Application Report

TPS65987D GPIO Events

Wang, Hao

ABSTRACT

The TPS65987D device from the Texas Instruments family of USB Type-C® and USB PD controllers provides a set of GPIO events to achieve desired system behavior. A developer may program custom behavior triggered by GPIO to enable new functionality, and even load modified device configurations using GPIO events functionality.

These firmware-based GPIO events are simple to configure using the provided GUI software tools. The core TI PD controller firmware is unchanged when using GPIO events which ensures reliability, USB PD compliance, and also eases and speeds up development. This application report describes the procedure for configuring GPIO events on the TPS65987D and provides some concrete examples.

Table of Contents

List of Figures

List of Tables

Trademarks

USB Type-C® is a registered trademark of USB Implementer's Forum.
All trademarks are the property of their respective owners.

1 Introduction

GPIO Events feature of TPS65987D allow users to tie specific events within the PD controller to trigger a signal in the system and also control the PD controller behavior by an external signal. These GPIO toggles in response to a defined PD or USB event can be used for customizing system behavior. TPS65987D Configuration Tool is used to assign events to specific GPIO. TPS65987D device has a number of configurable GPIOs that can be used for this purpose and each GPIO behavior can be configured independently with such events depending on the system need.

The ability to configure independent GPIO events allows PD system designers to achieve variety of system behavior. This helps TPS65987D users to implement unique applications and differentiate their end products with innovative system implementations. There are also GPIO events available that can trigger loading a modified device configuration settings real-time based on the requirements of an application that require configuration change on-the-fly.

Unlike some PD controllers in the market that require firmware customization, TI PD controller can deliver the same custom behavior using GPIO events feature keeping the core firmware same. This ensures that a fully tested and verified firmware can be used by all end users without having to modify PD controller internal firmware. This helps speed up end product development cycle and ensures that overall system behavior is robust and reliable.

The following topics are covered in this chapter:

1. Available GPIO events in TPS65987D.
2. Setting up and modifying examples of GPIO events capabilities.
3. Verifying correct behavior of event that is configured.
4. Analyzing the results of the PD trace data.
5. Modifying GPIO events by using TPS6598x Application Customization tool.

2 TPS65987D GPIO Event List

TPS65987D firmware implements specific events that can be tied to GPIOs. These assigned events dictate the behavior of a system in response to a defined hardware or USB event. The TPS6598x Application Customization tool can be used to assign events to specific GPIOs. Table 2-1 lists all the GPIO events that are available in TPS65987D and their behavior.

Table 2-1. List of TPS65987D GPIO Events

Event Name	I/O	Active State	Behavior
Port 0 Plug Event	Output	High	• Asserted high when plug event (attached state) has occurred. • Asserted low when disconnected.
Port 0 Cable Orientation Event	Output	High	• Asserted high when plug is connected upside-down. • Asserted low when the plug is connected upside-up or disconnected.
Port 0 AMSEL Event	Output	N/A(Tri-state)	• Asserted high when DisplayPort alternate mode is entered and pin assignment A/C/E. • Asserted low when DisplayPort alternate mode is entered and pin assignment B/D/F. • High-Z when DisplayPort alternate mode is not entered.
Port 0 Source PDO 0 Negotiated Port 0 Source PDO 1 Negotiated Port 0 Source PDO 2 Negotiated Port 0 Source PDO 3 Negotiated	Output	High	Asserted high when source PDO x has been negotiated, otherwise low.
Port 0 Source PDO Negotiated TT 1 Port 0 Source PDO Negotiated TT 2 Port 0 Source PDO Negotiated TT 3	Input	High	These 3 Events combine to form a 3-bit truth table to allow digital outputs indicating the active state of up to 7 PDOs. TT 3 is the most-significant bit (MSB) and TT 1 is the least significant bit (LSB).
Output Enabled Without Event	Output	N/A	Acting as an output without any event.

Table 2-1. List of TPS65987D GPIO Events (continued)

Event Name	I/O	Active State	Behavior
Port 0 PDIO In 0 Event Port 0 PDIO In 1 Event Port 0 PDIO In 2 Event Port 0 PDIO In 3 Event	Input	N/A	Input GPIO event for PDIO Alternate Mode (when supported by both port partners and mode is entered). A change in state of PDIO In x will trigger a PDIO Alternate Mode message to be sent to the port partner. PDIO Out x will reflect the value of this signal after the PDIO Alternate Mode message is received by the port partner. These events do not have a pre-determined active state.
Port 0 PDIO Out 0 Event Port 0 PDIO Out 1 Event Port 0 PDIO Out 2 Event Port 0 PDIO Out 3 Event	Output	N/A	Output GPIO event for PDIO Alternate Mode. When PDIO Alternate Mode is supported by both port partners and entered, output follows GPIO pin mapped to PDIO In x event on port partner.
Port 0 USB3 Event	Output	High Z	High-Z when data connection is USB3 on Port 0, low in all other cases.
Port 0 DP Mode Select Event	Output	High	• Asserted high when data connection is DisplayPort (either 4-Lane mode or 2- Lane+USB3 mode). • Asserted low when Type-C port is disconnected or DisplayPort mode is not active.
Port 0 User SVID Active Event	Output	High	Asserted high when port is in User SVID Alternate Mode, otherwise low.
Port 0 Source Sink Event	Output	N/A(Tri-state)	• Asserted high when port is operating as a Source. • Asserted low when port is operating as a Sink.
Port 0 DP or USB3 Event	Output	High	• Asserted high when data connection is DisplayPort or USB3. • Asserted low if neither data mode is active or port is disconnected.
Port 0 UFP DFP Event	Output	High	• Asserted high when port is operating as UFP. • Asserted low when port is operating as DFP.
Port 0 TBT Event	Output	High	Asserted high when data connection is thunderbolt otherwise low.
Port 0 Billboard Event	Output	High	Asserted high when Billboard is presented, otherwise low.
Port 0 Fault Input Event	Input	Low	Used to allow external devices to enable error recovery on a given port. There is one fault condition input per port. When set low port enters Error Recovery State. When set high, no action.
Port 0 FRSwap Input Event	Input	N/A	On the falling edge of input event, a device configured as a Source will enable the FRS pulldown on the CC pin and start the FRS process. No action on rising edge of input event.
Port 0 Fault_Condition_Active_Low_ Event	Output	Low	Asserts low on an overcurrent event.
Port 0 Load App Config 1 Event Port 0 Load App Config 2 Event Port 0 Load App Config 3 Event	Input	N/A	Upon Rising Edge: • App Config Set for GPIO = High will be loaded as the active configuration. • 1st 4CC Data and Command is written to selected CMDX register (optional). • 2nd 4CC Data and Command (or PD Task) is written to selected CMDX register (optional).
			Upon Falling Edge: • App Config Set for GPIO = Low will be loaded as the active configuration. • 1st 4CC Data and Command is written to selected CMDX register (optional). • 2nd 4CC Data and Command (or PD Task) is written to selected CMDX register (optional).
Port 0 Sink Greater Than Threshold Event	Output	High	• Asserted high when in an active PD contract and Sinking less than threshold setting. • Asserted low when any other sink or source PD contract is active, no PD contract is active, or port is disconnected.

Table 2-1. List of TPS65987D GPIO Events (continued)

Event Name	I/O	Active State	Behavior
Port 0 Retimer_PWR_EN_Event	Output	High	Asserted high when USB Type-C connection is present, or when the "Retimer_SoC_Force_PWR_Event" is asserted high. Otherwise, this event is asserted low.
Port 0 Retimer_Reset_N_Event	Output	High	Asserted high when a USB Type-C connection is present and asserted low when there is no USB Type-C connection present. Upon a USB Type-C connection, first the "Retimer_PWR_EN_Event_Portx" event will occur, and then this event will occur tRetimerForcePowerDelay later. When a USB Type-C connection is removed, first this event will go low, and then "Retimer_PWR_EN_Event_Portx" event will happen tRetimerForcePowerDelay later. tRetimerForcePowerDelay is set in the 0x43 Delay Configuration Register.
Port 0 Prochot N Event	Output	High	A signal to the main SOC to notify it of any changes in power capabilities. When this event is asserted, typically the main SOC will reduce its power consumption until it has re-evaluated the new power capabilities of the system. This event is asserted high when the device enters Unattached.SRC, Unattached.SNK, when sending a Request message, sending an Accept message to a PR_SWAP request, or when a PD3.0 Fast Role Swap occurs. This event is asserted low when the ProcHot interrupt in the IntEventX register (0x14 for port 1, 0x15 for port 2) is cleared.
Retimer SOC OVR Force Power Event	Input	High	When this input is asserted (high), the PD controller (through the Retimer_PWR_EN_Event_Portx GPIO event) will instruct the external retimers to power on always, even when no USB Type-C connection is present. When this input is deasserted (low), the PD controller will only instruct the external retimers to power on when a USB Type-C connection is present.
Barrel Jack Detect Event	Input	High	Upon Rising Edge (Barrel Jack detected): • Clear Dead Battery Flag • Set Externally Powered = 1 • Swap to Source
			Upon Falling Edge (Barrel Jack removed): • Set Externally Powered = 0 • Swap to Sink
UFP Indicator Event	Output	High	Asserted high when at least one port has a data role of UFP, otherwise low.
Prevent DR Swap to UFP_Event	Input	High	When high DR_Swap requests that would result in the target port changing to the UFP role will be rejected.
High Current Contract Active Event	Output	High	Asserted high when at least one port has negotiated a source contract exceeding 5 V at 0.9 A, otherwise low.
Prevent High Current Contract Active Event	Input	High	When high source capabilities are reduced to only 5 V at 0.9 A.
Port 0 Audio Mode Event	Output	High	This event is asserted when an Audio Accessory (Ra/Ra) is attached.
Port 0 Source Power Greater Than Threshold	Output	High	• This event is asserted high when the USB Type-C implicit contract of the explicit USB PD contract currently negotiated is allowing the sourcing of power greater than the threshold value programmed in the PowerThresAsSourceContract Byte 7 in the Port Configuration Register (0x28). • Asserted low when the currently negotiated contract is less than the programmed threshold.
Port 0 Debug Accessory Event	Output	High	Asserted high when debug accessory mode is detected, otherwise low.
Port 0 Sink PDO 0 Negotiated Port 0 Sink PDO 1 Negotiated Port 0 Sink PDO 2 Negotiated Port 0 Sink PDO 3 Negotiated	Output	High	This event is asserted when the TXSinkPDO1 from the TX Sink Capabilities Register (0x33) has been negotiated. Otherwise, this event is deasserted.

