Application Note

TPS2674x-Q1 Firmware Update Guide

Seong Kim

Abstract

Texas Instruments provides resources to assist manufacturers implementing the TPS2674x-Q1 device. To support design, production, and firmware updates, TI offers reference designs, software, and hardware tools for programming and testing. This application note describes the TPS2674x-Q1 firmware update process using the TI Firmware Update (TFU) protocol. This document provides necessary information and guidance for users performing TFU programming on TPS2674x-Q1 devices.

Table of Contents

1 Purpose and Scope 2
2 Application Firmware Image Bundle 2
3 TFU Programming the TPS2674x-Q1 2
3.1 Programming Over I2C1 3
3.2 Programming Over I2C4 5
4 Custom Designing a TFU Programmer 8
5 Summary 9
6 Revision History 10

List of Figures

Figure 3-1. I2C1 Connection Using the TPS26742EQ1EVM 3
Figure 3-2. TPS26742EQ1EVM Connections 4
Figure 3-3. I2C1 Connection Using the TPS26744EQ1EVM 4
Figure 3-4. TPS26744EQ1EVM Connections 5
Figure 3-5. Internal Connection to I2C4 6
Figure 3-6. I2C4 Connection Using the TI-PD-ANALYZER 6

Trademarks

Raspberry Pi® is a registered trademark of Raspberry Pi LTD.
USB Type-C® is a registered trademark of USB Implementers Forum.
All trademarks are the property of their respective owners.

Copyright © 2026 Texas Instruments Incorporated

1 Purpose and Scope

This application note details the TPS2674x-Q1 TI Firmware Update (TFU) programming process. The TPS2674x-Q1 firmware includes boot and application code stored in integrated flash memory to execute device functionality. The device receives firmware image bundles containing application firmware and configuration data from an external host over the I2C bus. The system writes these bundles into the internal flash.

The TPS2674x-Q1 application customization tool, *TPS267xx-Q1-GUI* generates TFU firmware bundles. This document provides necessary guidance for programming firmware into TPS2674x-Q1 devices. For guidance on creating TFU bundles using the *TPS267xx-Q1-GUI*, see the *TPS267xx-Q1-GUI Application Customization Tool* User's Guide.

2 Application Firmware Image Bundle

The TFU Bundle, or application firmware image bundle, serves two purposes for the TPS2674x-Q1:

1. Provides the application firmware for the internal flash
2. Defines the application configuration for the internal flash

The TPS2674x-Q1 PD controller accepts two binary file types:

Table 2-1. TPS2674x-Q1 Binary File Types

File Type	Contains	Use Case
TFU Bundle	Firmware image + configuration	Initial programming; full firmware update
AppConfig	Configuration only	Config-only updates after firmware is already flashed

When the firmware is programmed using the TFU Bundle binary, future updates that modify only configuration parameters use the AppConfig binary. This approach enables configuration setting changes without reflashing and altering the main firmware image.

The TFU Bundle and AppConfig binary programs over I2C1 and I2C4, enabling developers to customize and initialize device settings through the PD controller host interface. The bundle changes any register bit and byte accessible through the host interface before the PD controller application starts normal operation. The bundle configures system-related settings per application requirements.

3 TFU Programming the TPS2674x-Q1

Production using the TPS2674x-Q1 device requires programming an application firmware image bundle into the device integrated flash memory for proper operation. The TFU protocol defines how this bundle transfers and stores. The TFU protocol supports *Broadcast Mode*, which allows gang programming of multiple PD controllers (up to four devices). Once the firmware image bundle programs, both the application firmware and configuration data can update at any time during the device lifetime.

Two TFU methods load the firmware image bundle into the integrated flash memory:

1. Programming over I2C1: Connect an external embedded controller, such as a TM4C or Raspberry Pi® RP2040 microcontroller, to the TPS2674x-Q1 I2C1 pins for direct programming.
2. Programming over I2C4: Connect a USB Type-C® dongle such as the TI-PD-ANALYZER, which incorporates an embedded controller and requires pulldown resistors on CC1 and CC2, to the P1_GPO7 and P1_GPO8 pins (internally mapped to I2C4) while in debug accessory mode. The onboard Type-C receptacle Sideband Use (SBU) or USB2.0 pins must be routed to the P1_GPO7 and P1_GPO8 pins. See also Section 3.2, for more information on using the TI-PD-ANALYZER for programming over I2C4.

