This document covers the usage of the Programmable Processor PMIC's graphical user interface (GUI) tool from Texas Instruments. This GUI is intended to be used with the TPS6594x-Q1 family of devices in the context of the TPS6594x-Q1 boosterpack, the TPS6594x-Q1 Evaluation Module (EVM), or a customer hardware platform.

Contents
1 Introduction ................................................................. 2
2 Supported Features ...................................................... 2
3 Revisions .................................................................. 2
4 Overview ................................................................... 2
5 Getting Started ........................................................... 3
6 Quick-start Page .......................................................... 7
7 Register Map Page ....................................................... 15
8 NVM Configuration Page ............................................. 17
9 NVM Validation Page .................................................. 19
10 Additional Resources .................................................. 20
Appendix A ................................................................. 21

List of Figures
1 GUI Overview ............................................................. 3
2 GUI Composer Gallery ............................................... 4
3 GUI Panel within Gallery ............................................. 5
4 Device Selection Page ................................................ 5
5 Device Home Page ...................................................... 6
6 Quick-start Scan Page ............................................... 7
7 Quick-start Scan Page Results ..................................... 8
8 Quick-start Page Highlights ........................................ 9
9 System Information .................................................... 10
10 BUCK Configuration ................................................ 11
11 LDO Configuration .................................................... 12
12 GPIO Configuration .................................................. 13
13 Interrupt Mask and Status ......................................... 14
14 Miscellaneous Settings .............................................. 15
15 Register Map ........................................................... 16
16 NVM Configuration Initial View ................................. 17
17 Programming an NVM Configuration .......................... 18
18 Lock Option During NVM Programming ...................... 18
19 NVM Validation ......................................................... 19
20 Hardware Platform Error .......................................... 21
21 PMIC Communication Error ...................................... 22
22 Accessing the Scripting Window ................................. 23
23 Scripting Window Sequence Capture ......................... 24
# Introduction

This tool is created using GUI Composer and requires using the SimpleLink MSP432E401Y microcontroller, which is integrated in the EVM and also available with the MSP-EXP432P401Y launchpad. The MSP432P401Y provides a USB interface to the host personal computer (PC) for receiving commands and then communicates with the PMIC using either an I^2C or SPI protocol.

The GUI supports multiple devices with a single executable (and single MSP432E401Y), which eliminates the need to install multiple GUIs when working with more than one device. Multiple devices are configured in a master slave configuration and the MSP432E401Y communicates to each device over the selected shared medium: I^2C or SPI.

For details on how to configure the EVM for the device being used, see.

## Supported Features

The GUI supports the following features:

- Multiple PMIC devices with a single GUI
- Automatic Detection of PMICs
- Quick-start and Register views to read and write to PMIC registers (PMIC settings after power up)
- Status Indicators for Interrupts and GPIO states
- Programming and Validation of non-volatile memory (NVM)
- Multiple platforms (1): Microsoft Windows®, Linux® 32 and 64-bit, and Mac OSX
- Web based and standalone versions available
- Links to additional collateral, support forums, and FAQ

---

**NOTE:** Currently the GUI only supports I^2C communication.

## Revisions

This section details the features added with each release of the Programmable Processor PMIC’s GUI.

### 1.0

This is a pre-production initial release containing the features listed in the supported features section.

## Overview

- Device Selection
- Device Home
- Quick-start
- Register Map
- NVM Configuration
- NVM Validation

---

(1) Please refer to dev.to.com for version compatibility with GUI Composer.
As shown in Figure 1, the GUI is broken into 6 major components or pages. The Device Selection page is the first page presented when the GUI is started. Currently only one device family, the TPS659xx-Q1, is available. Once the device is selected the GUI navigates to the Device Home page. From the Device Home page, all of the device specific pages are available. The Quick-start page is recommended as the starting point for configuring the device. In addition to the device detect feature, the Quick-start page also provides a graphical view for configuring the device registers. In addition to this abstracted view is the actual register view provided from the Register Map Page. The Register Map page is a list view of the available user registers with the ability to read and write to those registers. The final two pages deal with the non-volatile memory (NVM) available on the device. The NVM Configuration page provides a means to program the default register settings. This configuration is applied automatically to the device registers at power up. The NVM Validation tool is a way to read out and verify the contents of the NVM.