Table 2-1. List of TPS65987D GPIO Events (continued)

Event Name	I/O	Active State	Behavior
Port 0 Sink PDO Negotiated TT 1 Port 0 Sink PDO Negotiated TT 2 Port 0 Sink PDO Negotiated TT 3	Input	High	These 3 Events combine to form a 3-bit truth table to allow digital outputs indicating the active state of up to 7 PDOs. TT 3 is the most-significant bit (MSB) and TT 1 is the least significant bit (LSB).
Port 0 Vconn On	Output	High	Asserted high when PP_CABLE1 is enabled to source VCONN.
Disabled	N/A	N/A	GPIO is disabled.
Port 0 PR_Swap_Ext_Vbus_Dsch	Output	Low	This event causes the mapped GPIO to be pulled low after a PR_Swap to enable an external VBUS discharge circuit during a power-role swap on port 0
PP1 Switch Event	Output	High	• Asserted high when PP1 switch is closed. • Asserted low when PP1 switch opens.
Port 0 I2C1 Master IRQ Event	Input	High	When this input is asserted, it generates an interrupt to the I2C1 master so it can properly respond to an external retimer.
Port 0 I2C3 Master IRQ Event	Input	High	When this input is asserted, it generates an interrupt to the I2C3 master so it can properly respond to an external retimer.
Port 0 USB2 on HS MUX Event	Output	High	• Asserted high when USB2 is active, otherwise low.
Port 0 BC1.2_Host_Pull_Down_Enable_Event	Output	Low	This event is set low when BC1.2 ChargerAdvertiseEnable bits are set to one of the DCP modes in the Port Control Register (0x29) to disable the USB2.0 Host Pulldowns (Hi-Z them), if the USB Host needs an external signal to disable its pulldowns, so the BC1.2 DCP modes can function properly. This event is also asserted low when there is no USB Type-C connection. Otherwise, it is asserted high.
Sink Arbitration GPIO Output	Output	High	Works in conjunction with Sink_Arbitration_Input to ensure only one sinking path in the system is turned on.
Sink Arbitration GPIO Input	Input	High	On a falling edge of this GPIO, the PD controller will evaluate the policy engine state and context for each port. If appropriate, the PD controller will enable the sink paths for one or both ports. Before enabling the sink paths, the Sink_Arbitration_Output will be driven high, and the PD controller will wait for the MultiPortSinkNonOverlapTime which is set in the Global System Configuration register (0x27). On a rising edge of this GPIO, the PD controller disables the sink paths for the ports that are connected to a USB PD source The PD controller also drives the Sink_Arbitration_Output low.

3 GPIO Events Register and Example Settings

Configuration Registers

- 0x5C, GPIO Configuration

GPIO configuration registers of TPS65987D allows event mapping to available GPIOs. Each GPIO output can be configured as open drain or push-pull, and use either LDO_3V3 or VDDIO as the supply. Internal pullup and pulldown resistors for each GPIO can also be configured using configuration register. Note that some of the GPIOs that are pre-configured in the firmware for specific event can't be changed using the Application Customizer tool.

3.1 GPIO Event Example Settings

The TPS6598x Application Customization tool can be used to set different GPIO Event Capabilities. Using I/O Configuration page of the tool, any event can be assigned to a GPIO as shown in Figure 3-1.

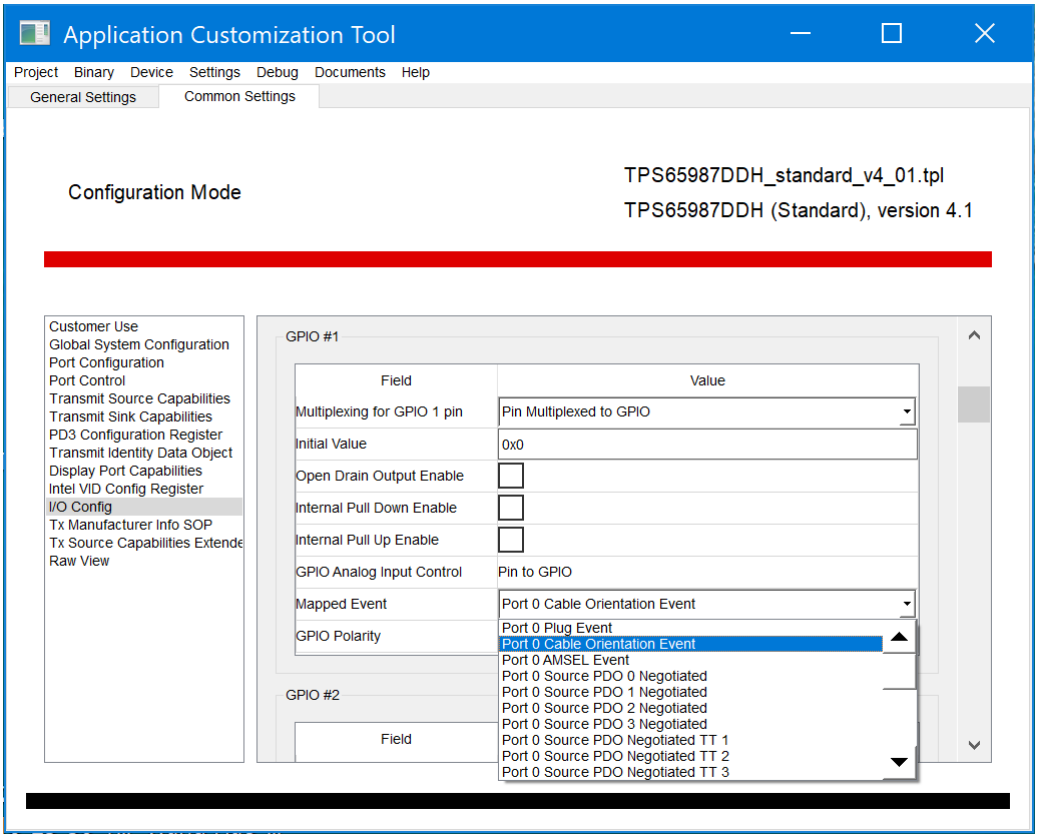

Figure 3-1. Mapping a GPIO Event using TPS6598x Application Customization Tool

The TPS6598x Application Customization tool also contains example projects with different GPIO Event Capabilities already mapped depending on system need. The project template named "*TPS65987DDH_standard_v4_01.tpl*" demonstrates an example of how the GPIO Events are mapped for TPS65987D EVM. Once the project template is loaded all the relevant GPIO Events that are configured can be seen from "*I/O Configuration*" page of the tool as shown in Figure 3-2.