Note

On unprogrammed (factory-fresh) units, entering debug accessory mode by pulling CC1 and CC2 low is not required when loading a TFU Bundle via I2C4. However, subsequent TFU operations over I2C4 must be performed with the device in debug accessory mode.

Copyright © 2026 Texas Instruments Incorporated

3.1 Programming Over I2C1

This section outlines the required signal configurations and hardware connection examples for programming over the I2C1 interface. Configuration and generation of the AppConfig and TFU Bundle binaries, as well as programming the TPS2674x-Q1 through I2C1, can be performed using the *TPS267xx-Q1-GUI*. See also the *TPS267xx-Q1-GUI Application Customization Tool* User's Guide.

3.1.1 Configuration of I2C1

Programming the integrated flash memory through the TPS2674x-Q1 I2C1 interface requires the use of the following pins:

- 19: I2C1_SCL
- 20: I2C1_SDA
- 5: GND

The I2C requirements and CMOS logic level specifications are found in the *Electrical Characteristics* of the TPS2674x-Q1 data sheet.

The following sections describe how to use the TPS26742EQ1EVM or TPS26744EQ1EVM to program the PD controller through the I2C1 interface.

Note

After making the I2C and GND connections between the EVM and the custom board, power on the EVM before powering the custom board to avoid unintended RP2040 back-powering through the interface signals.

Note

The EVM includes onboard I2C pull-up resistors. To avoid excessive pull-up strength on the I2C bus, populate pull-up resistors on either the EVM or the custom board, but not both.

3.1.2 I2C1 Hardware Connection Using the TPS26742EQ1EVM

The TPS26742EQ1EVM can provide the required USB to I2C1 interface for programming the integrated flash memory using the *TPS267xx-Q1-GUI*. The onboard Raspberry Pi RP2040 uses a logic level of 3.3V. The TPS26742EQ1EVM connects through USB from the J14 USB Type-C receptacle to the PC. Make sure that the J16 jumpers shorting the I2C path to the onboard TPS26742E-Q1 device are removed. On the product being programmed, the relevant TPS2674x-Q1 pins must be brought out for physical contact with the EVM (such as male headers or test pads), and must be driven by no other source while programming.

