Application Note TPS65988DH: Integrated Type-C PD Dual-Port Wall Socket With Power Balancing



Prajith Cheerakkoda, Shashank Meti

ABSTRACT

Higher power and wider compatibility with the battery powered devices makes a Type-C PD power port the preferred choice of today's gadget lovers. Consequently, integrated Type-C PD wall sockets are gaining more popularity than the Type-A integrated wall sockets at a faster rate. Why TI's highly configurable and integrated PD controller TPS65988DH is an integral part of your next Type-C/ PD design is the topic of this application note. With the highly flexible GPIO event feature, TI PD controller can implement the control logic required for power balancing between the ports and function as a complete standalone solution. This feature allows for an otherwise mandatory MCU (Microcontroller unit) to be removed from the system, alleviating firmware development and system complexity. This application note walks through how GPIO event feature has been utilized to implement power balancing between the ports. This feature is beneficial when designing end equipments like smart plugs, multi-port adapters, avionics seat backs, and other multi-port source-only applications using TI's highly configurable PD controller.

This application note also covers step by step instructions on GUI configuration for implementing the control logic required for power balancing. The example project file available with this application note can be used for testing the control logic on TPS65988EVM.

Table of Contents

1 Introduction	2
2 Integrated USB Wall Socket Design	3
2.1 Thermal Considerations	4
2.2 Dynamic Port Power Balancing	
2.3 GPIO Events	
2.4 Control Logic Implementation with GPIO Events.	7
2.5 GUI Configuration	
2.6 Test Setup and Results	13
3 References	
4 Revision History	18

List of Figures

Figure 2-1. Integrated Type-A USB Charging Port Block Diagram	3
Figure 2-2. Integrated Type-C PD USB Charging Port Block Diagram	
Figure 2-3. Dual Port Integrated Type-C PD Wall Socket Architecture Using TPS665988DH's Internal Power Switch	
Figure 2-4. Dual Port Power Socket Design With Power Balancing	5
Figure 2-5. GPIO Event	
Figure 2-6. GPIO Events Usage in Dual Port Power Socket Design with Power Balancing	
Figure 2-7. Load Template File	
Figure 2-8. Generate Configuration Sets and Map to Virtual Address	
Figure 2-9. App Configuration Register Entry	10
Figure 2-10. Select Transmit Source Capabilities Register	11
Figure 2-11. Configure Transmit Source Capabilities	
Figure 2-12. GPIO Mapping	13
Figure 2-13. EVM Block Diagram	
Figure 2-14. GPIO Setting on EVM	15
Figure 2-15. PD Trace Showing PDOs Corresponding to Single and Dual Port Connections	

Figure 2-16. Scope Capture of Vbus Transitions.....



.....17

Trademarks

USB Type-C $^{\textcircled{R}}$ is a registered trademark of Texas Instruments. All trademarks are the property of their respective owners.

1 Introduction

In an effort to reduce e-waste, USB-IF redefined the role of a USB port as a universal interface for power too. Focus was given for faster and efficient power transfer to meet the growing needs of battery-powered applications. This approach was widely accepted in the USB world and is also gaining popularity among the non-USB OEMs as a preferred method of power delivery, such as electric shavers, portable speakers. Unlike Type-A ports on the wall socket, a Type-C PD port with higher power can serve a wider spectrum of consumer electronics and really help to reduce adapter clutter.

The intrinsic design challenges (restricted PCB area and thermal dissipation inside the gang-box) in a wall socket design make the Type-C PD integration more challenging. That being said, let's see how TI's highly integrated and configurable TPS665988DH PD controller can be of help to roll out your next Type-C PD wall socket, with a short development cycle to keep up market pace.



2 Integrated USB Wall Socket Design

An integrated USB wall socket accommodates USB charging ports on the face plate along with the AC outlet terminals.

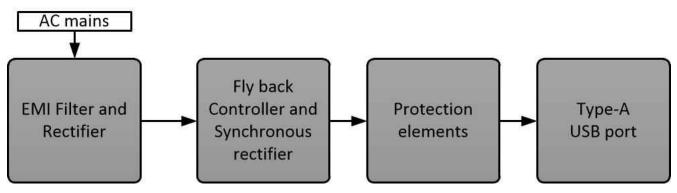


Figure 2-1. Integrated Type-A USB Charging Port Block Diagram

Figure 2-1 shows different blocks that constitute a conventional USB Type-A charging port on a wall socket. The EMI filter and rectifier block generate high voltage DC from AC mains input. The high frequency isolated fly back controller and synchronous rectifier block provides regulated DC for the charger port. The protection elements are meant for protecting the USB port from faults such as under voltage protection (UVP), over voltage protection (OVP), reverse current protection (RCP) and over current protection (OCP).