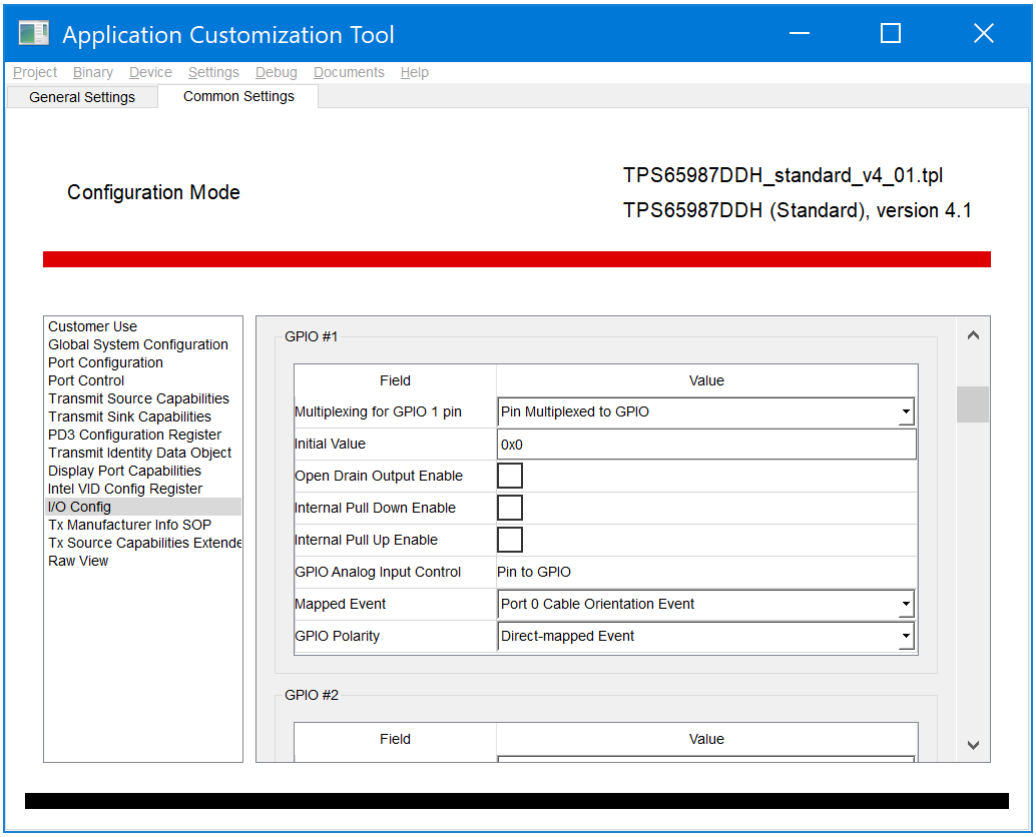

Figure 3-2. Template with GPIO Events Mapped for TPS65987D EVM

3.2 Application Configuration GPIO Event Settings

There are advanced GPIO events that can be used to load modified configurations to device at run-time. The example shows that TPS65987D transmit new source capabilities 5 V, 3 A when GPIO4 goes from high to low and 5 V, 3 A and 9 V minimum 20 V maximum, 3 A when GPIO4 goes from low to high. Here are the steps to set up *I/O Configuration*:

1. Load a template. The new project template *TPS65987DDH_advanced_v4_01.tpl* can be loaded by clicking *Project→TPS65987DDH→Advanced→Dual Role Port (DRP), prefers power source→None (DisplayPort Only)*.
2. Change *Number of Configuration Sets* in *General Settings* to 2, and then change *Virtual Device 1* to *AC GPIO Low*, and *Virtual Device 2* to *AC GPIO High* as shown in Figure 3-3.

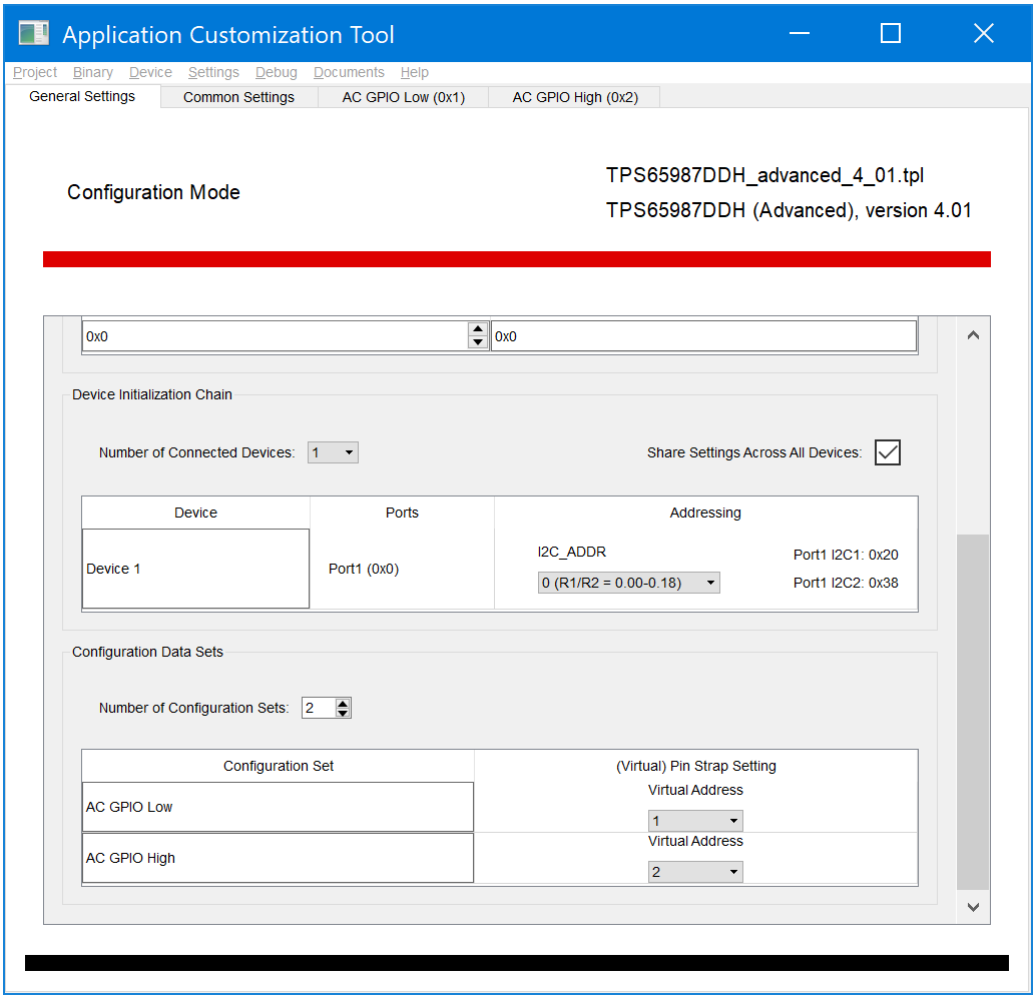

Figure 3-3. Number of Configuration Sets

3. Set App Configuration Group 1 in *Common Settings*.

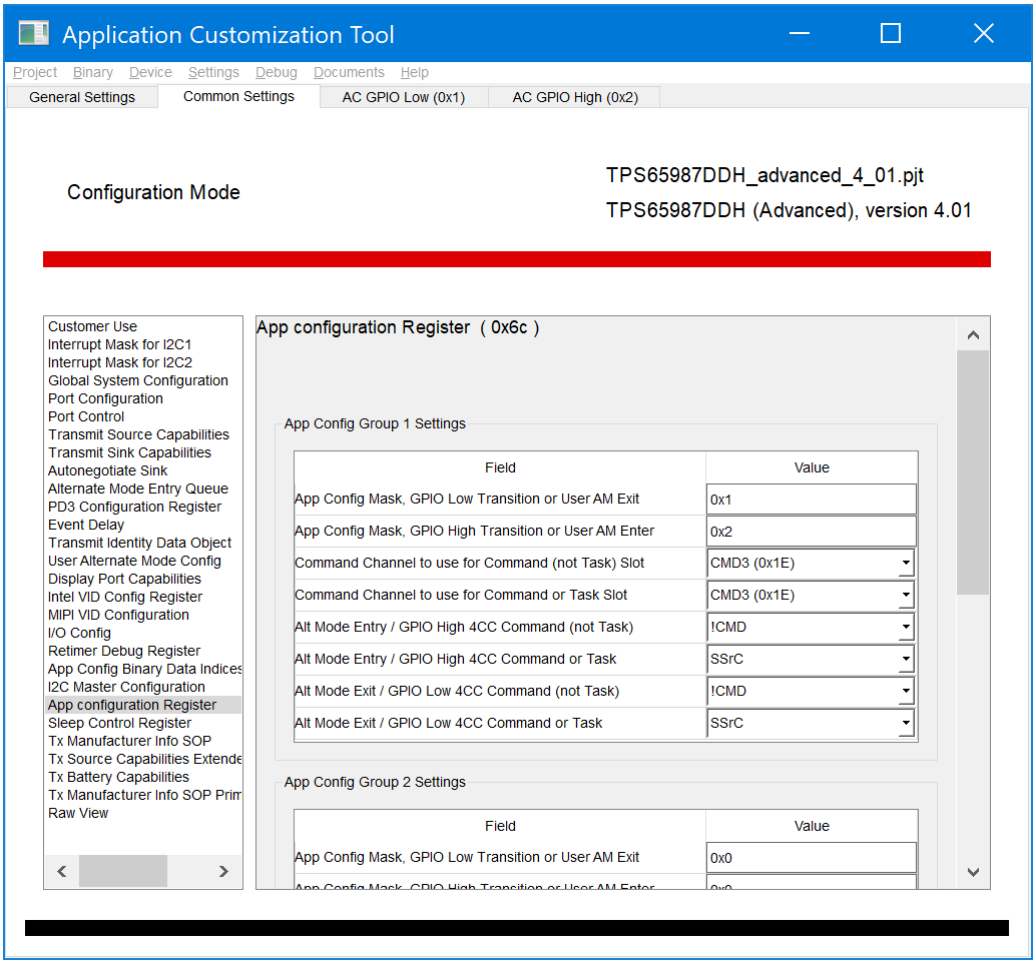

Figure 3-4. App Config Group 1 Settings

4. Adjust Registers in *AC GPIO Low (0x1)* to set the behavior when GPIO4 goes from high to low. External hardware event can trigger the PD controller to change configuration. In this example, GPIO4 high to low transition would configure Transmit Source Capabilities resister (0x32) with one PDO as shown in Figure 3-5 and Figure 3-6.

