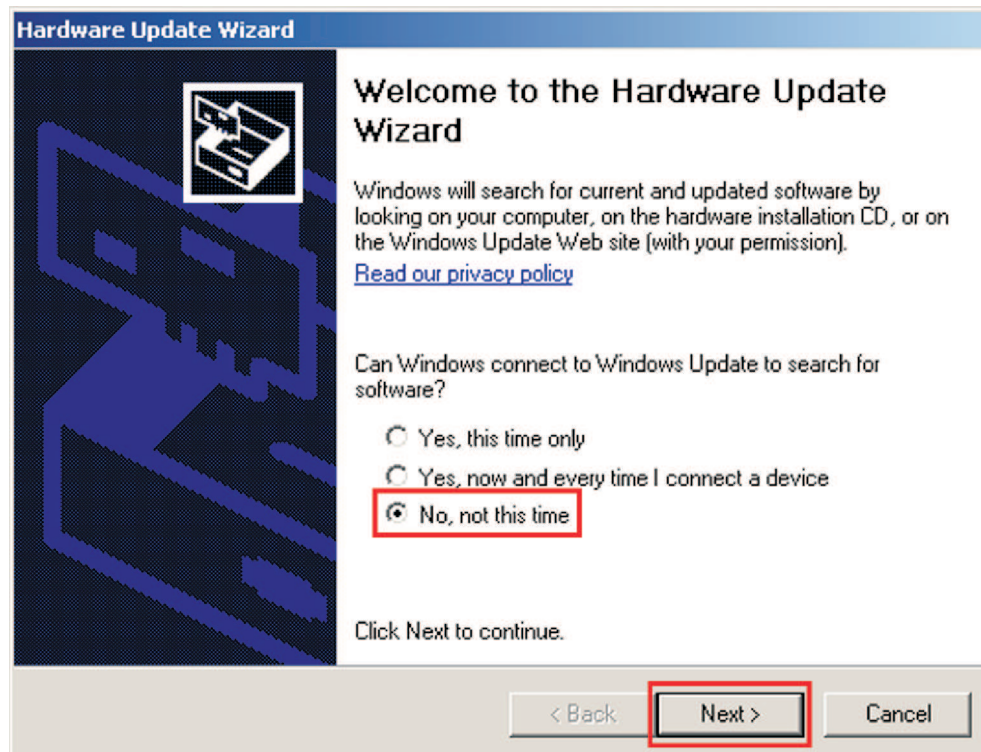


Figure 5-3. Skipping Windows™ Update for Driver Installation

Since the Flash Burner installer has already copied the proper drivers to the system, the user can now select “Install the software automatically...” and click Next to start searching for the proper drivers for such hardware instance. Wait until the yellow alert disappears and the instance is properly enumerated under the “USB controllers” category (see [Figure 5-4](#)).