5 Getting Started

Getting started involves three steps:
1. Access the GUI from ti.com.
2. Configure and Connect to the hardware platform.
3. Launch the GUI.
5.1 Accessing the GUI

The GUI is available through the TI Cloud tools found at dev.ti.com. The GUI is based upon GUI Composer and therefore recommended to use either Chrome™ (version 46+) or Firefox™ (version 38+) to access and launch the GUI. Specifically, the GUI can be found in the GUI Composer Gallery as shown in Figure 2.

Once in the gallery, navigate and locate the Programmable Processor PMIC’s GUI panel shown in Figure 3. If the panel is not visible on the main page, then sign in and use the search bar.

![Figure 2. GUI Composer Gallery](image)

5.2 Configuring and Connecting the hardware platform

The GUI tool will automatically detect the port enumerated by the micro controller, so it is recommended to configure, connect, and power the hardware platform before launching the GUI. Refer to the corresponding User's Guide for the hardware platform being used. For example, when using the TPS6594xEVM, a jumper connection on J15 connects 3.3 V (regulated from the USB 5 V) to the VCCA of the PMIC. This provides a very quick starting point, without the need for additional power supplies. Refer to the EVM User's Guide for more details about the hardware requirements.

5.3 Launching the GUI

There are several icons within the GUI panel regarding information and downloads for running a standalone GUI (for future use without access to the cloud). For this getting started section only the cloud version of the GUI will be considered. Downloading and installing a standalone version is discussed in the appendix under advanced topics. In order to use the TI Cloud version of the GUI, simply click anywhere in the panel that is not associated with the download or information icons.
Launching the GUI automatically loads the Device Selection page, shown in Figure 4. The bottom left corner indicates that the GUI has successfully connected to both the host MCU and the PMIC.

NOTE: If the GUI indicates that the controller is not responding, then attempt to change the Serial Port through the Options tab found on the top left of the GUI. If the GUI reports connection to the AEVM controller but failed to connect to the device, then please refer to the troubleshooting section, Section A.1.2.

The device variant selection will lead to the device home page shown in Figure 5. From the device Home page the Quick-start, Register, NVM Configuration, and NVM Validation Pages are available as well as the ability to return to the Device Selection page.
The Quick-start page can be entered either through the red *Quick-start* box or the Quick-start icon within the left panel.
6 Quick-start Page

From the initial Quick-start page in Figure 6, the Detect Devices box is provided to automatically detect all PMIC devices connected to the microcontroller's I²C bus.

![Figure 6. Quick-start Scan Page](image-url)
### 6.1 Device Scan Results

Figure 7 shows that for this example, two PMICs are detected. The two TPS6594x-Q1 EVMs are stacked, with one EVM configured as the master (denoted by the blue M) and the other as the slave. Additional information regarding the BUCK phase configuration and the $I^2C$ ID of each device is indicated. At this time only the $I^2C$ interface is available, and the $I^2C$ IDs represented here and throughout the rest of the document refers to the address of page 0 for each device. The device label is an editable field and can be updated to provide a custom naming convention. By clicking the Proceed button, the quick-start page will advance to device configuration.

---

**NOTE:** The hardware platform must conform to the phase configuration of one or more devices. The phase configuration is not editable from the Register or quick-start pages, but would need to be programmed through the NVM Configuration page. Any change in the phase configuration would need to be coordinated with the hardware platform.

---

![Figure 7. Quick-start Scan Page Results](image-url)
6.2 Configuration and Monitoring

Within the Quick-start page there are six horizontal tabs for editing and configuring and three vertical tabs for monitoring and accessing advanced features. These are highlighted in Figure 8. These tabs in the Quick-start page are described in the following sections. It is important to remember that each update or change in value in these sections results in a communication to the PMIC and an immediate update of the associated register or registers. The device can be reset with a power cycle to restore the register settings to the NVM and default values.

