

Texas Instruments TUSB8040  
EEPROM Programming Utility

# User's Guide

## Program Overview

The TUSB8040 EEPROM Programmer enables the user to program an on-board EEPROM (if present) via the Texas Instruments TUSB8040 and the Universal Serial Bus (USB).

## Requirements

Microsoft Windows XP/Vista/Server 2008/7

Microsoft DOTNET v2.0 (or greater)

A Texas Instruments TUSB8040 SuperSpeed hub with an attached EEPROM

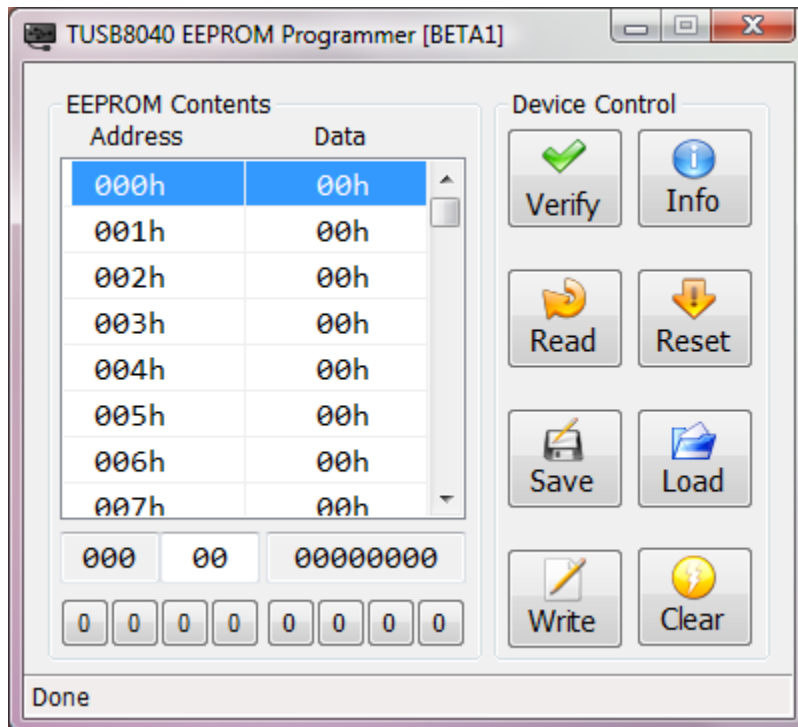
A USB 2.0 Host Controller (SuperSpeed USB is not required to use this program).

## Features

- In-Line Single-Byte Edit
- Write New EEPROM Contents from File
- Save Current EEPROM Contents to File
- Remote Disconnect
- Bulk Erase EEPROM
- Read-Back Verify of Writes
- Dynamic Device Attach/Removal Detection

## Using the Utility

To begin using this utility, simply attach the TUSB8040 SuperSpeed Hub via its upstream port to the PC host and run the program. Upon detecting an attached TUSB8040 in EEPROM programming mode, the program will automatically attach to the device and the main program window will appear:

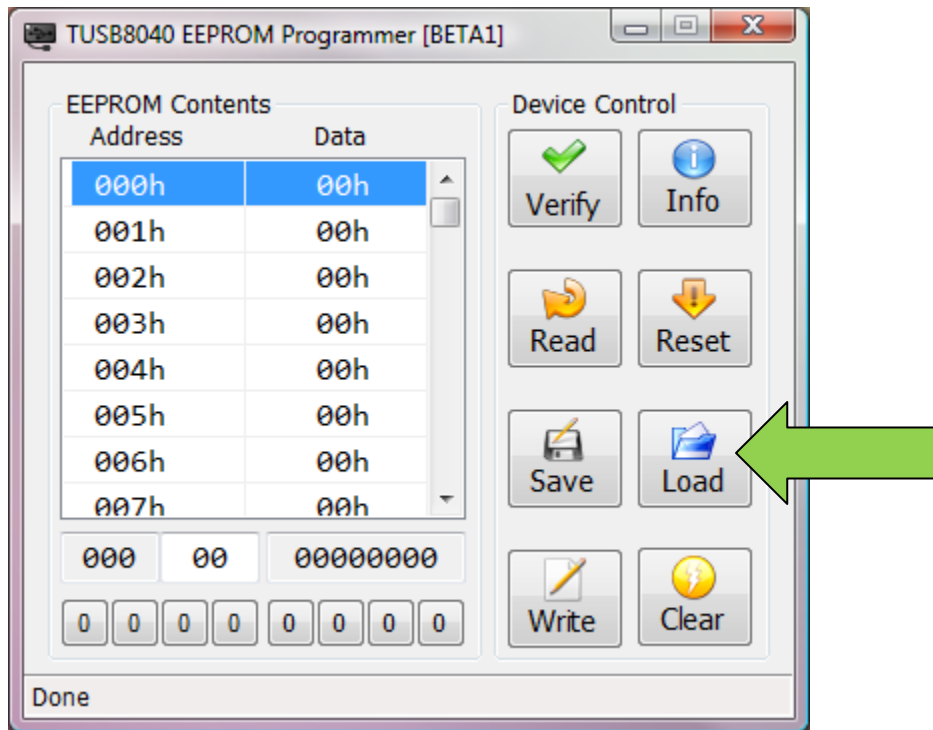


If no TUSB8040 is detected, a warning prompt will appear informing the user that no attached device was found. Please note that the TUSB8040 must be in 'programming mode' before this utility will load or the EEPROM can be accessed. There are three conditions that will enable programming mode:

- 1) A blank EEPROM is present and functional
- 2) An EEPROM is present in which address 0x00 = 0xFF
- 3) Programming mode has been externally enabled.

Please review the Troubleshooting section at the end of this document if you have a TUSB8040 in programming mode attached but it is not detected by this utility.

## Writing Data to the EEPROM from a File



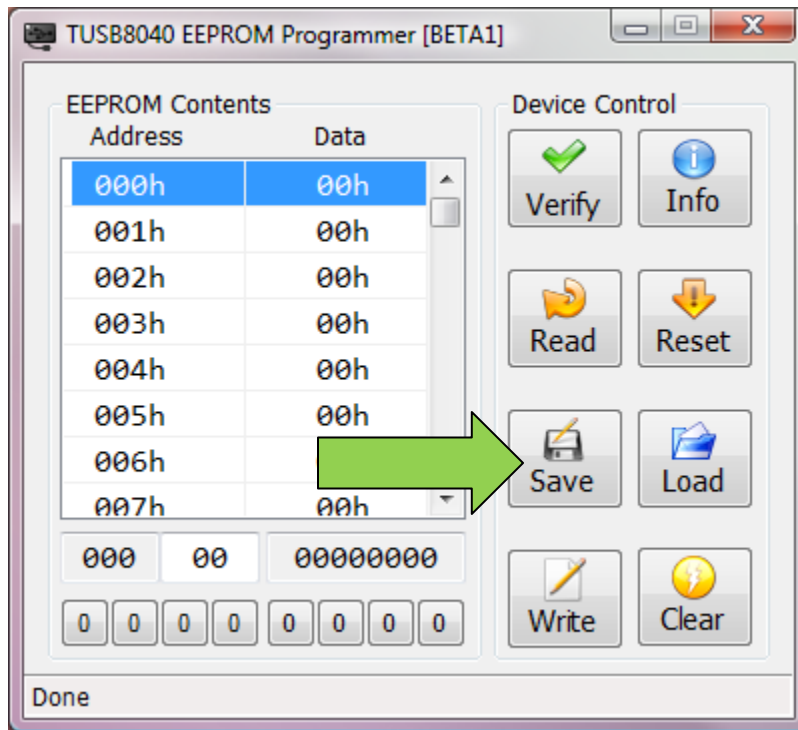
Clicking the “Load” button will prompt the user to provide a formatted data file, the contents of which will be flashed into the EEPROM. The data file is a simple tab-delimited text file that is formatted as such:

(HEX EEPROM ADDRESS) tab (HEX EEPROM DATA)

### Data File Format Notes:

- The data can be presented in any logical order.
- Parsing of the data file will stop on the first blank line.
- Lines beginning with an apostrophe ( ' ) will not be parsed.
- Both the address and the data must be represented in hexadecimal.
- No hexadecimal notation (“0x” or “h”) is required