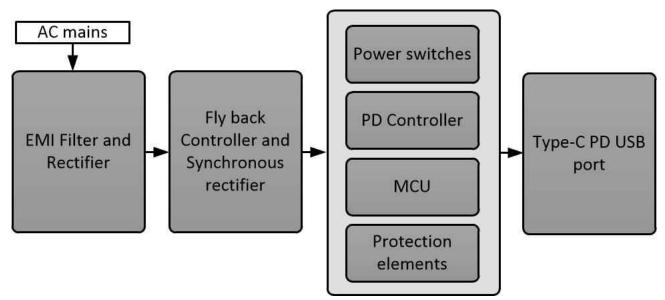


Figure 2-2. Integrated Type-C PD USB Charging Port Block Diagram

Now, as shown in Figure 2-2 a Type-C PD charging port requires few additional components compared to conventional Type-A port.

- 1. PD controller: To add Type-C PD functionality.
- 2. Power switch: to control the power delivery to the port as per PD specification.
- 3. Protection elements: to protect USB port from faults such as OVP, UVP, OCP, RCP, and so forth.
- 4. MCU: Required in fixed function PD controller to implement customized power policy (if required).

Manufacturers of plugs and sockets have to follow mechanical dimensions defined by the regulatory standards (Eg BS 1363). One electrical gang box is usually of 4"(L)x2.75"(W)x2.5"(D) in size. The actual available area for PCB inside the box will be smaller than this as the gang box behind the front panel has to accommodate thick wires connecting AC terminals on the face plate. Now, with these restrictions, finding additional space for the Type-C PD related components in the wall socket design can be a challenge for system designers.



Today, most of the PD controllers in the market require external power mosfets for controlling power delivery to the Type-C port. System designers using such PD controllers will have to build their own protection circuits around the external power switches with the expense of additional PCB area and BOM. As per Type-C/ PD specification OCP protection is mandatory for Source ports (Type-C PD port that delivers power). TI's highly integrated dual Port PD controller TPS665988DH comes with integrated power switches and built in protection features, avoiding the need for these additional external components.

USB PD device applications often require customizations in order to achieve desired system behavior. For instance, customization is required in the dual port wall socket design (to be discussed later in this application note) for live power balancing between the ports. Any customization in the core firmware affects reliability, USB PD compliance and also slows down the development cycle. Typically, systems using fixed function PD controllers are required to have an on board microcontroller in this regard, with the TI PD controller being an exception. The GPIO event features of TPS65988DH allow PD system designers to customize the PD controller behavior using GUI software tool without affecting the core PD firmware.

2.1 Thermal Considerations

Face plate is the only means for dissipating the heat generated inside a gang box. Hence, focus should be given in selecting components with less power dissipation. In integrated wall socket designs, the maximum output power of USB ports are capped by the efficiency of ACDC and DCDC regulators. The power dissipated in TPS665988DH's internal power switch with Rdson of 28m Ω (at Junction temperature of 84°C) is very less compared to the heat contributed by the aforementioned components on the power path. The junction temperature calculation at ambient temperature of 65°C shows that TPS665988DH is well below the max junction temperature of 150°C.

- Ambient temperature inside the Gang box (mainly contributed by ACDC transformer) (Ta) = 65°C
- Junction-to-ambient thermal resistance of TPS65988DH (R_{θJa}) = 36.4°C/W
- Power dissipation on internal power paths due to IPPHV(continuous current through power path from PP_HVx to VBUSx) of 3A = Rdson * lout * lout = 28 m Ω * 3 A * 3A = 252 mW.
- Total power dissipation from both power paths = 504 mW
- Total power dissipation (considering GPIO leakage and active power I_{VIN 3V3}) P ~= 520mW
- Junction temperature = Ta + (R_{θJa} * P) = 65 °C + (36.4°C/W * 0.52W) = 83.9°C

The Junction temperature can be brought down to a lower value by careful PCB design as recommended in the data sheet.

The above thermal calculation is considering the architecture (shown in Figure 2-3), which uses internal power paths of the TI PD controller. In this design, the Buck-Boost regulator is fed from the output of ACDC regulator. Therefore, the ACDC regulator is required to maintain the minimum voltage required for the Buck-Boost regulator to operate. The I2C master of PD controller can be programmed to configure the regulators to supply fixed DC voltage based on the PD negotiation on the respective ports. More details on configuring TPS665988DH's I2C master to trigger slave write on specific PD events can be found Using I2C Master in TPS65987D and TPS65988 PD Controllers application report.