![Quick-start Page Highlights](Image)

**Figure 8. Quick-start Page Highlights**

**NOTE:** At the top of the Quick-start page is a drop down menu to select the device when multiple PMICs are connected. *Device_0* is defined in Figure 7.
6.2.1 System Info

As shown in Figure 9, the system info tab is related to the VCCA input voltage. Drop down menus are provided to show the possible configurations for each field.

Figure 9. System Information
6.2.2 BUCK

The phase configuration will determine which BUCKs are available for editing through the quick-start. In Figure 10, the BUCK2 information is not available because BUCK2 (slave) is multiphased with BUCK1 (master). Please refer to SLVSEA7 for a more detailed description of the fields and the associated operation.

![BUCK Configuration Diagram](image-url)

**Figure 10. BUCK Configuration**
6.2.3 LDO

From the LDO tab, configuration of the four LDOs is available in addition to the comparators associated with the LDOINT and LDORTC. Please refer to the SLVSEA7 for a more detailed description of the fields and the associated operation.

![Figure 11. LDO Configuration](image)

6.2.4 GPIO

The GUI provides the interface to configure the PMIC GPIOs. When the GPIO is configured as an input, the GUI provides an additional mechanism to drive the GPIO from an associated microcontroller pin and view the state change through the GPIO Status vertical tab on the right side of the page. Table 1 shows which microcontroller pins are connected to the PMIC GPIO.

<table>
<thead>
<tr>
<th>PMIC Input</th>
<th>Microcontroller Output</th>
</tr>
</thead>
<tbody>
<tr>
<td>GPIO1</td>
<td>PM0</td>
</tr>
<tr>
<td>GPIO2</td>
<td>PM7</td>
</tr>
<tr>
<td>GPIO3</td>
<td>PP5</td>
</tr>
<tr>
<td>GPIO4</td>
<td>PA7</td>
</tr>
<tr>
<td>GPIO5</td>
<td>PP3</td>
</tr>
<tr>
<td>GPIO6</td>
<td>PQ1</td>
</tr>
<tr>
<td>GPIO7</td>
<td>PH0</td>
</tr>
<tr>
<td>GPIO8</td>
<td>PH1</td>
</tr>
<tr>
<td>GPIO9</td>
<td>PK6</td>
</tr>
<tr>
<td>GPIO10</td>
<td>PK7</td>
</tr>
<tr>
<td>GPIO11</td>
<td>PP4</td>
</tr>
</tbody>
</table>
The PMIC GPIO can be configured as inputs or outputs or be mapped to internal functions within the PMIC. For this example, the Device_0 is using GPIO1 and GPIO2 for the second I²C instance. The GPIO PIN STATUS window pane can be used to confirm the function and the level of each pin when configured as GPIO. The **CONFIGURE GPIO LEVEL** box will appear when the GPIO direction is set to input in order to initially set the microcontroller output level. Please ensure that the hardware platform is configured to support the intended GPIO operation. For example, with the TPS6594x-Q1 EVM, headers are provided to easily jumper the microcontroller outputs listed in Table 1 with the PMIC inputs.

**NOTE:** When evaluating a stacked EVM solution (multiple PMICs are present), the control of the microcontroller outputs is only for the microcontroller which is connected to the GUI (connected to the PC through the USB port).

The GPIO pin status, see Figure 12, indicates the function of the GPIO as well as the current state (high or low) of GPIO which are not configured for special functionality. The colors are grey, red, and green. In addition to the color, a text description to the right of each indicator is also provided.

![Figure 12. GPIO Configuration](image-url)

Figure 12. GPIO Configuration
6.2.5 Interrupts

From the interrupts tab the user can decide to mask or monitor various interrupt sources (State Machine, GPIOs, BUCKs, and so forth).

The interrupt status, seen on the right side of Figure 13, can be used to monitor the interrupt events. The interrupts are grouped according to function and can be expanded to see each individual interrupt source. **TOP LEVEL INTERRUPTS** are read only and cannot be reset. Other interrupts can be cleared at a register or bit level, as indicated by the reset symbol. An **ERROR** status with a red dot indicates that an interrupt has occurred while a **NORMAL** status with a green dot indicates that no interrupt has occurred or that the interrupt has been cleared. Typically, grey represents interrupts which are masked. If an interrupt has a **NORMAL** status with a grey dot, then this indicates that the interrupt is not applicable for the specified phase configuration. The GUI will ignore any attempt to unmask or generate an interrupt that is not applicable to the device phase configuration.