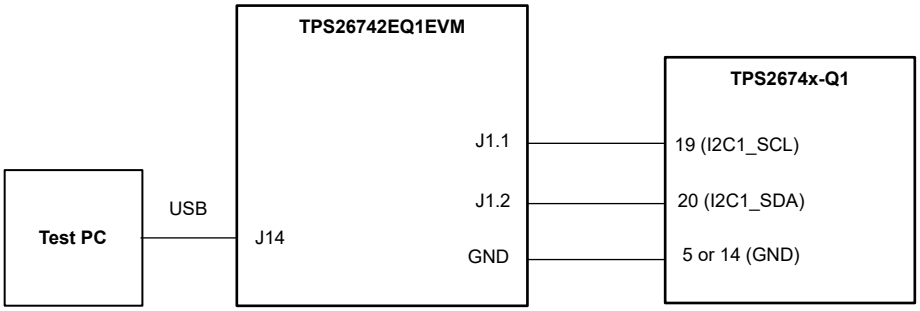


Figure 3-1. I2C1 Connection Using the TPS26742EQ1EVM

Copyright © 2026 Texas Instruments Incorporated

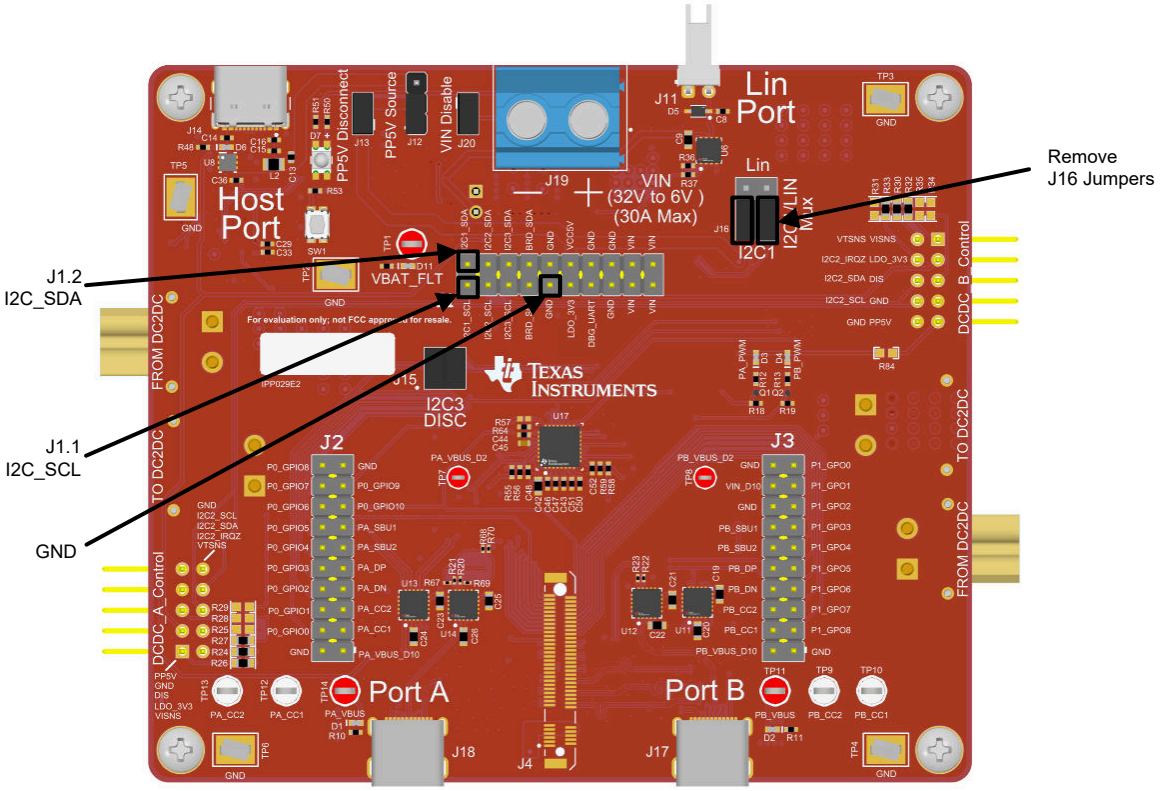


Figure 3-2. TPS26742EQ1EVM Connections

3.1.3 I2C1 Hardware Connection Using the TPS26744EQ1EVM

The TPS26744EQ1EVM can also provide the required USB to I2C1 interface for programming the integrated flash memory using the *TPS267xx-Q1-GUI*. The onboard Raspberry Pi RP2040 uses a logic level of 3.3V. The TPS26744EQ1EVM connects through USB from the J12 Type-B receptacle to the PC. Make sure that the J5 jumpers shorting the I2C path to the onboard TPS26744E-Q1 device are removed. On the product being programmed, the relevant TPS2674x-Q1 pins must be brought out for physical contact with the EVM (such as male headers or test pads), and must be driven by no other source while programming.