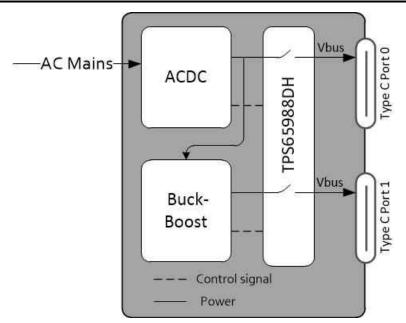


Figure 2-3. Dual Port Integrated Type-C PD Wall Socket Architecture Using TPS665988DH's Internal Power Switch

2.2 Dynamic Port Power Balancing

Higher level of integration available in TPS665988DH PD controller makes it the best suited part for power socket design. Protection features such as over current, over voltage, under voltage, and reverse current protections are highly desirable in any kind of power supply designs and most of the time system designers have to add these protection elements externally. The 100W capable internal integrated power paths come with all these protection features built in it.

For effective use of system power, the design is required to dynamically share the available power between the ports based on the port connection as listed below.

- When no ports are connected, the total power shall be made available to the first device, getting connected to either of the ports.
- When the second device gets connected, the total power shall be split equally b/w the two ports.

For the discussion, let's assume that the maximum system power available is 60W.

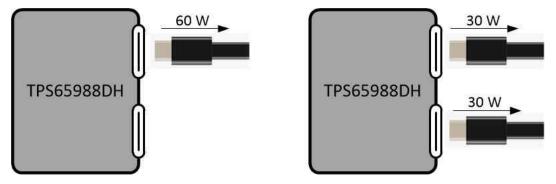


Figure 2-4. Dual Port Power Socket Design With Power Balancing

Without firmware modification, this can be achieved using real-time loading of modified PD Port configuration, which usually requires an external MCU. The same design can be made possible with TPS665988DH using GPIO events, without any firmware modification and without the need for an external MCU.

Each port can be configured to have two port power configurations and select them dynamically based on the control logic.



- **Configuration-1:** *100% Power PDO* [1] (advertise total power available to the system, 60W in this example, when only one port is connected)
- **Configuration-2:** 50% Power PDO (50% of the total power, 30W in this example, when both ports are connected).

Note

Power Data Object (PDO): Data Object used to expose a Source Port's power capabilities as part of a Source_Capabilities message.

GPIO events are used to dynamically load the appropriate configuration (100% or 50% PDOs) at run time for a particular port, based on Plug Event from the other port. The following section introduces the GPIO events and its usage for this particular design as an example.

2.3 GPIO Events

GPIO events feature enables users to map a PD or USB Type-C event inside the PD controller to a GPIO (see Figure 2-5).

- An output GPIO event asserts/de-asserts a GPIO pin based on the user selected polarity, when the mapped PD/ Type-C event occurs. For example, *Plug Event* can be used to reflect Type-C plug/unplug on a GPIO.
- Similarly, an Input GPIO event (rising or falling edge on a TPS665988DH GPIO pin, from an external signal in the system) can be used to trigger a specific PD or USB Type-C event inside the PD controller. For example, the *Fault Input Event* allows external devices to trigger error recovery on a given port.[2].

Using the Configuration Tool, a predefined event picked from the drop-down menu can be mapped to any of the GPIOs. A list of all the GPIO events can be found in the *TPS65987DDH and TPS65988DH Host Interface Technical Reference Manual* of the PD controller.

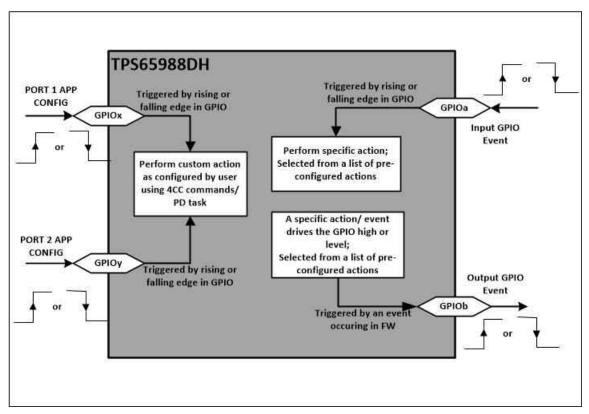


Figure 2-5. GPIO Event

 Unlike input GPIO event that triggers a predefined PD or USB Type-C[®] event, an Application Configuration GPIO event provides more flexibility in the design to trigger custom actions using an input GPIO (see Figure 2-5). Custom actions could be configured by the user using 4CC commands or PD tasks in Configuration



Tool, which otherwise needs an EC for issuing the same. In addition, an input GPIO mapped to an Application configuration event can be used to load a modified configuration to the PD controller at run-time.