![Figure 13. Interrupt Mask and Status](image-url)
6.2.6 Miscellaneous Settings

The MISCELLANEOUS tab includes settings for the power good (PGOOD) and spread spectrum clock features of the PMIC.

![Figure 14. Miscellaneous Settings](image)

6.2.7 Advanced

The Advanced tab provides direct write and read access to and from the registers. The access format is 0xabc, where a is the page number and bc is the register address. Refer to the device datasheet, Section 10, for the page number and register address for a given device register.

The Advanced tab also provides a scripting window, which can be used for reading and writing to the device. This window allows the user to write a script, upload scripts for reuse, or record a series of reads and writes through use of the register map. This is described in more detail in Section A.2.2.

7 Register Map Page

Once the PMIC is powered, the contents of the default registers are loaded into the registers. The Registers page provides a map to the different registers available for configuration. Unlike the Quick-start page, there is no device ID to select if multiple PMICs are present. In the Register Map page, the device address is used to distinguish which PMIC is written to or read from. The device address is the I²C1_ID, which is the I²C address for page 0 of the device. By inputting the page 0 I²C address in the Device Address input at the top of the page, the register map will automatically adjust the I²C address when reading from other I²C pages of the device.

NOTE: Figure 7 indicates two PMICs, one at address 0x48 and another at address 0x4C. At the bottom of the Registers Page are the Watchdog Registers, and as indicated, these registers are located in page 4. When the device address is specified as 0x48, then the GUI will automatically update the address to 0x4B to access page 4 of this device. Please refer to the device datasheet, Section 10, for a detailed description of the page addresses as well as a description of the PMIC registers.

---

(2) Refer to the device datasheet for information regarding which registers are loaded from NVM and which are loaded from the default registers.
The Register Map page is intended for direct read and writes to the PMIC registers. The read can be done individually or all at once. Similarly, writing to registers can also be all at once or individually. In the Immediate Write mode (option located at the top right of the page), only individual registers are written to (immediately) with each change in the Field View, change in bits, or change in hexadecimal value. In Deferred Write mode, the writing of a single register or all registers is deferred until the WRITE REGISTER or WRITE ALL REGISTERS button is pressed.

Figure 15. Register Map
8 NVM Configuration Page

The NVM configuration page is used for programming the device NVM. Once this is done, the previous NVM settings are erased and not recoverable. In order to move to the programming window, select the device on the left hand side which will place the device in the \textit{SELECTED DEVICES} pane. With a valid device in the pane, the \textit{Skip to Programming} is now allowed.

Figure 16. NVM Configuration Initial View

\textit{Figure 17} shows the PROGRAM window within the NVM Configuration page. The program to be uploaded must conform to the descriptive text format for the PMIC.

\textbf{NOTE:} Contact a TI representative to get assistance with programming settings and format.

\textbf{CAUTION}

Confirm that the I\textsuperscript{2}C address is correct before programming so that the correct device is programmed. Also make sure there is communication to both I\textsuperscript{2}C interfaces if GPIO1 and GPIO2 are configured as I2C2 or the programming will fail.
As shown in Figure 18, at the end of programming there is an option to lock the device.

The **Save Program** simply creates a copy of the file to be uploaded while **Save as a Binary** is used to translate the descriptive text file into a binary representation which can be used in the NVM Validation to confirm the NVM contents.

---

**CAUTION**

Locking a device will permanently disable any reprogramming capability.
9  NVM Validation Page

As mentioned in the previous section the NVM Validation page is used to download the current or NVM settings to the host PC or compare the settings to a file from the host PC. One of the important aspects of this page is that the NVM settings are accessed by overwriting the current settings. If the NVM Settings button is selected, as shown in Figure 19, when the DOWNLOAD CONFIGURATION button is pressed then the GUI immediately issues a command set to the PMIC to overwrite all existing register settings with the contents from NVM. This will overwrite any settings or configuration to the device made through the Quick-start or Register Map pages.