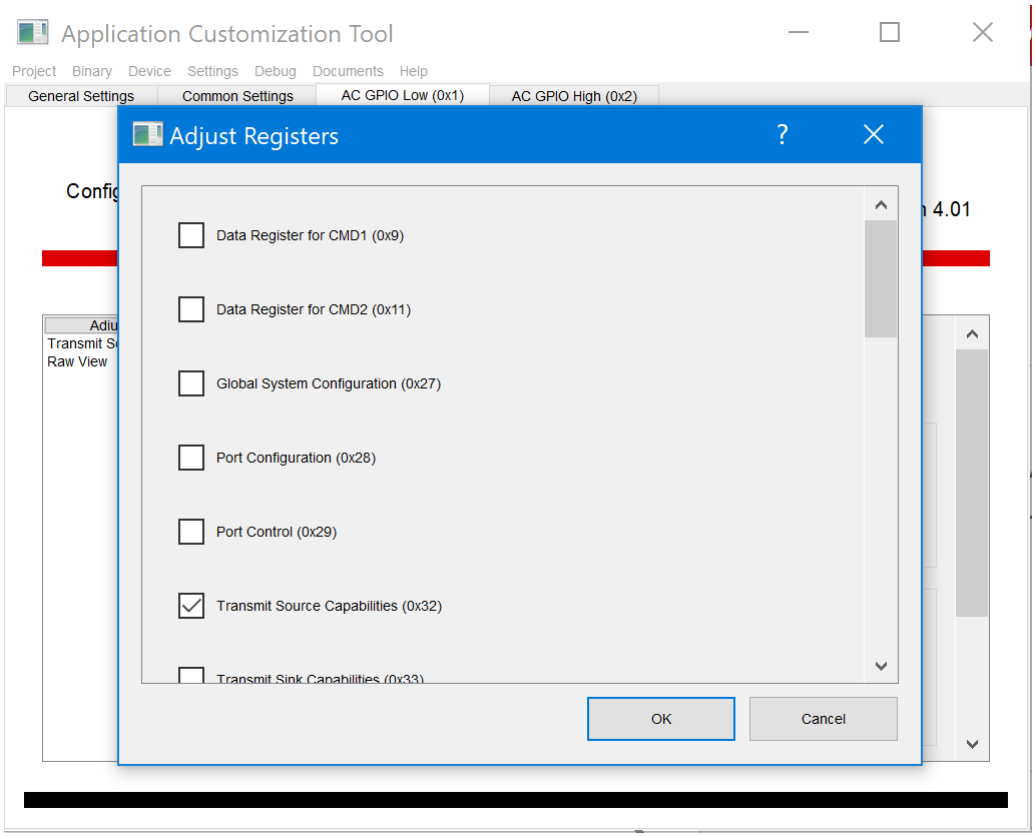

Figure 3-5. Adjust Registers in AC GPIO Low (0x1)

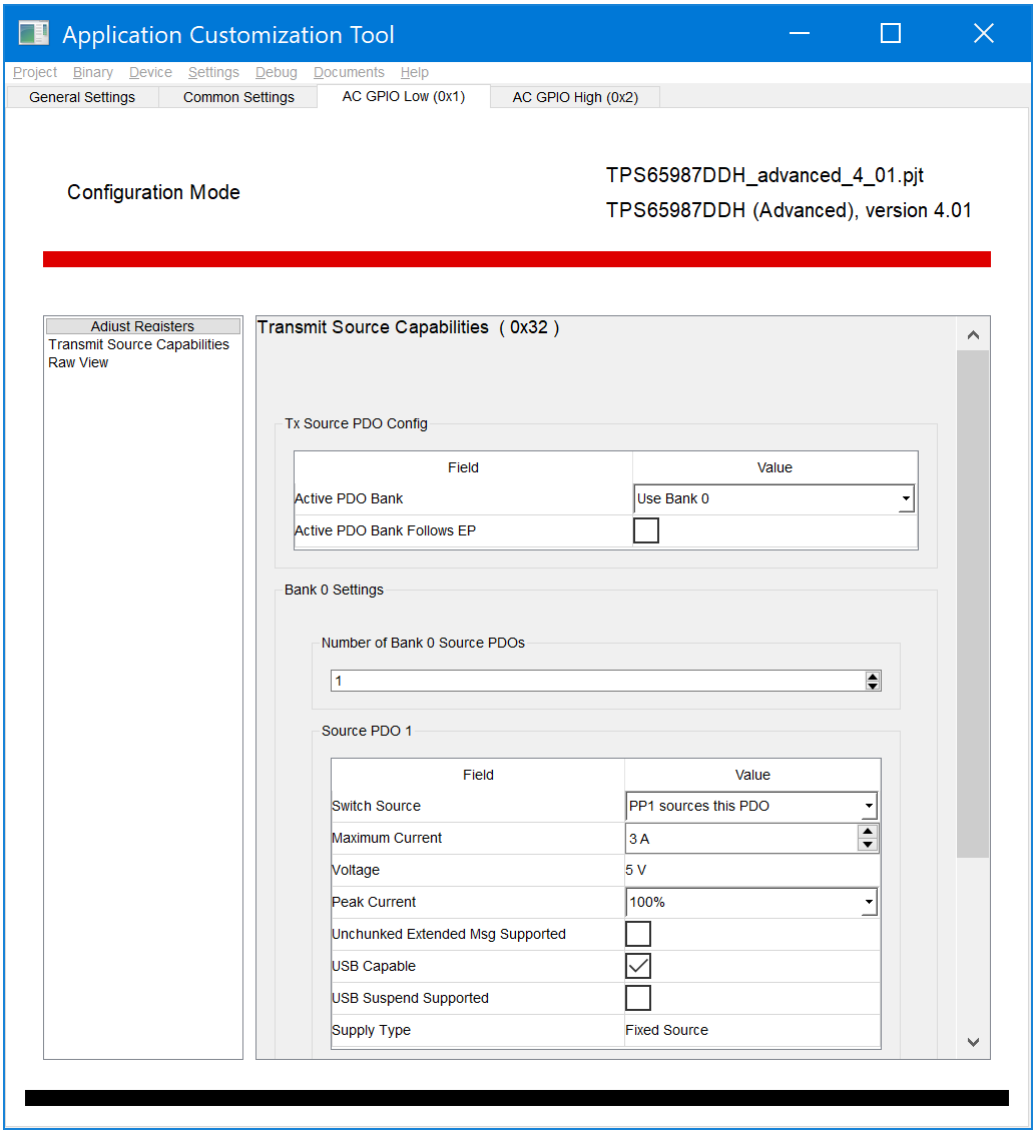

Figure 3-6. App Config GPIO Set Event, GPIO Low Settings Example

5. Adjust Registers in *AC GPIO High (0x2)* to set the behavior when GPIO4 transients from low to high. In this example, GPIO4 low to high transition would configure Transmit Source Capabilities register (0x32) with two PDOs as shown in Figure 3-7.