Note

- For details on 4CC commands/PD tasks, see the *TPS65987DDH and TPS65988DH Host Interface Technical Reference Manual.*
- Error recovery is a Type-C state to recover from any fault state. During error recovery state, a Type-C port removes CC termination for a defined amount of time and puts it back. This is to simulate a detach event and to make a new connection.

2.4 Control Logic Implementation with GPIO Events

Figure 2-6 shows pictorial representation of Events and GPIOs used in the Dual port power socket design with power balancing.

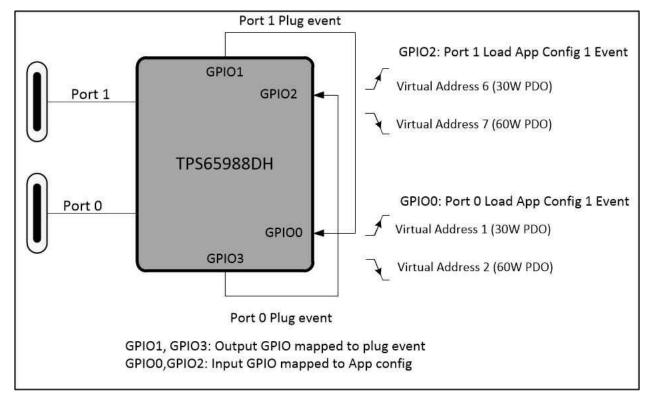


Figure 2-6. GPIO Events Usage in Dual Port Power Socket Design with Power Balancing

- Output Plug event GPIOs: GPIO1 (Port-1 plug event) and GPIO3 (Port-0 plug event) are configured to
 assert on *PLUG_EVENT* event. These output GPIOs should assert high when a device is connected to the
 corresponding Port.
- Application Configuration event GPIOs: GPIO2 (Port-1 App Config) and GPIO0 (Port-0 App Config) are input GPIOs configured to load the appropriate configuration (100% or 50% PDOs) based on the input signal and also execute a PD message task (to send SSrC - PD Send Source Capabilities).

The output Plug event GPIOs are connected to the input App Config GPIOs of the alternate port. When the input GPIO of a particular port is high (implies plug event on the other port), input GPIO event shall load 50% Power PDO configuration. Otherwise, when the input GPIO of a particular port is low (implies no *PLUG_EVENT* on the other port, for example, when the other port in disconnected state), input GPIO event shall load *100% Power PDO* configuration. This implementation can be tested on TPS65988D EVM using the project file available on the E2E Design Support forum.



2.5 GUI Configuration

- 1. The first step is to create a new DFP only (Source only) project by loading the default template TPS65988DDH_advanced_v6_1_2.tpl.
 - a. Open TPS6598x Configuration Tool and click the Project tab.
 - b. Select TPS65988DDH→Advanced→Downstream Facing Port (DFP) only.

Application Customiza	tion Tool			1		
Adapter Documents	New Project				?	, ,
Configuration Mode	What is the port type of the design? Upstream Facing Port (UFP) Onl Downstream Facing Port (DFP) (Dual Role Port (DRP), prefers da Dual Role Port (DRP), prefers po Dual Role Port (DRP), prefers po	Only ita host wer source				
Firmware Base Image : Change File Firmware Version: 0000. Allocated Application Conf Used Application Configur						
Device Initialization Chain Number of Connected D						
Device					Can	cel
Device 1			d) 🔻			1000

Figure 2-7. Load Template File

- 2. As shown in Figure 2-6, the application requires two sets of power profiles per port: corresponding 30W and 60W PDOs. In this step, generate these configurations and map them to virtual addresses.
 - a. Change Number of Configuration Sets in General Settings to four.
 - b. Rename Configuration Set as shown below for convenience.
 - i. Virtual Address 1 -> Port-0 30W
 - ii. Virtual Address 2 -> Port-0 60W
 - iii. Virtual Address 3 -> Port-1 30W
 - iv. Virtual Address 4 -> Port-1 60W
 - c. Now, map Configuration sets to Virtual Address as shown in Figure 2-8.