• Compare a file.
  1. Confirm that the correct device is connected.
     a. Device Family.
     b. Address (I²C Mode Only).
  2. Select the file from the host PC to compare NVM with.
     a. Use the Drag and Drop feature or the file navigator.
     b. When a valid file is selected, the COMPARE button will become active.
  3. Press the COMPARE button.

• Download the Current Settings.
  1. Confirm that the correct device is connected.
     a. Device Family.
     b. Address (I²C mode only).
  2. Select the Current Settings button
  3. Press the DOWNLOAD CONFIGURATION button.

• Download the NVM Settings.
  1. Confirm that the correct device is connected.
     a. Device Family.
     b. Address (I²C mode only).
  2. Select the NVM Settings button.
  3. Press the DOWNLOAD CONFIGURATION button.

![Figure 19. NVM Validation](image-url)
10 Additional Resources

1. TPS6594x-Q1 Power Management IC (PMIC) with 4-Phase 14-A Buck for Processors.
2. TPS6594x-Q1 Evaluation Module Users Guide.

---

**Programmable Processor PMIC's GUI User's Guide**

Copyright © 2019, Texas Instruments Incorporated
A.1 Troubleshooting

A.1.1 Hardware Platform Not Recognized

The GUI will automatically connect to the hardware platform (micro controller with the analog evm controller firmware) and then to the PMIC. Typically, if the GUI cannot find the hardware platform this is due to either a faulty USB connection or the GUI is attempting to connect to the wrong communication (COMM) port. The hardware platform will enumerate as three devices; two CDC classes and one DFU class. From another program, like the device manager in windows, the user can verify which COM port is the ACCtrl and which is the ACCtrl Console. From the GUI option, the user should select the COM port number of the ACCtrl and **not** the ACCtrl Console. The micro controller supports a baud rate of 115200.
A.1.2 PMIC Device not found

If the hardware platform is connected but the PMIC device is not found, then the GUI will report an error regarding the address: *Connected to AEVM Controller, but failed to connect to device TPS6594x_external on I2C @xx*. In the case of I2C this means that the address was not acknowledged (NACKed). Perform appropriate I2C bus checks: appropriate pull-up resistors, verify that no device is holding the clock low (clock-stretching), and so forth. Verify that the correct I2C address is being sent. The default address which the GUI uses may not be correct and it is necessary to move to the Register or NVM Validation Pages to update the address being used.

A.2 Advanced Topics

A.2.1 Downloading and Installing a standalone GUI

The GUI can be imported to GUI composer or downloaded as a standalone GUI on the host PC. Click the speedometer icon to import into GUI composer or click the download icon to download a standalone executable. The user will be prompted to download the appropriate run time engine depending upon the platform being used.

A.2.2 Scripting Window

As shown in *Figure 22*, the **ADVANCED** icon within the quick-start page provides access to the scripting window. Scripting is a convenient way to send a sequence of commands (reads or writes) to the PMIC device registers as opposed to the individual commands associated with an update to a parameter in either the quick-start or Register page views. Opening the scripting window will open a new window while the GUI window will still be active as shown in the following paragraph.
Figure 22. Accessing the Scripting Window

Figure 23 shows the initial scripting window and the default text provided. This file can be edited to provide the desired sequence of commands. One quick way to start using the scripting window is to use the record feature to capture a sequence of commands. In the upper right hand corner is the record icon. Hovering the cursor over the icon with reveal a Start Recording help box. In this example, the recording is started and then when returning to the GUI window the Register Page is used to read, write, and then read again from the GPIO1 configuration register. Once these sequence of actions are completed, then returning to the scripting window will reveal the recorded commands. At this point the recording can be stopped and these commands can be expanded and repeated for the various registers. Once the sequence is complete, then click the Run icon to execute the sequence.
Figure 23. Scripting Window Sequence Capture
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, 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 will 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 (www.ti.com/legal/termsofsale.html) 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.

Mailing Address: Texas Instruments, Post Office Box 655303, Dallas, Texas 75265
Copyright © 2020, Texas Instruments Incorporated