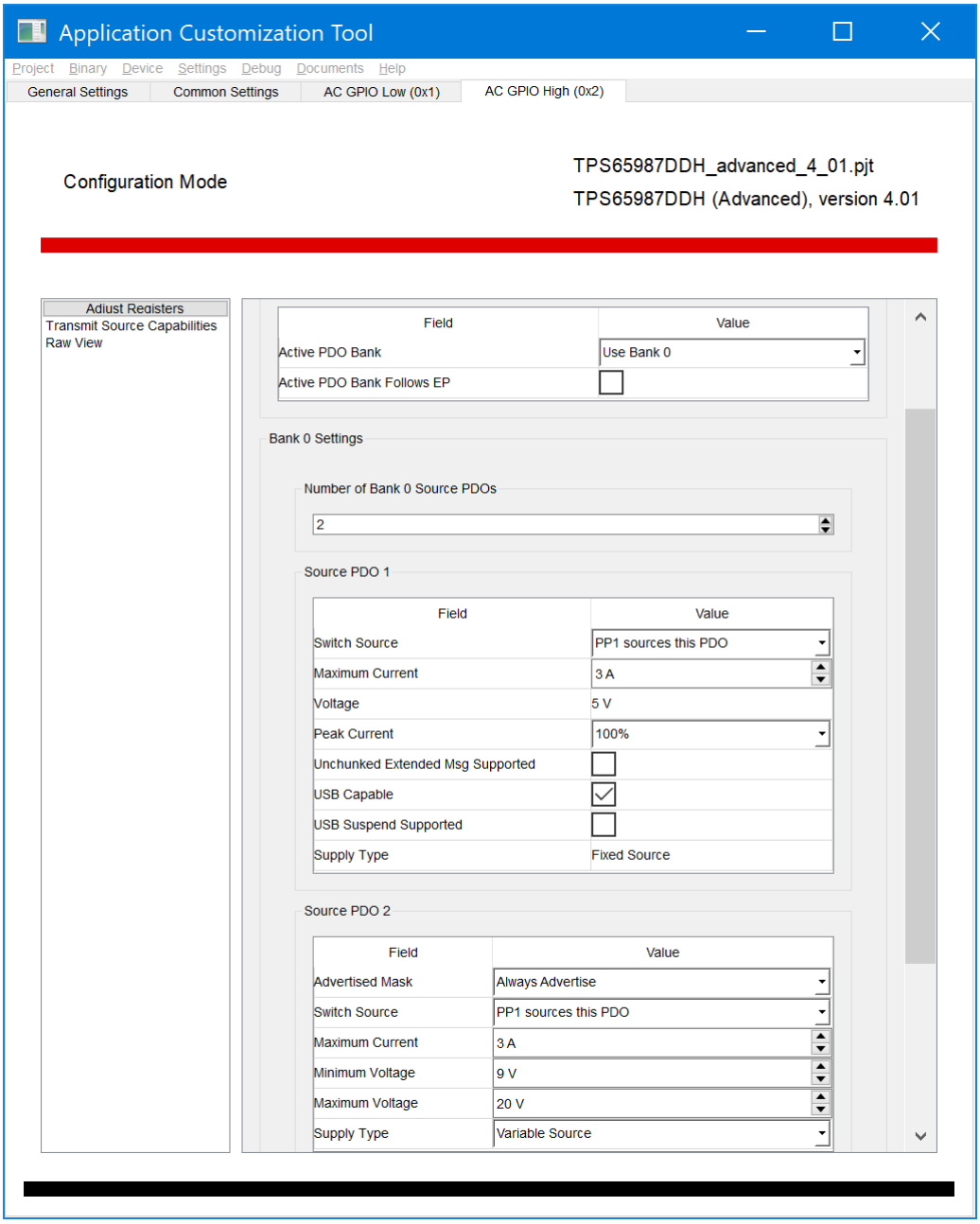

Figure 3-7. App Config GPIO Set Event, GPIO High Settings Example

6. Map *Load App Config Set 1* to *GPIO4* as shown in Figure 3-8.

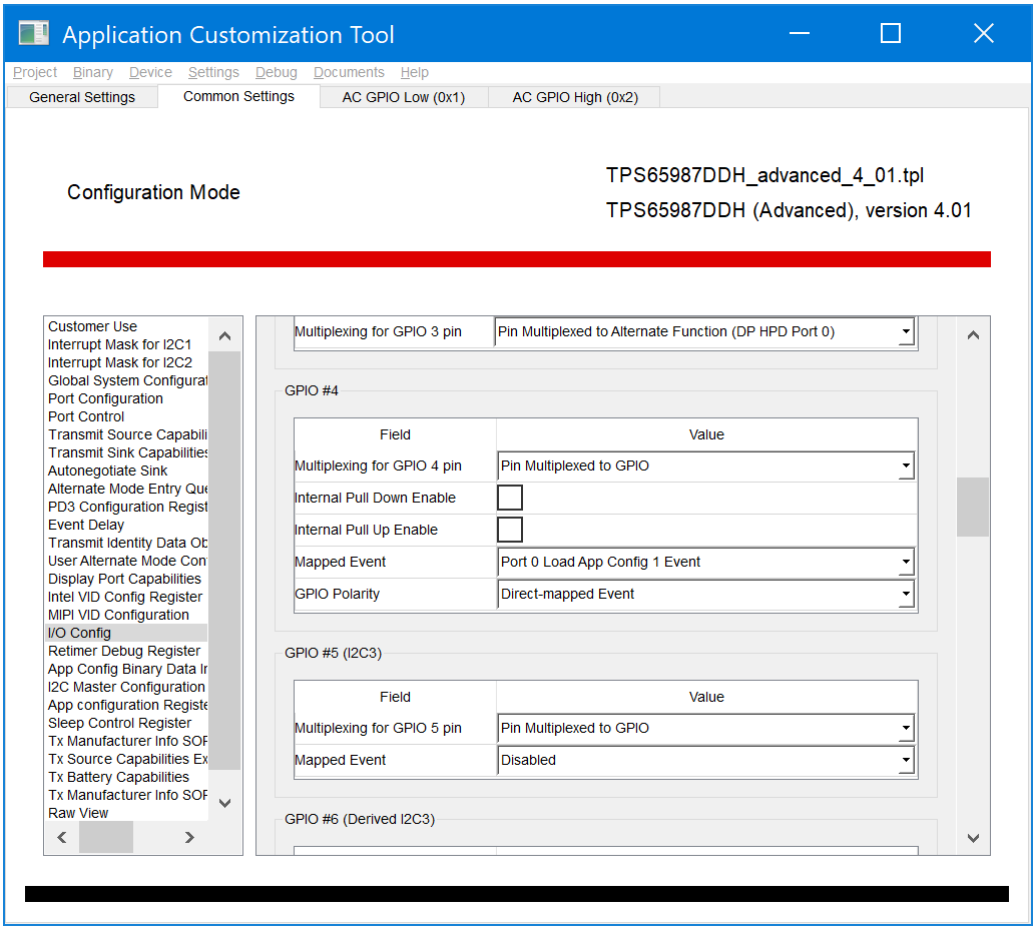

Figure 3-8. Map Load App Config Set 1 to GPIO 4

4 PD Controller Customization by GPIO Events

This section shows how TI PD controller GPIO events can be used in a system to alter system behavior while keeping the core firmware same. A Barrel Jack Event is used as example to show how a docking application can initiate power role swap when external power is connected to the system. Removal of the external power would generate PD traffic to reverse the power role swap and put the system back to original state.

4.1 Barrel Jack Connect Event PD Flow

Actual PD trace of this example Barrel Jack Event implementation in a system is shown in this section. This event can be used in a docking application when external power becomes available to the docking station. Rising edge on the GPIO that has been assigned for Barrel Jack Event initiates the required PD message flow for power role swap.

Two TPS65987D EVMs loaded with a binary created from the example template by clicking *Project→TPS65987DDH→Advanced→Dual Role Port (DRP), prefers data host→None (DisplayPort Only)* and GPIO21 is set to Barrel Jack Detect Event as shown in Figure 4-1.

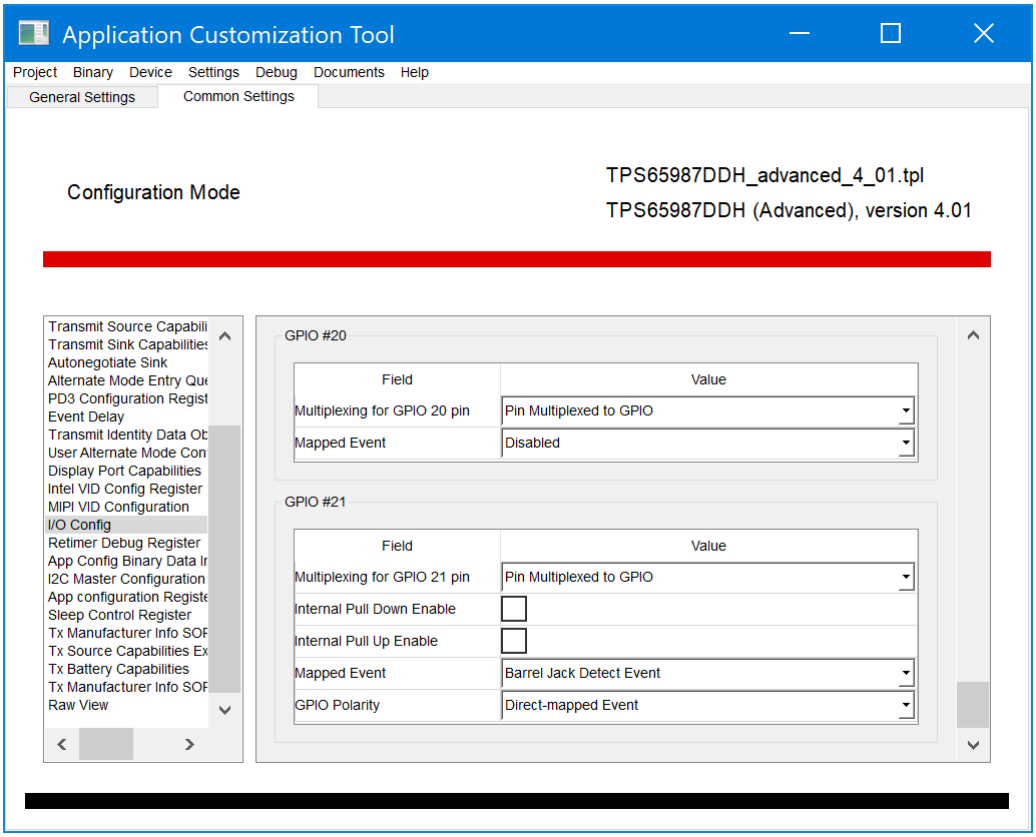

Figure 4-1. GPIO 21 Settings

PD message trace is taken with a Teledyne LeCroy PD analyzer between two TPS65987D EVMs as shown in Figure 4-2.