TEXAS INSTRUMENTS www.ti.com

Integrated USB Wall Socket Design

Application Customization Tool					-8	
t <u>Binary Device</u> <u>Settings</u> <u>Adapter</u>						
eral Settings Port 1 Settings Port	t 2 Settings Port	-0 30W (0x1)	Port-0 60W (0x2)	Port-1 30W (0x6)	Port-1 60
		GUI Build V	Version: 6.1.2			
		Configuratio	on File Version : 6.1.2	2		
Configuration Mode		Configuratio	on File Supported De	evice : TPS65	5988 EV	М
		Configuration	on File Name: TPS6	5988DH_Adv	vanced_v	/6_1_2.tp
		USB to I2C	SPI Adapter : FTDI			
		Constant of the second				
Number of Connected Devices: 1 -			Share Settin	igs Across All D	evices:	
	4			5	-	
Device	Ports		Addressin	Ig		
Po	ort1 (0x0)	100	222		12C1: 0x2	2.74
Device 1		12C_A	/R2 = 0.00-0.18) *		12C2: 0x3	355.0
Po	ort2 (0x4)	UKI	///2 - 0.00-0.10)		12C1: 0x2	A. (*
	a					
Configuration Data Sets						
Number of Configuration Sets: 4					С	
Number of Comiguration Sets. 4	1					
Configuration Set			(Virtual) Pin Strap			
Port-0 30W			Virtual Addres			
			1 Virtual Addres			
Port-0 60W			2 .			
Port-1 30W			Virtual Addres	ss		
Port-1 30W			6			
Port-1 30W Port-1 60W				• 55		-

Figure 2-8. Generate Configuration Sets and Map to Virtual Address

Note Do not use address 0x00 and 0x04 as they are reserved for port configurations.

- 3. The App configuration register (0x6C) holds the index pointing to the application configuration that needs to be loaded when the GPIO assert/ de-assert and also holds the command channel interface for executing the PD message task. In this step, modify Port-0 *App Configuration Register* to point to the application configuration index *Virtual Address 1* on GPIO high transition (corresponding to application configuration Port-0 30W) and point to *Virtual Address 2* on GPIO high transition (corresponding to application configuration Port-0 60W) and then execute SSrC task using CMD1 channel.
 - a. Click the Port 1 Settings tab.
 - b. Click the App Configuration Register.

- c. The index of configurations that need to be loaded on GPIO low/high transition are selected from the drop-down menu below *Value* tab.
- d. The default command channel CMD1 (0x08) can be used for executing PD message task.
- e. Select PD message task *SSrC* from the drop-down menu to transmit the Source Capability configured in the application configuration space indexed by Virtual Address 1 or Virtual Address 2. Enter *!CMD* in *not Task* field to indicate no action.

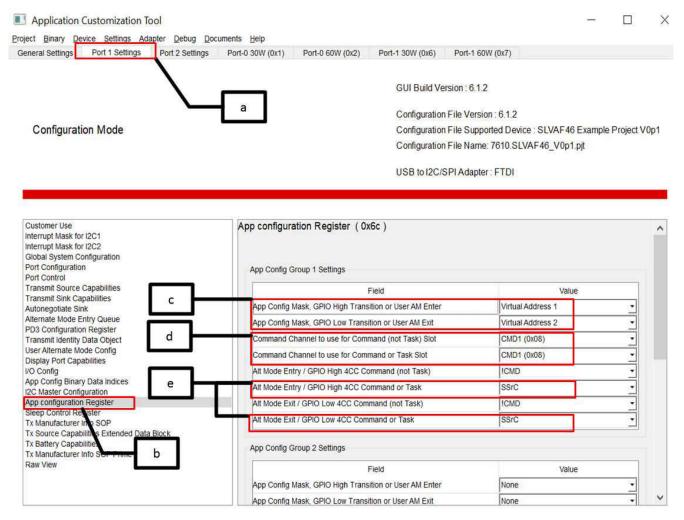


Figure 2-9. App Configuration Register Entry

- 4. Similarly, make entry for Port-1 App Configuration Register by referring the pjt file attached.
- Now, configure Transmit Source Capabilities register (0x32) pointed by Virtual Address indexes. Virtual address 1 is the index corresponding to 30W PDO.
 - a. Click Port-0 30W (0x1) and then click Adjust Registers.
 - b. From the pop-up box, select Transmit Source Capabilities (0x32) Register.
 - c. Click OK.
 - d. Click Transmit Source Capabilities and make entry for 30W PDO as shown in Figure 2-11.



ct Binary Device Settings Adap			
neral Settings Port 1 Settings	Port 2 Settings Port-0 30W (0x1) Port-0 60W (0x2) Port-1 30W (0x6) Port-1 60W (0x7)		
	Adjust Registers	?	>
Configuration Mode	Data Register for CMD1 (0x9)		3
	Data Registration 01D2 (0x11) a Global System Configuration (0x27)		
Adjust Registers lobal System Configuration ort Configuration ort Control	Port Configuration (0x28)		ļ
ransmit Source Capabilities ransmit Sink Capabilities utonegotiate Sink ternate Mode Entry Queue	Port Control (0x29) Transmit Source Capabilities (0x32)		
ransmit Identity Data Object ser Alternate Mode Config isplay Port Capabilities C Master Configuration ebug Control Register	Transmit Sink Capabilities (0x33)		
eep Control Register « Manufacturer Info SOP « Source Capabilities Extended Data B « Battery Capabilities	Autonegotiate Sink (0x37)	<u>ل</u>	
x Manufacturer Info SOP Prime aw View		Cance	
	IZCT Enable as Master	Carle	a
	I2C3 Enable as Master		