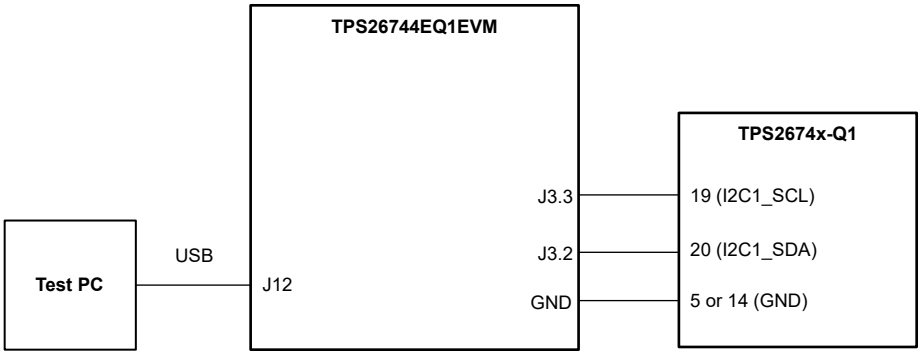


Figure 3-3. I2C1 Connection Using the TPS26744EQ1EVM

Copyright © 2026 Texas Instruments Incorporated

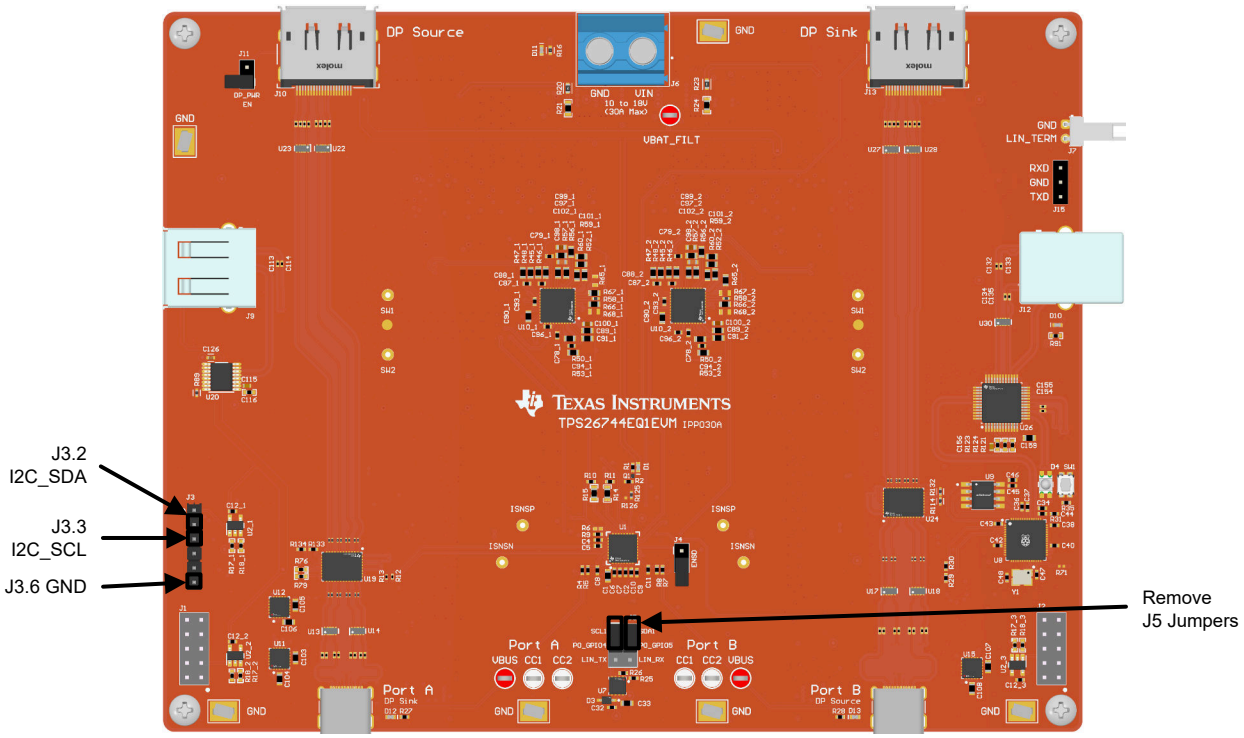


Figure 3-4. TPS26744EQ1EVM Connections

3.2 Programming Over I2C4

This section outlines the required signal configurations and hardware connection examples for programming over the I2C4 interface. After generating a TFU Bundle or AppConfig binary using the *TPS267xx-Q1-GUI*, the binary can be programmed over I2C4 using a USB Type-C dongle such as the TI-PD-ANALYZER with the TI-PD-ANALYZER-GUI. See also Section 3.2.3 for guidance on using the *TFU Update* feature of TI-PD-ANALYZER.

3.2.1 Configuration of I2C4

Programming the integrated flash memory through the TPS2674x-Q1 I2C4 interface requires the use of the following pins:

- 28: DBG_SCL/PA_DM (P1_GPO8)
- 29: DBG_SDA/PA_DP (P1_GPO7)
- 5: GND

The I2C requirements and CMOS logic level specifications are found in the *Electrical Characteristics* of the TPS2674x-Q1 data sheet.

The DBG_SCL (P1_GPO8) and DBG_SDA (P1_GPO7) pins provide access to the internal I2C4 interface. See Figure 3-5.