	Packet	Left		PD Msg	Msg Type	DR	PR	Msg ID	Obj Cnt	Duration	Idle	Time Stamp
PD	1	"Left"	SOP SNK	PD Msg	PR Swap	UFP	SNK	7	0	494.978 us	83.022 us	7 . 835 619 000
PD	Packet 2	Right "Right"	SOP SRC	PD Msg	Msg Type GoodCRC	DR DFP	PR SRC	Msg ID 7	Obj Cnt 0	Duration 496.617 us	Idle 119.383 us	Time Stamp 7 . 836 197 000
PD	Packet 3	Right "Right"	SOP SRC	PD Msg	Msg Type Accept	DR DFP	PR SRC	Msg ID 1	Obj Cnt 0	Duration 496.617 us	Idle 80.383 us	Time Stamp 7 . 836 813 000
PD	Packet 4	Left "Left"	SOP SNK	PD Msg	Msg Type GoodCRC	DR UFP	PR SNK	Msg ID 1	Obj Cnt 0	Duration 496.617 us	Idle 30.274 ms	Time Stamp 7 . 837 390 000
PD	Packet 5	Left "Left"	SOP SNK	PD Msg	Msg Type PS Ready	DR DFP	PR SNK	Msg ID 2	Obj Cnt 0	Duration 489.951 us	Idle 87.049 us	Time Stamp 7 . 868 160 328
PD	Packet 6	Left "Left"	SOP SNK	PD Msg	Msg Type GoodCRC	DR UFP	PR SNK	Msg ID 2	Obj Cnt 0	Duration 489.951 us	Idle 1.561 ms	Time Stamp 7 . 868 737 328
PD	Packet 7	Right "Right"	SOP SRC	PD Msg	Msg Type PS Ready	DR UFP	PR SRC	Msg ID 0	Obj Cnt 0	Duration 496.617 us	Idle 81.383 us	Time Stamp 7 . 870 788 000
PD	Packet 8	Left "Left"	SOP SNK	PD Msg	Msg Type GoodCRC	DR DFP	PR SNK	Msg ID 0	Obj Cnt 0	Duration 494.978 us	Idle 4.238 ms	Time Stamp 7 . 871 366 000

Figure 4-2. PD Trace of Barrel Jack Connect Event

Messages in Figure 4-2 represent PD traffic flow once the Barrel Jack adapter supplying 20 V is connected to the EVM-DCK configured with settings appropriate for a docking station.

Packet 1 → EVM-DCK is UFP/SNK and sends “PR Swap” message to the EVM-LPT which is DFP/SRC.

Packet 2 → DFP/SRC sends “GoodCRC” acknowledgment response for “PR Swap” message.

Packet 3 → DFP/SRC sends “Accept” message to signal that it is willing to do a Power Role Swap and has begun the Power Role Swap sequence.

Packet 4 → UFP/SNK sends “GoodCRC” acknowledgment response.

Packet 5 → EVM-LPT changes role to DFP/SNK and sends “PS Ready” message. It is important to note that the initial Source Port is now setting the “Port Power Role” field to Sink in the “PS Ready” message indicating that the initial Source’s power supply is turned off.

Packet 6 → EVM-DCK sends “GoodCRC” acknowledgment response for “PS Ready” message. Note that the GoodCRC Message sent by the initial Sink in response to the “PS Ready” message from the initial Source will have its Port Power Role field set to Sink since this is initiated by the Protocol Layer.

Packet 7 → EVM-DCK changes role to UFP/SRC and sends “PS Ready” message.

Packet 8 → EVM-LPT which is now DFP/SNK sends “GoodCRC” acknowledgment response.

4.2 Barrel Jack Remove Event PD Flow

Once power is removed from the EVM-DCK, falling edge generated on the GPIO would initiate the reverse process so that EVM-LPT can become the Power Source again. Actual PD trace of the removal event is shown in Figure 4-3.

PD	Packet	Right	SOP	PD Msg	Msg Type	DR	PR	Msg ID	Obj Cnt	Duration	Idle	Time Stamp
PD	1	"Right"	SOP SRC →	PD Msg	PR Swap	UFP	SRC	3	0	496.617 us	80.383 us	5 . 327 193 000
PD	2	"Left"	SOP ← SNK	PD Msg	GoodCRC	DFP	SNK	3	0	496.617 us	120.383 us	5 . 327 770 000
PD	3	"Left"	SOP ← SNK	PD Msg	Accept	DFP	SNK	1	0	496.617 us	81.383 us	5 . 328 387 000
PD	4	"Right"	SOP SRC →	PD Msg	GoodCRC	UFP	SRC	1	0	496.617 us	30.258 ms	5 . 328 965 000
PD	5	"Left"	SOP ← SNK	PD Msg	PS Ready	UFP	SNK	4	0	489.951 us	88.033 us	5 . 359 719 328
PD	6	"Left"	SOP ← SNK	PD Msg	GoodCRC	DFP	SNK	4	0	488.334 us	1.587 ms	5 . 360 297 312
PD	7	"Right"	SOP SRC →	PD Msg	PS Ready	DFP	SRC	2	0	496.617 us	80.383 us	5 . 362 373 000
PD	8	"Left"	SOP ← SNK	PD Msg	GoodCRC	UFP	SNK	2	0	496.617 us	24.676 ms	5 . 362 950 000

Figure 4-3. PD Trace of Barrel Jack Removal Event

5 Status Register and 4CC Commands

GPIO status can be monitored by reading a register and system controller can take appropriate actions based on that. There are also GPIO related 4CC commands that can be used by system controller to alter GPIO behavior.

Status Register

- 0x72, GPIO Status

4CC Commands

- ‘GPie’, GPIO Input Enable
- ‘GPoe’, GPIO Output Enable
- ‘GPsh’, GPIO Set Output High
- ‘GPsl’, GPIO Set Output Low

Status register and 4CC command capabilities of TPS6598x Application Customization tool provides a way to test and modify GPIO configurations of a real system. GPIO configurations can be changed on-the-fly over I2C bus to try new settings quickly. Once the expected system behavior is confirmed, appropriate GPIO configurations can be implemented through the system controller processor.

5.1 GPIO Status Monitoring

GPIO status register can be used to monitor various GPIOs that are configured to achieve desired system behavior. For example to support PD Power Rules with 5 V, 9 V, 15 V and 20 V variable supplies, TPS65987D EVM is designed to use PDO GPIO events that trigger the power supply circuit and generate the desired voltage output. In this case GPIO14, GPIO15, GPIO17 and GPIO20 are assigned with appropriate PDO events to achieve the variable DC-DC supply. Figure 5-1 shows that GPIO14 is driven high indicating PD contract is done for 5 V. Once an explicit PD contract is negotiated for 20 V supply, GPIO20 is driven High by the PD controller as indicated in Figure 5-2.