Figure 2-10. Select Transmit Source Capabilities Register



ect Binary De	evice Settings Hu	apter Debug Doc	and Telb				
eneral Settings	Port 1 Settings	Port 2 Settings	Port-0 30W (0x1)	Port-0 60W (0x2)	Port-1 30W (0x6)	Port-1 60W (0x7)	
				GUI Build Ve	ersion : 6.1.2		
				Configuratio	n File Version : 6.1.2		
Configurat	tion Mode			Configuratio	n File Supported Dev	vice : TPS65988 E	MV
				Configuratio	n File Name: TPS65	988DH_Advanced	d_v6_1_2
<i>_</i>	d			USB to I2C/	SPI Adapter : FTDI		
	Adjust Register	s	Transmit Sou	Irce Canabilities (0x32)		
	Adjust Register Capabilities	rs	Transmit Sou	urce Capabilities (0x32)		
		'S	Transmit Sou	urce Capabilities(0x32)		
		5	Transmit Sou		0x32)		
		3			0x32)	Value	
		3		DO Config Field	0x32) Use Bank	007027676-0	•
		8	- Tx Source Pl Active PDO	DO Config Field		007027676-0	•
		3	- Tx Source Pl Active PDO	DO Config Field Bank Bank Follows EP		007027676-0	•
		8	Tx Source Provide Active PDO Active PDO Active PDO Bank 0 Settin	DO Config Field Bank Bank Follows EP	Use Bank	007027676-0	•
		3	Tx Source Pl Active PDO Active PDO Bank 0 Settin Number	DO Config Field Bank Bank Follows EP	Use Bank	0	
		3	Tx Source Provide Active PDO Active PDO Active PDO Bank 0 Settin	DO Config Field Bank Bank Follows EP	Use Bank	0	· •
		3	Tx Source Pl Active PDO Active PDO Bank 0 Settin Number	DO Config Field Bank Bank Follows EP ngs of Bank 0 Source PDOs	Use Bank	0	
		3	Tx Source Pl Active PDO Active PDO Bank 0 Settin Number 4	DO Config Field Bank Bank Follows EP ngs of Bank 0 Source PDOs	Use Bank	0	
Transmit Source Raw View		3	Tx Source Pl Active PDO Active PDO Bank 0 Settin Number 4	DO Config Field Bank Bank Follows EP Igs of Bank 0 Source PDOs PDO 1 Field	Use Bank	0 Value es this PDO	

Figure 2-11. Configure Transmit Source Capabilities

- 6. Similarly, make entries for remaining Virtual Addresses referring the pjt file attached with this application.
- 7. Map App Config and GPIO events to respective GPIOs as shown in Figure 2-12.
 - a. Select IO Config Register.
 - b. Map events for each GPIO. Refer to project file for GPIO 2 and 3.