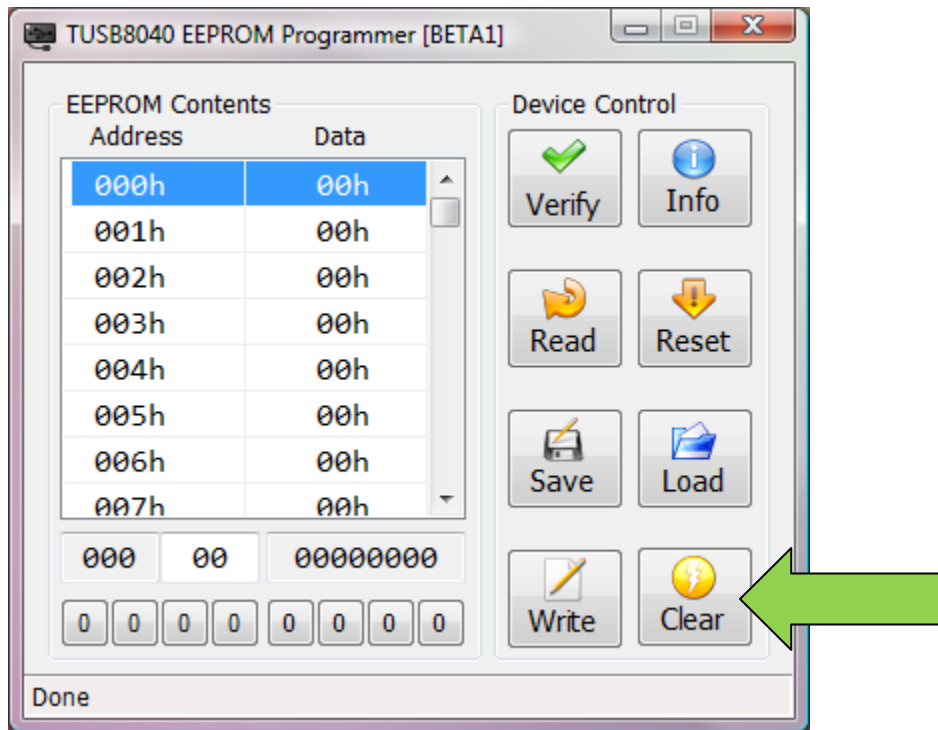
## Saving EEPROM Data to a File



Clicking the “Save” button saves the current contents of the EEPROM to a text file.

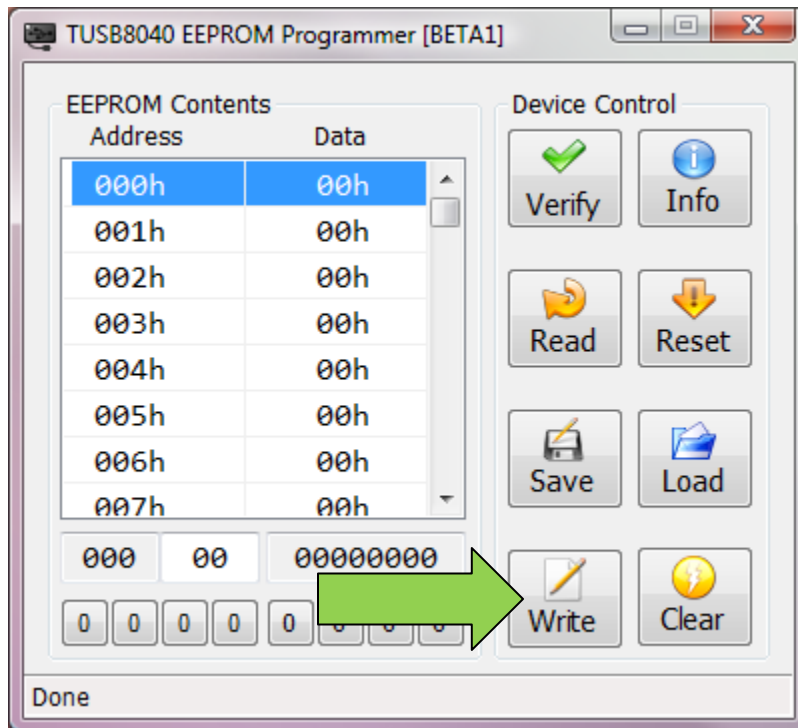
This file will be formatted correctly for use as a source file in the event that you wish to restore the values contained in the file. It can also serve as a template in the event you wish to make changes to the data contained in the file.