Copyright © 2026 Texas Instruments Incorporated

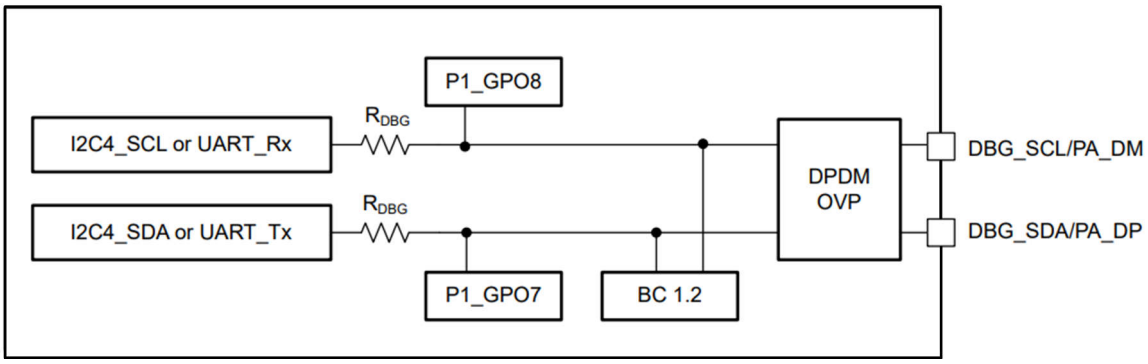


Figure 3-5. Internal Connection to I2C4

3.2.2 I2C4 Hardware Connection Using the TI-PD-ANALYZER

The TI-PD-ANALYZER provides the required I2C4 interface for programming the integrated flash memory using the TI TI-PD-ANALYZER-GUI. The dongle connects to the PC through the Micro-B receptacle, while the USB Type-C plug connects directly to the Port A Type-C receptacle of the target device (see Figure 3-6). On the target board, the TPS2674x-Q1 DBG_SDA and DBG_SCL pins must be routed to the SBU1 and SBU2 pins or D+ and D- pins. The TI-PD-ANALYZER has internal pulldown resistors on CC1 and CC2, automatically placing the TPS2674x-Q1 in debug accessory mode when connected.

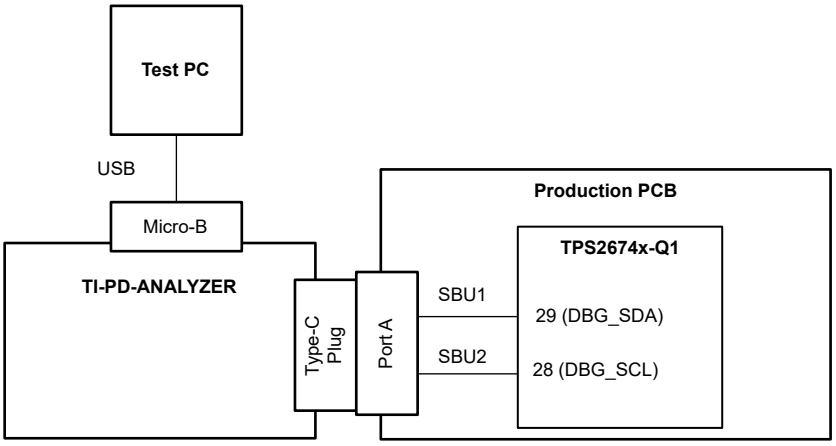
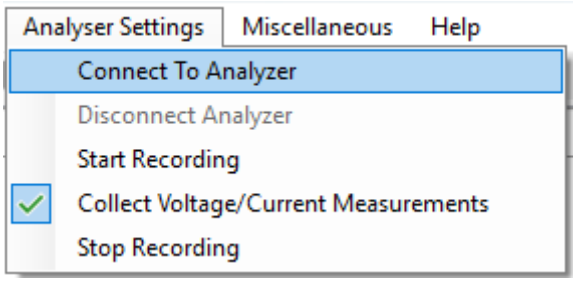


Figure 3-6. I2C4 Connection Using the TI-PD-ANALYZER

3.2.3 Using the TI-PD-ANALYZER

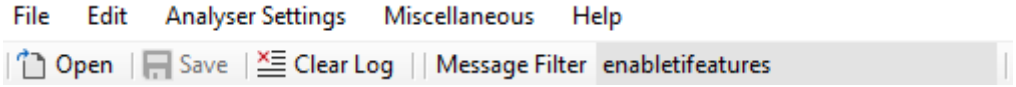
TFU programming over I2C4 can be accomplished using the TI-PD-ANALYZER with the TI-PD-ANALYZER-GUI by following these steps:

1. Connect the TI-PD-ANALYZER to a PC using a Micro-B USB cable.
2. Open the GUI.
3. Connect the GUI to the TI-PD-ANALYZER by selecting *Analyzer Settings → Connect to Analyzer.*