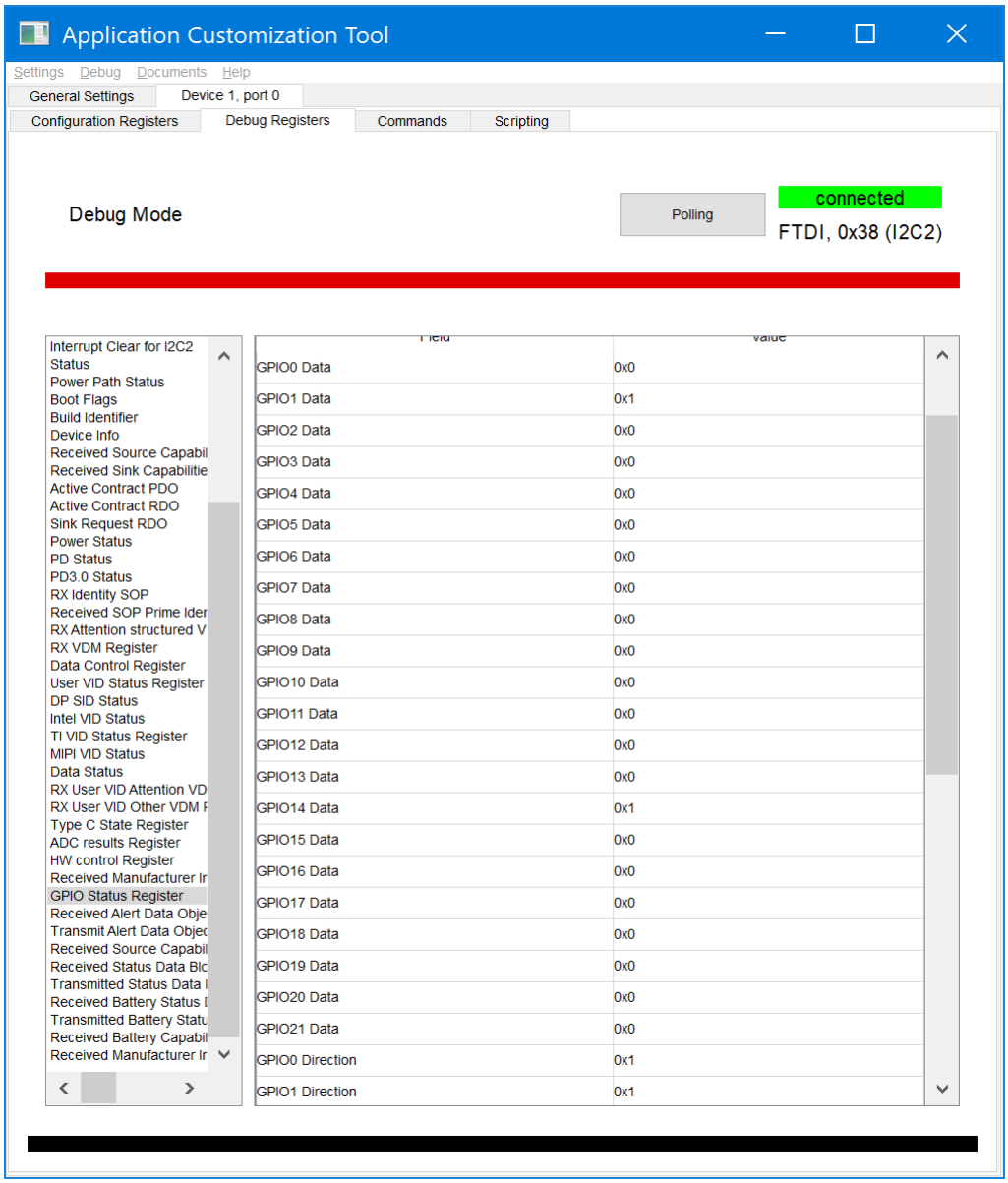

Figure 5-1. Variable DC/DC GPIO Status for 5-V Supply

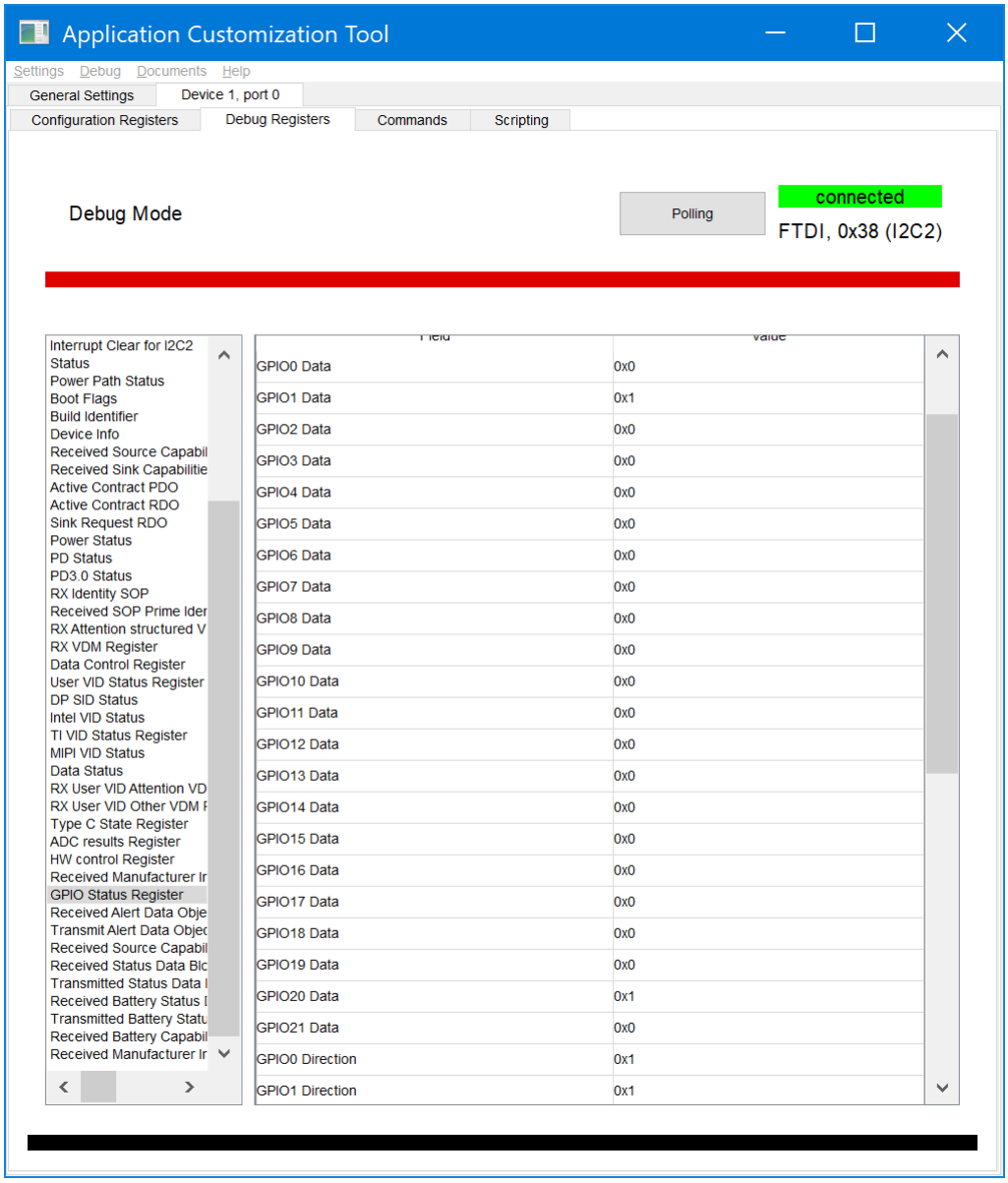

Figure 5-2. Variable DC/DC GPIO Status for 20-V Supply

5.2 Using 4CC GPIO Commands

TPS6598x Application Customization tool can be used to exercise the GPIO related 4CC commands and observe, develop system behavior before system controller implements the desired driver software. Figure 5-3 shows the commands list page of the tool that can be used to exercise the 'GPxx' 4CC commands.

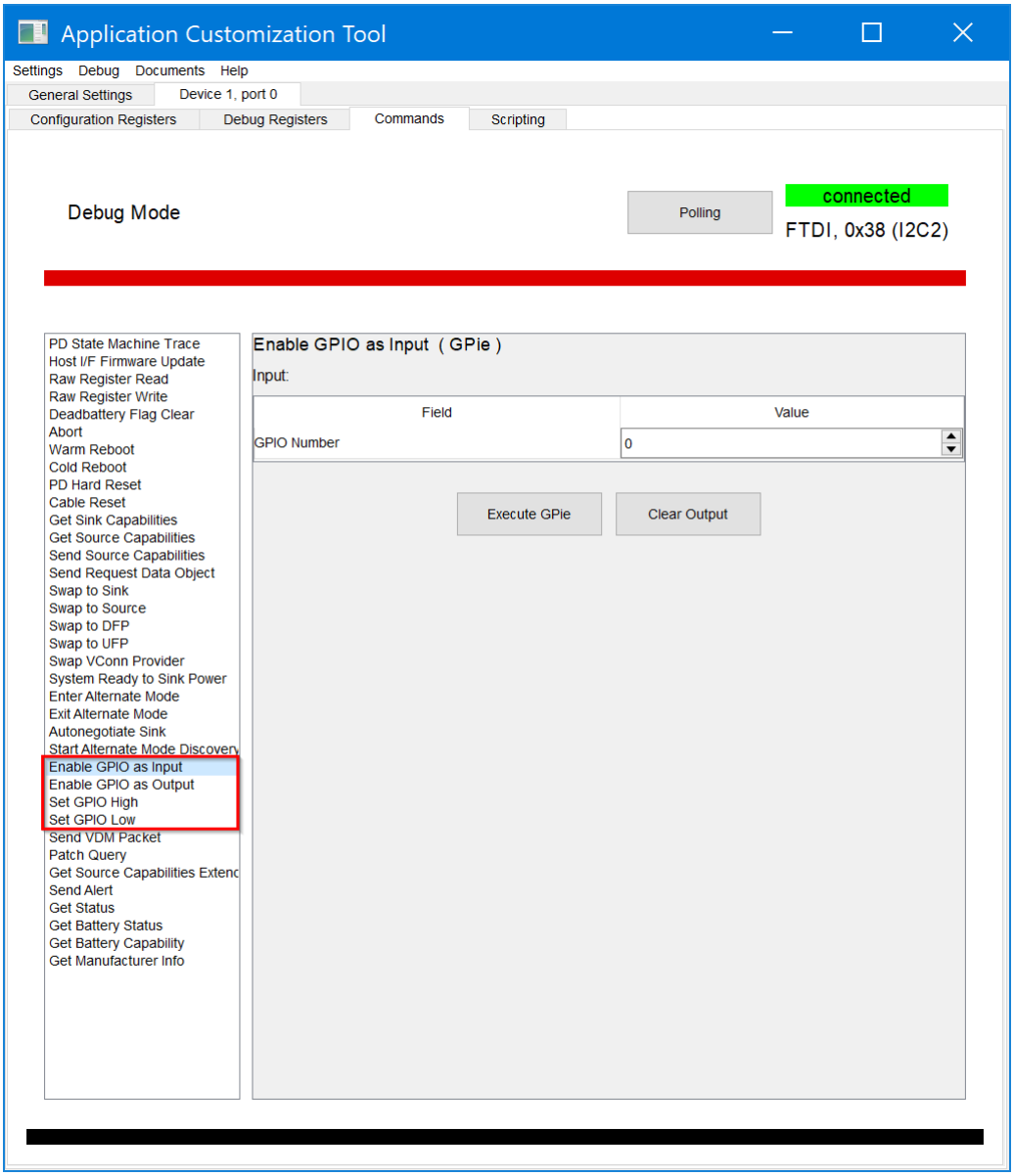

Figure 5-3. 4CC Commands in TPS6598x Application Customization Tool

For example, to set the GPIO7 to High:

- First send ‘GPoe’ 4CC command as shown in Figure 5-4.
- Then send ‘GPsh’ 4CC command as shown in Figure 5-5.
- In the GPIO Status (0x72) it can be seen that GPIO7 has been set to High.