Integrated USB Wall Socket Design

Application Customization Tool	locumente Help				2
eneral Settings Port 1 Settings Port 2 Settings		0x2) Port-1 30W (0x6) I	Port-1 60W (0x7)		
	GUI E	Build Version : 6.1.2			
	Confi	guration File Version : 6.1.2			
Configuration Mode	1970 CON1	guration File Supported Device	e : TPS65988 E	VM	
	Confi	guration File Name: TPS6598	8DH Advanced	1 v6 1	2.tpl
Customer Use		C/SPI Adapter : FTDI			
Interrupt Mask for I2C1	Internal Pull Down Enable				^
Interrupt Mask for I2C2 Global System Configuration	Internal Pull Up Enable Mapped Event	Port 0 Load App Config 1 Event		-	
Port Configuration Port Control	GPIO Polarity	Direct-mapped Event			
Transmit Source Capabilities Transmit Sink Capabilities		In second production			
Autonegotiate Sink	GPIO #1				
Alternate Mode Entry Queue					
Alternate Mode Entry Queue PD3 Configuration Register	Field	Value			
	Field Multiplexing for GPIO 1 pin	Value Pin Multiplexed to GPIO	2	·	
PD3 Configuration Register Transmit Identity Data Object User Alternate Mode Config Display Port Capabilities			;	•	
PD3 Configuration Register Transmit Identity Data Object User Alternate Mode Config	Multiplexing for GPIO 1 pin	Pin Multiplexed to GPIO		•	
PD3 Configuration Register Transmit Identity Data Object User Alternate Mode Config Display Port Capabilities VO Config App Config Binary Data Indices 12C Master Configuration	Multiplexing for GPIO 1 pin Initial Value	Pin Multiplexed to GPIO		•	
PD3 Configuration Register Transmit Identity Data Object User Alternate Mode Config Display Port Capabilities VO Config App Config Binary Data Indices	Multiplexing for GPIO 1 pin Initial Value Open Drain Output Enable	Pin Multiplexed to GPIO		•	
PD3 Configuration Register Transmit Identity Data Object User Alternate Mode Config Display Port Capabilities VO Config App Config Binary Data Indices I2C Master Configuration App configuration Register Sleep Control Register Sleep Control Register Tx Manufacturer Info SOP	Multiplexing for GPIO 1 pin Initial Value Open Drain Output Enable Internal Pull Down Enable	Pin Multiplexed to GPIO		•	
PD3 Configuration Register Transmit Identity Data Object User Alternate Mode Config Display Port Capabilities VO Config App Config Binary Data Indices I2C Master Configuration App configuration Register Sleep Control Register	Multiplexing for GPIO 1 pin Initial Value Open Drain Output Enable Internal Pull Down Enable Internal Pull Up Enable	Pin Multiplexed to GPIO Ox0		•	

Figure 2-12. GPIO Mapping

2.6 Test Setup and Results

The control logic to balance port power (as shown in Figure 2-6) can be tested on TPS65988EVM using the project file. TPS65988EVM has two Buck regulators (LM3489) corresponding to Port A and Port B as shown in Figure 2-13. The GPIO pins GPIO12 and GPIO13 are mapped to output GPIO events *Port 0 Source PDO Negotiated TT1* and *Port 0 Source PDO Negotiated TT2* events, respectively. These GPIOs toggle based on the negotiated PDO and set the feedback voltage of LM3489 accordingly, to generate the negotiated PDO voltage. Similarly, voltage on Port B (Port 1) is controlled by GPIO14 and GPIO15. In the actual application, Buck regulators shall be replaced by ACDC and DCDC regulators as shown in Figure 2-13. TPS65988DH includes three I2C ports out of which one I2C port can work as both master and slave, one I2C port can work as a I2C master only and one can work as a I2C slave only. The I2C masters allow the PD controller to control various kinds of slaves directly based on events inside the PD controller. I2C master events can be used to configure ACDC/DCDC with I2C interface.



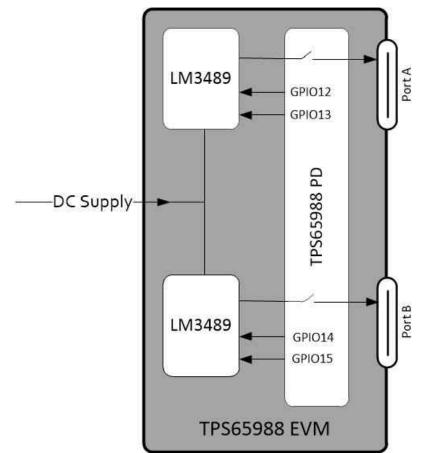
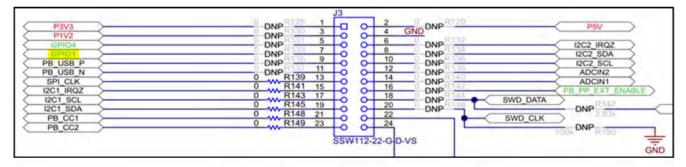


Figure 2-13. EVM Block Diagram

Following are the steps to prepare EVM and then test the port power balancing control logic.

- 1. Populate 0E resistors R170-R173 and R133 to make GPIO0-GPIO3 signals available on signal headers J3 and J4.
- 2. Short GPIO0 (on J4) to GPIO1 (on J3) and GPIO2 (on J4) to GPIO3 (on J4) using jumper wires.