## Clear the EEPROM



Clicking the “Clear” button will restore the EEPROM to its default state by writing 0xFF to each address in the EEPROM.

## Write a Single Byte



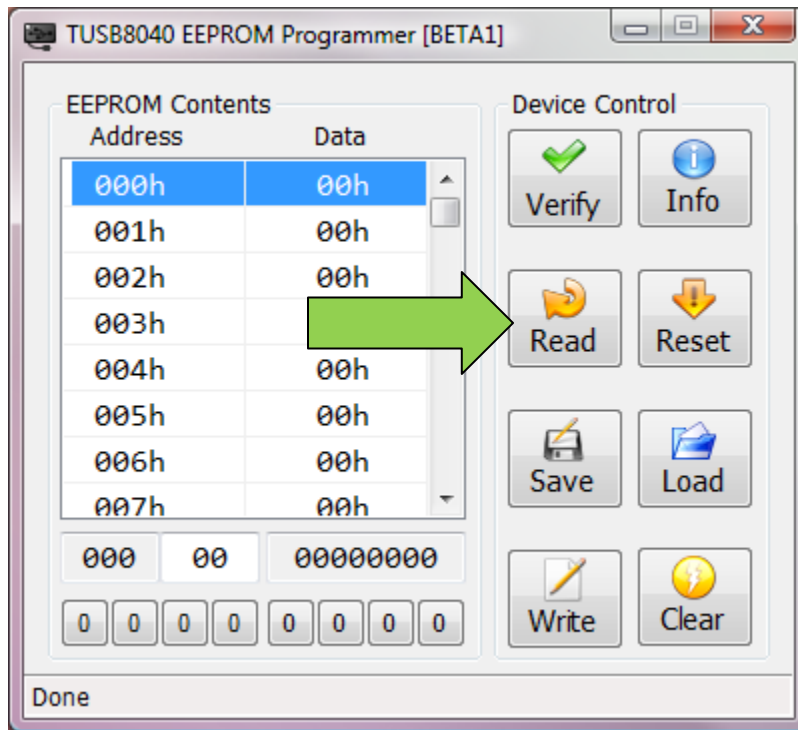
Clicking the “Write” button will write a single byte of data to the currently selected address. In the example above, 0x00 will be written to address 0x00 of the EEPROM.

To choose a different address, simply click on the address with the mouse, or scroll the “EEPROM Contents” window with the arrow-up/arrow-down keys on your keyboard.

To enter different data, click the “bits” represented by the buttons below the contents window, or enter a new hexadecimal value in the data entry field. Please note that non-hexadecimal values will not be accepted in the data entry.

**IMPORTANT NOTE!** Writing 0x55 to EEPROM address 0x00 will disable EEPROM programming mode after the next device reset. Ensure that all data in the EEPROM is correct before resetting the device.

## Re-Read the Contents of the EEPROM



Clicking the “Read” button will force a re-read of the entire EEPROM contents. This can be useful if you wish to ensure that a previously written value is still present on the EEPROM.

Please note that there are only two valid data entries for EEPROM address 0x00:

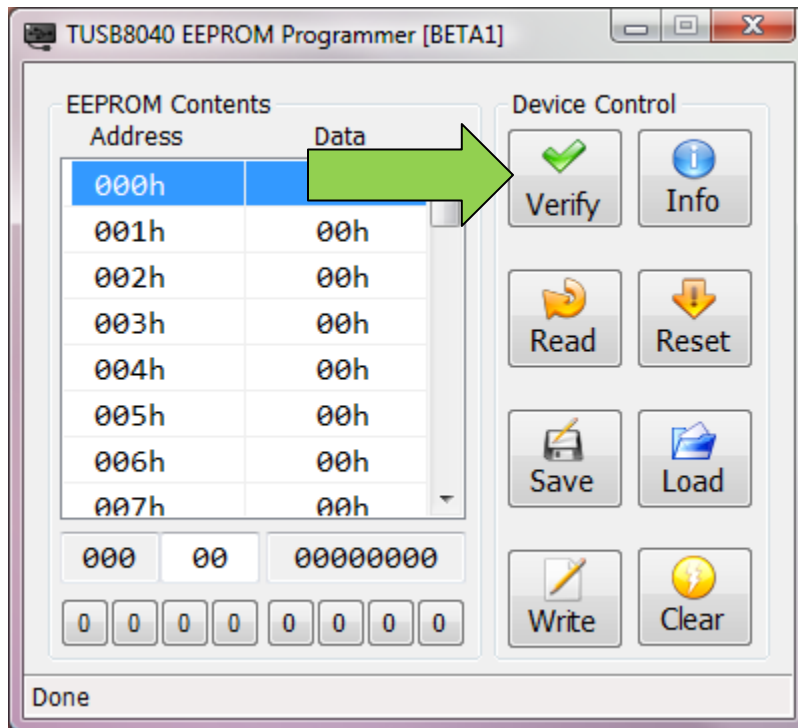
**0x55** : This EEPROM value serves as a flag that informs the TUSB8040 that the EEPROM contents are valid, and to load them into device register space.