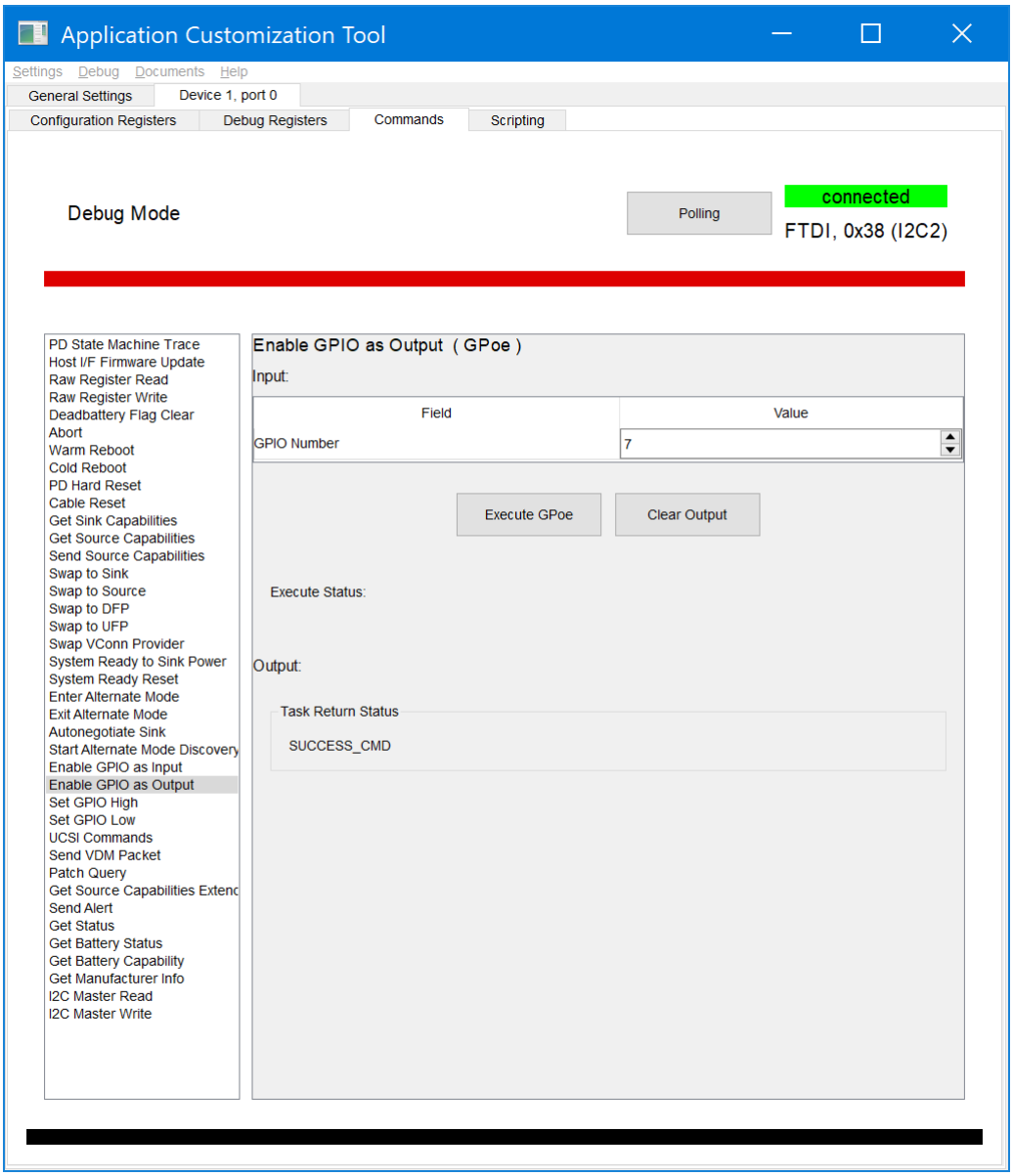

Figure 5-4. Using ‘GPoe’ 4CC Command

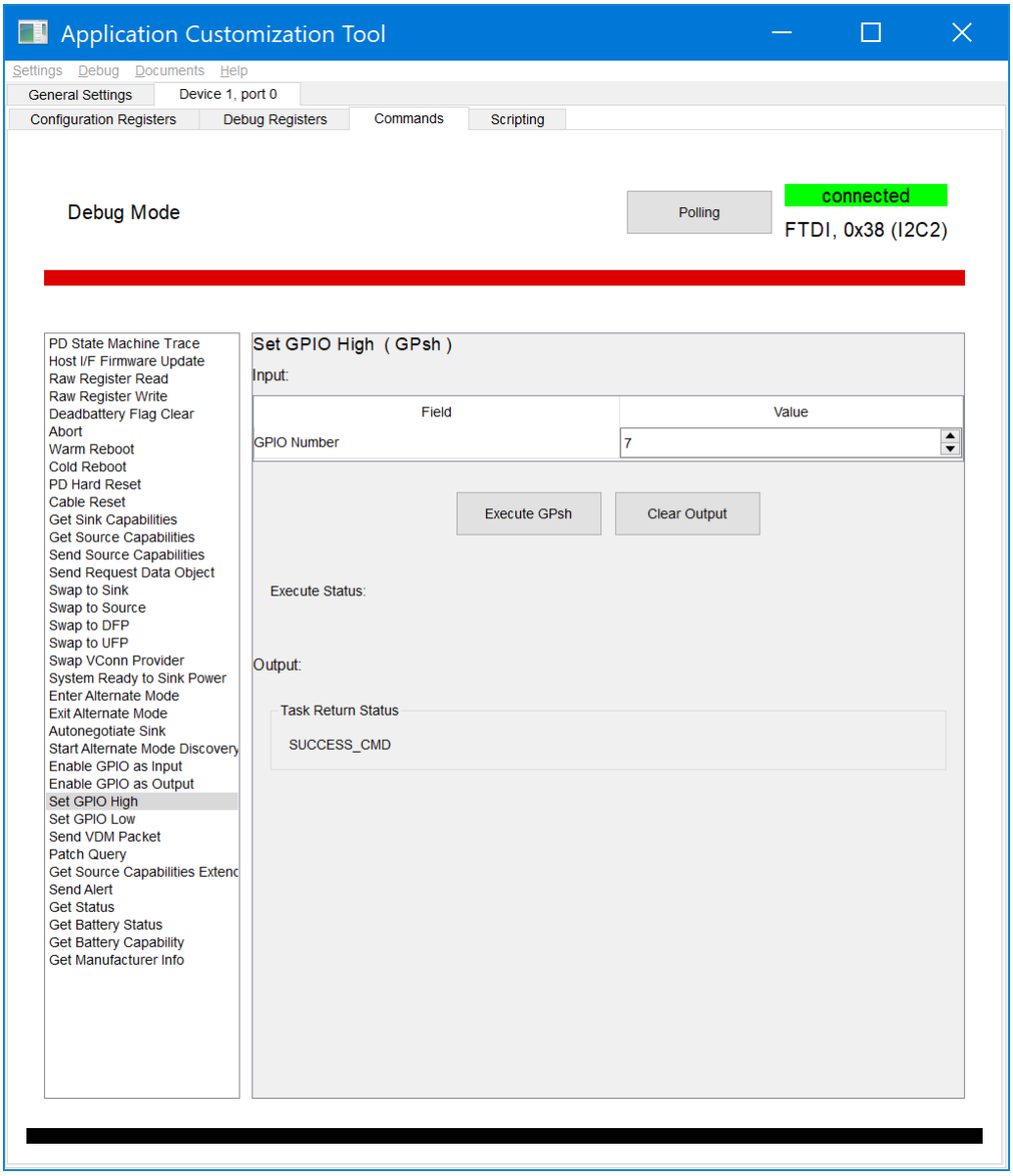

Figure 5-5. Using 'GPsh' 4CC Command

6 Revision History

NOTE: Page numbers for previous revisions may differ from page numbers in the current version.

Changes from Revision * (August 2018) to Revision A (January 2021)	Page
• Updated the numbering format for tables, figures, and cross-references throughout the document	1
• Updated *List of TPS65987D GPIO Events* content	2
• Changed "9 V at approximately 20 V" to "9 V minimum 20 V maximum"	7

IMPORTANT NOTICE AND DISCLAIMER

TI PROVIDES TECHNICAL AND RELIABILITY DATA (INCLUDING DATA SHEETS), 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 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 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.

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

Mailing Address: Texas Instruments, Post Office Box 655303, Dallas, Texas 75265