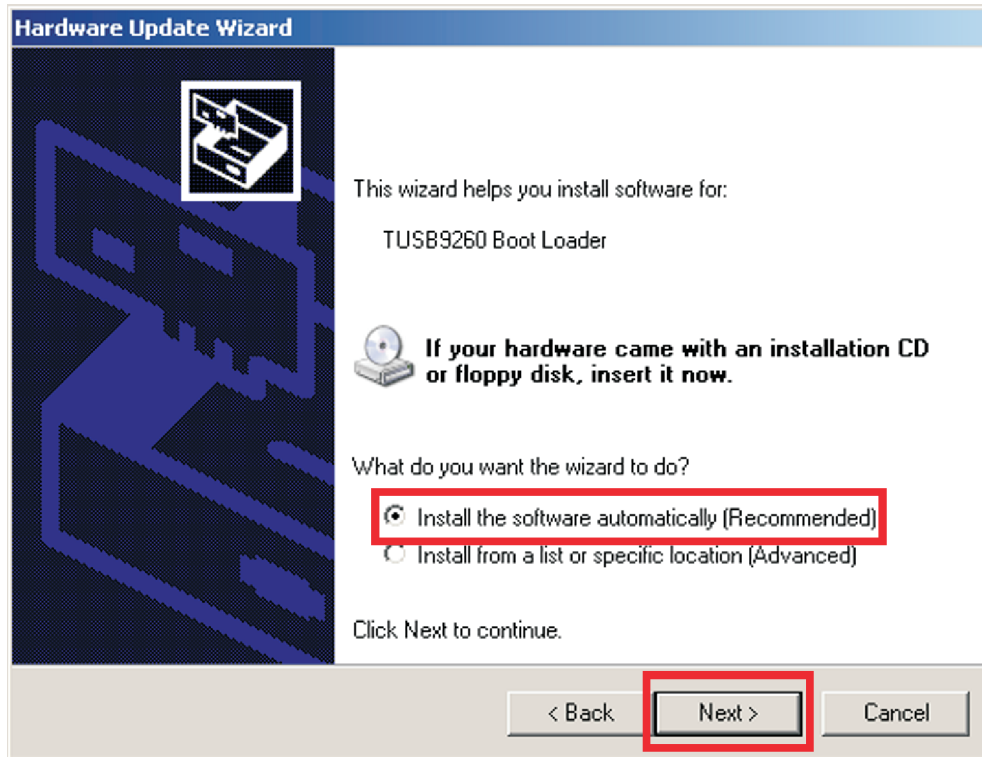


Figure 5-4. Automatic Driver Search

5.2 TUSB926x Not Detected Due to Corrupted Firmware

In case the TUSB926x EVM is not detected by the Flash burner GUI app, it is possible that the SPI Flash device was not properly programmed, or its firmware image is corrupted. To get the TUSB926x EVM detected, let it enumerate using the boot loader embedded in the TUSB926x device. To do so, please perform the following steps:

1. Close the Flash Burner GUI application.
2. Unplug the TUSB926x EVM.
3. Remove the SPI Enable jumper.
4. Plug the TUSB926x EVM and wait until it is properly enumerated and detected.

After the TUSB926x boot loader is enumerated, follow these additional steps to properly re-program the SPI Flash:

1. Open the Flash Burner GUI application.
2. Put the SPI Enable jumper back in its original position.
3. Follow [Section 4.2](#) through [Section 4.4](#). The user can now select the proper device and burn the desired USB descriptors and firmware in the SPI flash.

5.3 TUSB926x Unrecognized by the Flash Burner GUI and Device Manager

In some cases, after burning a new firmware on the device, or after installing either the Flash Burner, or the Erase Flash driver in one of the instances of the device, Windows™ can take a considerably long time to properly enumerate the device. While this happens, the device might be listed as “Unknown Device” in the Device Manager and the Flash Burner GUI does not show any of the device instances.

If that is the case, simply unplug and plug the device back in, and it should be properly enumerated in both the Flash Burner GUI and Device Manager.

5.4 GUI Succeeded, But Device is Not Properly Programmed

When the Flash Burner GUI displays the “Programming Succeeded” message, it indicates that the TUSB926x device acknowledged the reception of the firmware data; however, there is no implemented procedure to verify that the firmware data was indeed programmed into the SPI flash.

If the device was not properly programmed after the GUI “Programming Succeeded” message is displayed:

1. Unplug and plug the device back in, and wait for the device to be fully enumerated by the Device Manager.
2. Verify the SPI memory supports the following Op-codes:

Table 5-1. OP-Codes

OP-CODE DESCRIPTION	OP-CODE DATA
Write Enable	0x06
Write Disable	0x04
Read Data	0x03
Page Program	0x02
Chip Erase	0xC7

3. Verify the connections between the TUSB926x and the SPI Flash and make sure there is activity on those lines when attempting to program the SPI Flash.