**0xFF** : This EEPROM value serves as a flag that informs the TUSB8040 that the EEPROM is blank and to enter EEPROM programming mode. No data from this EEPROM will be loaded into the device registers.

Any other value will be ignored and the EEPROM interface will be disabled without loading any of the contents into device register space.



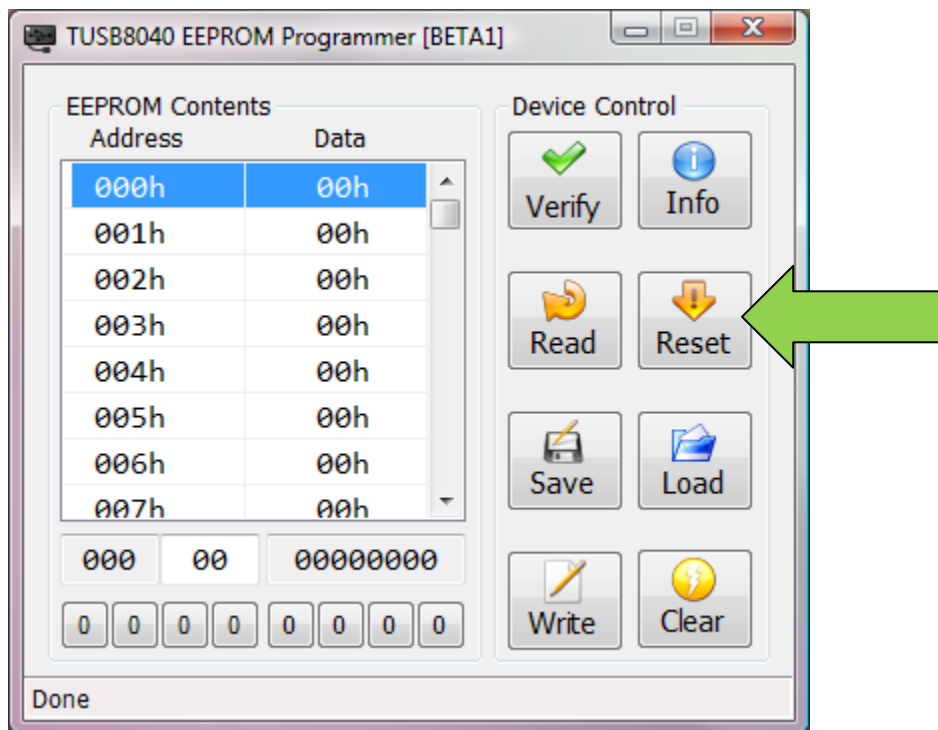
## Enable Read-Back Verification



Clicking the “Verify” button enables read-back verification of all writes. This button has a toggle function, meaning that once clicked, read-back verification of all writes will remain enabled until the button is clicked again. If any read-back -verify fails, the user will be notified via program prompt.