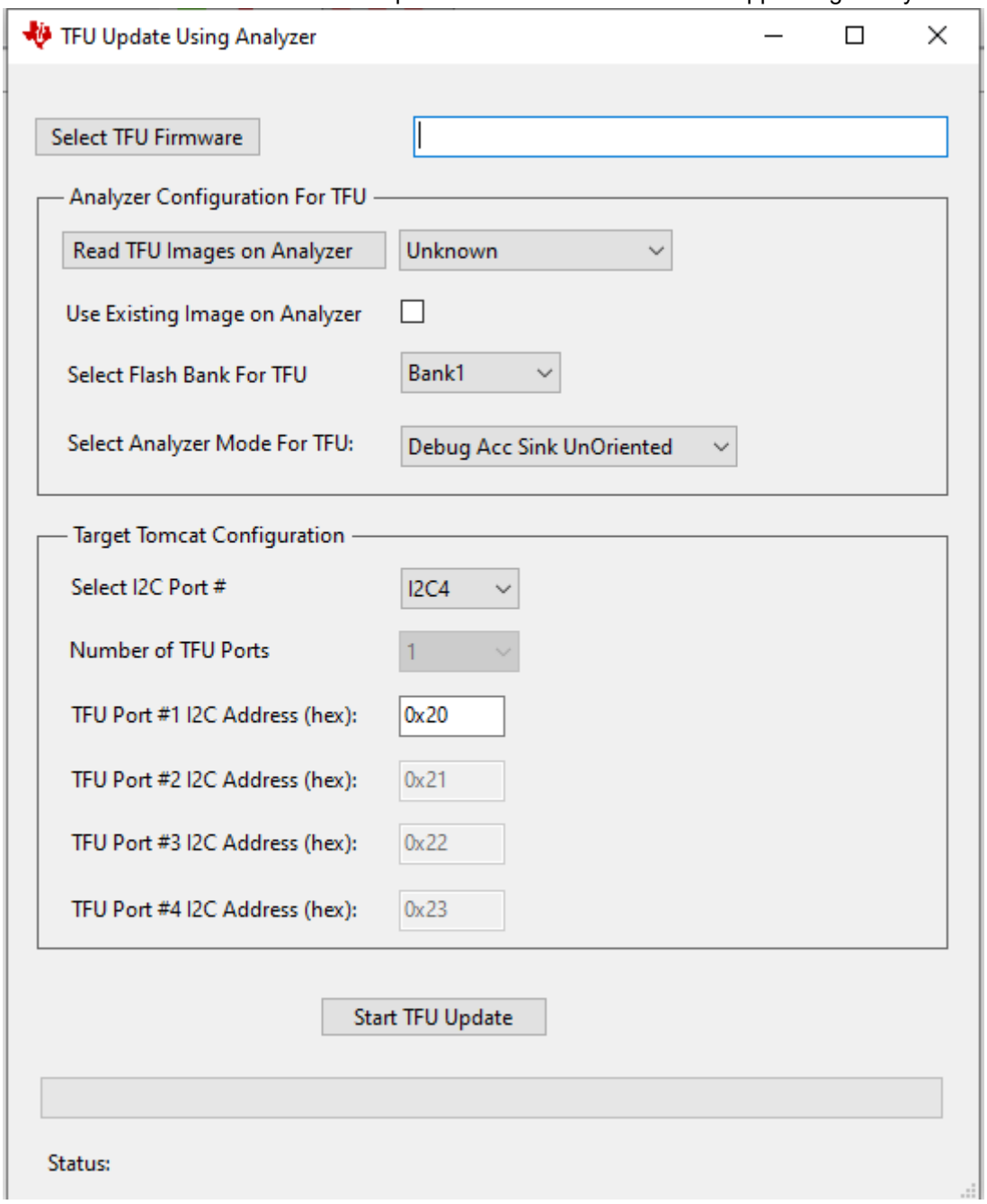


Copyright © 2026 Texas Instruments Incorporated

4. The GUI's *TFU Update* feature must be unlocked first before the TI-PD-ANALYZER can be used to program TFU Bundle or AppConfig binaries. After connecting the GUI to the TI-PD-ANALYZER, unlock the *TFU Update* feature by typing "enabletifeatures" in the *Message Filter* text box.



5. Navigate to *Miscellaneous* → *TFU Update* to open the *TFU Update Using Analyzer* menu.
6. Click the *Select TFU Firmware* button to import the desired TFU Bundle or AppConfig binary.