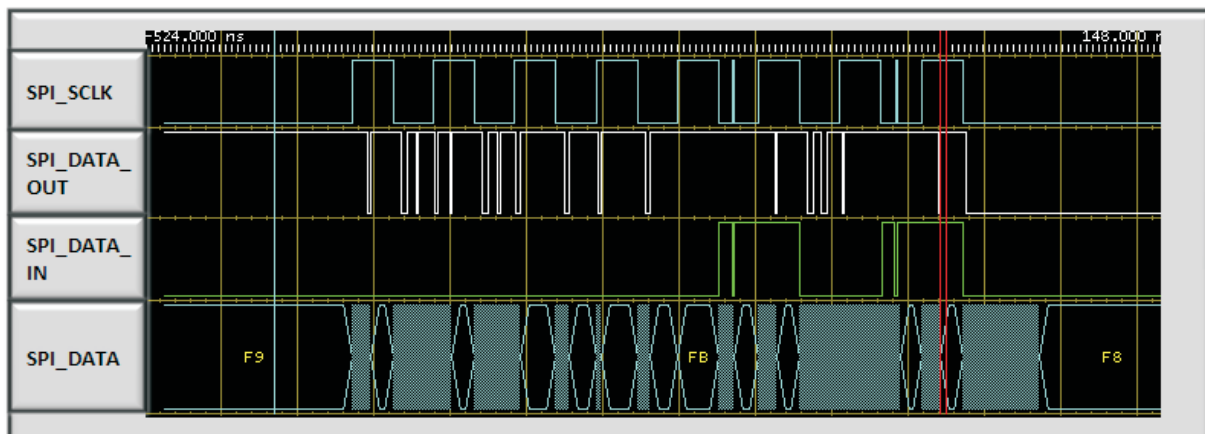


Figure 5-5. Data Transmission During Flash Programming

5.5 TUSB926x is Properly Programmed But has Yellow Alert in Device Manager

Programming multiple devices on the same system with different device descriptors, can occasionally produce Windows™ registry conflicts. When this happens, the Device Manager displays the yellow alert that the TUSB926x devices are in conflict, thus preventing the device from proper enumeration.



Figure 5-6. TUSB926x Based Device in Conflict

The TUSB926x FlashBurner GUI implements the required functionality to perform a Windows™ Registry cleanup to resolve the above mentioned conflicts. If the conflicted device is using TI's VID (0x0451) and PID (0x9260 or 0x9261), simply press the Registry Cleanup button (see Figure 5-7).

NOTE: Please unplug any TUSB926x based devices before cleaning up the registry.



Figure 5-7. Registry Cleanup Button

If programming the device with a customized VID and PID, check the Options → User Defined VID/PID Reg Cleanup menu, enter the customized VID and PID within the “Descriptors Info” group-box, and finally click the Registry Cleanup button.

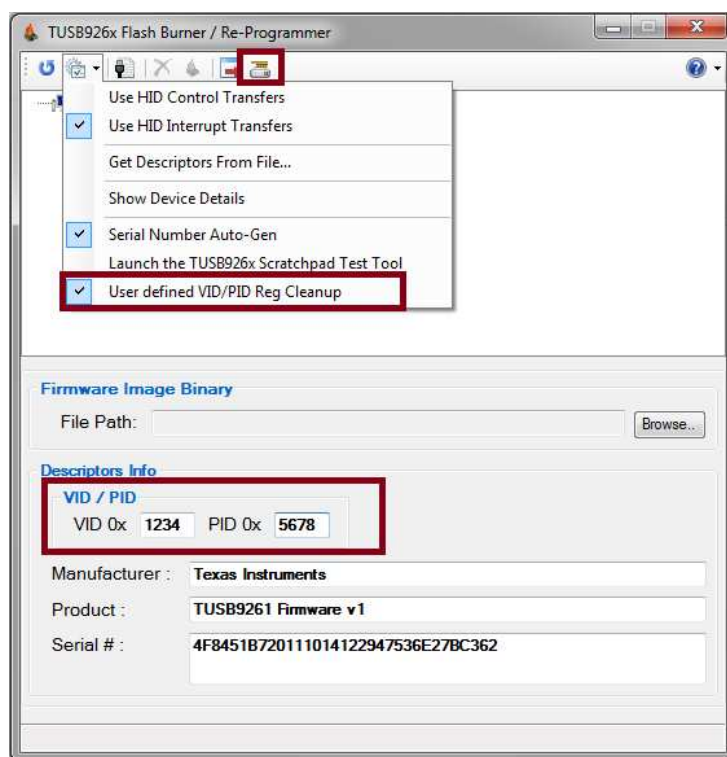


Figure 5-8. User-Defined VID/PID Cleanup

IMPORTANT NOTICE

Texas Instruments Incorporated and its subsidiaries (TI) reserve the right to make corrections, enhancements, improvements and other changes to its semiconductor products and services per JESD46, latest issue, and to discontinue any product or service per JESD48, latest issue. Buyers should obtain the latest relevant information before placing orders and should verify that such information is current and complete. All semiconductor products (also referred to herein as "components") are sold subject to TI's terms and conditions of sale supplied at the time of order acknowledgment.