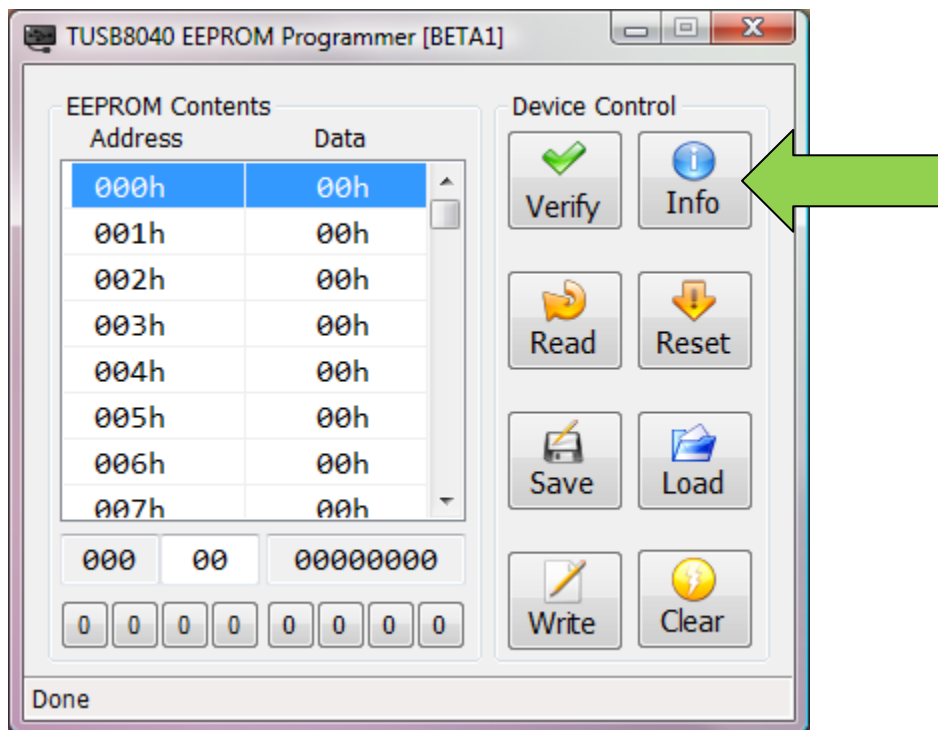
*Note: Enabling read-back verify will negatively impact write performance.*

## Resetting the Device



Clicking the “Reset” button forces the TUSB840 into a reset state for 5 seconds which will cause it to drop off the bus. After the reset timer has elapsed, the device will exit the reset state and attempt to reload the contents of the EEPROM and enumerate on the USB accordingly.

## Device Information



Clicking the “Info” button will spawn a secondary window informing the user of the current device location and instance number of the TUSB8040 on the USB.

Later versions of this utility may include other pertinent information in this window as well.

## Hot-Key Shortcuts

**ALT-V** : Toggles read-back verification

**ALT-I** : Displays the device information pane

**ALT-R** : Re-reads the entire EEPROM

**ALT-E** : Performs a device remote disconnect

**ALT-S** : Saves the current EEPROM data to a file

**ALT-L** : Write the contents of a local data file to the EEPROM

**ALT-W** : Write the currently selected byte with the data in the data entry field

**ALT-C** : Clear the EEPROM by writing 0xFF to each address

## Troubleshooting

*My TUSB8040 device is connected, but the utility does not see it.*

- Ensure the TUSB8040 is in Programming Mode
  - Check that “TI TUSB8040 EEPROM Programming Endpoint” is present in Device Manager.
  - Ensure that the device is enabled in Device Manager.
  - All downstream port LEDs should be off.
- Ensure that the TUSB8040 is powered.
- Ensure that the Host Port is enabled and functional.
- Ensure that the USB cable is not defective.

*I click the “Write” button, but the EEPROM value does not change.*

- Ensure that the EEPROM is not write-protected.
  - Enable read-back verification to ensure the EEPROM is not defective.

*The utility loads, but is “grayed out” and won’t let me do anything.*

- Ensure the TUSB8040 is still connected.
  - Check Device Manager for the presence of the “TI TUSB8040 EEPROM Programming Endpoint”

*When I try to run the program, I get a “DOTNET Framework Error”.*

- This program requires that the Microsoft DOTNET Framework (Version 2.0 or later) be installed. This can be downloaded from Microsoft: (<http://www.microsoft.com/net/>)

*I programmed 0x55 into EEPROM address 0x00 and now the TUSB8040 will no longer enter programming mode.*

- There are two options to recover an incorrectly programmed EEPROM:
  - 1) Replace the EEPROM with a new (blank) one.
  - 2) Refer to the TUSB8040 datasheet for instructions on externally entering programming mode.