7. After confirming the TFU Port #1 I2C Address is correct, click *Start TFU Update* to program the TPS2674x-Q1 device.

Copyright © 2026 Texas Instruments Incorporated

4 Custom Designing a TFU Programmer

If building a custom TFU programmer design is desired for high-volume production, where using EVMs or the TI-PD-ANALYZER is impractical, request access to the TPS2674x-Q1 Design Resources content page, which includes the following materials:

1. TFU Programmer Sample Schematic
2. Sample Embedded Controller (EC) Application Code
3. *TPS2674x-Q1 TFU Programming Protocol* Application Note

The *TPS2674x-Q1 TFU Programming Protocol* Application Note details the 4CC ASCII commands and example EC host code demonstrating how to program TPS2674x-Q1 devices through I2C1 or I2C4.

Copyright © 2026 Texas Instruments Incorporated

5 Summary

The TPS2674x-Q1 supports two firmware update methods, each suited to different phases of a product lifecycle. Programming over I2C1 using an external embedded controller offers a direct, low-overhead path appropriate for initial bring-up, bench evaluation, and small-scale production setups using the TPS26742EQ1EVM or TPS26744EQ1EVM. Programming over I2C4 through the USB Type-C port using the TI-PD-ANALYZER leverages the device's debug accessory mode to eliminate additional wiring, making it convenient for field updates and in-circuit programming without dedicated test hardware. For either method, the TPS267XX-Q1-GUI application customization tool streamlines the generation and transfer of both TFU Bundle and AppConfig binaries, where the TFU Bundle handles full firmware and configuration programming while the AppConfig binary enables lightweight configuration-only updates after the application firmware is already in flash. For high-volume production environments where EVMs and the TI-PD-ANALYZER are impractical, TI provides a TFU programmer sample schematic, embedded controller application code, and the TPS2674x-Q1 TFU Programming Protocol application note to support a fully custom programmer implementation.

Copyright © 2026 Texas Instruments Incorporated

6 Revision History

Changes from Revision * (November 2025) to Revision A (November 2026)	Page
• Reformatted binary file type descriptions from bullet points to a table	2
• Added a note clarifying that factory-fresh units can be programmed through the I2C4 interface without entering Debug Accessory Mode	2
• Added EVM/custom board power-up sequencing and I2C pull-up notes	3
• Updated Figure 3-3	4
• Added explicit GUI connection procedure	6
• Added Summary section	9

Copyright © 2026 Texas Instruments Incorporated

IMPORTANT NOTICE AND DISCLAIMER

TI PROVIDES TECHNICAL AND RELIABILITY DATA (INCLUDING DATASHEETS), DESIGN RESOURCES (INCLUDING REFERENCE DESIGNS), APPLICATION OR OTHER DESIGN ADVICE, WEB TOOLS, SAFETY INFORMATION, AND OTHER RESOURCES “AS IS” AND WITH ALL FAULTS, AND DISCLAIMS ALL WARRANTIES, EXPRESS AND IMPLIED, INCLUDING WITHOUT LIMITATION ANY IMPLIED WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE OR NON-INFRINGEMENT OF THIRD PARTY INTELLECTUAL PROPERTY RIGHTS.

These resources are intended for skilled developers designing with TI products. You are solely responsible for (1) selecting the appropriate TI products for your application, (2) designing, validating and testing your application, and (3) ensuring your application meets applicable standards, and any other safety, security, regulatory or other requirements.

These resources are subject to change without notice. TI grants you permission to use these resources only for development of an application that uses the TI products described in the resource. Other reproduction and display of these resources is prohibited. No license is granted to any other TI intellectual property right or to any third party intellectual property right. TI disclaims responsibility for, and you fully indemnify TI and its representatives against any claims, damages, costs, losses, and liabilities arising out of your use of these resources.

TI’s products are provided subject to TI’s Terms of Sale, TI’s General Quality Guidelines, or other applicable terms available either on ti.com or provided in conjunction with such TI products. TI’s provision of these resources does not expand or otherwise alter TI’s applicable warranties or warranty disclaimers for TI products. Unless TI explicitly designates a product as custom or customer-specified, TI products are standard, catalog, general purpose devices.

TI objects to and rejects any additional or different terms you may propose.

Copyright © 2026, Texas Instruments Incorporated

Last updated 10/2025