TI warrants performance of its components to the specifications applicable at the time of sale, in accordance with the warranty in TI's terms and conditions of sale of semiconductor products. Testing and other quality control techniques are used to the extent TI deems necessary to support this warranty. Except where mandated by applicable law, testing of all parameters of each component is not necessarily performed.

TI assumes no liability for applications assistance or the design of Buyers' products. Buyers are responsible for their products and applications using TI components. To minimize the risks associated with Buyers' products and applications, Buyers should provide adequate design and operating safeguards.

TI does not warrant or represent that any license, either express or implied, is granted under any patent right, copyright, mask work right, or other intellectual property right relating to any combination, machine, or process in which TI components or services are used. Information published by TI regarding third-party products or services does not constitute a license 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 of the third party, or a license from TI under the patents or other intellectual property of TI.

Reproduction of significant portions of TI information in TI data books or data sheets is permissible only if reproduction is without alteration and is accompanied by all associated warranties, conditions, limitations, and notices. TI is not responsible or liable for such altered documentation. Information of third parties may be subject to additional restrictions.

Resale of TI components or services with statements different from or beyond the parameters stated by TI for that component or service voids all express and any implied warranties for the associated TI component or service and is an unfair and deceptive business practice. TI is not responsible or liable for any such statements.

Buyer acknowledges and agrees that it is solely responsible for compliance with all legal, regulatory and safety-related requirements concerning its products, and any use of TI components in its applications, notwithstanding any applications-related information or support that may be provided by TI. Buyer represents and agrees that it has all the necessary expertise to create and implement safeguards which anticipate dangerous consequences of failures, monitor failures and their consequences, lessen the likelihood of failures that might cause harm and take appropriate remedial actions. Buyer will fully indemnify TI and its representatives against any damages arising out of the use of any TI components in safety-critical applications.

In some cases, TI components may be promoted specifically to facilitate safety-related applications. With such components, TI's goal is to help enable customers to design and create their own end-product solutions that meet applicable functional safety standards and requirements. Nonetheless, such components are subject to these terms.

No TI components are authorized for use in FDA Class III (or similar life-critical medical equipment) unless authorized officers of the parties have executed a special agreement specifically governing such use.

Only those TI components which TI has specifically designated as military grade or "enhanced plastic" are designed and intended for use in military/aerospace applications or environments. Buyer acknowledges and agrees that any military or aerospace use of TI components which have **not** been so designated is solely at the Buyer's risk, and that Buyer is solely responsible for compliance with all legal and regulatory requirements in connection with such use.

TI has specifically designated certain components as meeting ISO/TS16949 requirements, mainly for automotive use. In any case of use of non-designated products, TI will not be responsible for any failure to meet ISO/TS16949.

Products

Audio	www.ti.com/audio
Amplifiers	amplifier.ti.com
Data Converters	dataconverter.ti.com
DLP® Products	www.dlp.com
DSP	dsp.ti.com
Clocks and Timers	www.ti.com/clocks
Interface	interface.ti.com
Logic	logic.ti.com
Power Mgmt	power.ti.com
Microcontrollers	microcontroller.ti.com
RFID	www.ti-rfid.com
OMAP Applications Processors	www.ti.com/omap
Wireless Connectivity	www.ti.com/wirelessconnectivity

Applications

Automotive and Transportation	www.ti.com/automotive
Communications and Telecom	www.ti.com/communications
Computers and Peripherals	www.ti.com/computers
Consumer Electronics	www.ti.com/consumer-apps
Energy and Lighting	www.ti.com/energy
Industrial	www.ti.com/industrial
Medical	www.ti.com/medical
Security	www.ti.com/security
Space, Avionics and Defense	www.ti.com/space-avionics-defense
Video and Imaging	www.ti.com/video

TI E2E Community

e2e.ti.com