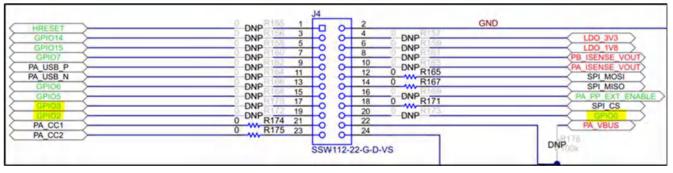


Figure 2-14. GPIO Setting on EVM

- 3. Populate PPHV jumpers to connect buck regulators' outputs to the system side of internal power switches. Short pins 5-7 and 6-8 of J11.
- 4. Power up the EVM by connecting the Power Adapter to barrel Jack connector J1.
- 5. Load the project file in GUI Software tool and load the firmware patch in PD controller.
- 6. Power cycle the EVM by unplugging and plugging the Power Adapter.
- 7. Connect a PD capable Sink device to one of the ports through a PD analyzer. The TPS665988DH Source port must advertise 60W capability; ensure the same by checking the Source Capability message.
- 8. Connect another PD device to the second port and verify that the already connected port advertises modified 30W Source Capability message.

Figure 2-15 shows the PD trace captured on Port 0. At t1 - only Port 0 is connected and therefore the source capability message shows 60W. At t2 - Port 1 is also connected and therefore Port 0 scaled down its power capability to 30W. The modified 30W source cap message transmitted from Port 0 at t2 is also show in the figure.



Role	Message	Data
	E PD	
Source:DFP +	✓ ∲ [2]Source_Cap	SOP H=0x45A1 0x2B01912C 0x0002D12C 0x0004B 2C 0x0006412
Source:DFP 4	BMC	01
Source:DFP +	PD Header	Msgld=2 Msg=Source_Cap ObjCounts=4
Source:DFP +	E PD Data 0	Pos=1 Type=Fixed FixedVol=5.00V MaxCur=3.00A
Source:DFP +	PD Data 1	Pos=2 Type=Fixed FixedVol=9.00V MaxCur=3.00A
Source:DFP 4	E PD Data 2	Pos=3 Type=Fixed FixedVol=15.00V MaxCur=3.00A
Source:DFP +	PD Data 3	Pos=4 Type=Fixed FixedVol=20.00V MaxCur=3.00A
Sink:UFP 1	> + [2]GoodCRC	SOP H=0x0441 CRC=0xAFD6A8A2 EOP
Sink:UFP 1	> 1 [0]Request	SOP H=0x1082 0x4384B12C CRC=0x1C184A36 EOP
Source:DFP \$	> * [0]GoodCPC	SOP H=0x0161 CRC=0x4A38788F EOP
Source:DFP +	> 1 [3]Accept t1	SOP H=0x07A3 CRC=0x5A976876 EOP
Sink:UFP 1	> + [3]GoodCi	SOP H=0x0641 CRC=0x41D8C98E EOP
Source:DFP +	> + [4]PS RDY	SOP H=0x09A6 CRC=0xC058B134 EOP
Sink:UFP &	> + [4]GoodCRC	SOP H=0x0841 CRC=0xA660E489 EOP
Source:DFP +	> * [5]VDM:Discidentity	SOP H=0x1BAF 0xFF00A001 CRC=0x027E7183 EOP
Sink:UFP &	> * [5]GoodCRC	SOP H=0x0A41 CRC=0x486E85A5 EOP
Sink:UFP 1	> + [1]VDM:Discidentity	SOP H=0x428F 0xFF00A041 0xC4000451 0x0000000 0x00000712
Source:DFP +	> + [1]GoodCRC	SOP H=0x0361 CRC=0xA43619A3 EOP
Source:DFP +	> 1 [6]VDM:DiscSVID	SOP H=0x1DAF 0xFF00A002 CRC=0x9F8B2BCD EOP
Sink:UFP 1	> + [6]GoodC t2	SOP H=0x0C41 CRC=0xA10D2090 EOP
Sink:UFP &	> + [2]VDM:DISCSVID	SOP H=0x248F 0xFF00A042 0xFF010000 CRC=0xF9F7E6EB EOP
Source:DFP \$	> + [2]GoodCRC	SOP H=0x0561 CRC=0x4D55BC96 EOP
Source:DFP +	✓	SOP H=0x4FA1 0x2B01912C 0x0002D12C 0x0004B0C8 0x0006409
Source:DFP +	BMC	01
Source:DFP +	PD Header	Msgld=7 Msg=Source Cap ObjCounts=4
Source:DFP +	E PD Data 0	Pos=1 Type=Fixed FixedVol=5.00V MaxCur=3.00A
Source:DFP \$	PD Data 1	Pos=2 Type=Fixed FixedVol=9.00V MaxCur=3.00A
Source:DFP +	PD Data 2	Pos=3 Type=Fixed FixedVol=15.00V MaxCur=2.00A
Source:DFP +	PD Data 3	Pos=4 Type=Fixed FixedVol=20.00V MaxCur=1.50A
Sink:UFP 🕆	> * [7]GoodCRC	SOP H=0x0E41 CRC=0x4F0341BC EOP
Sink:UFP 1	> 1 [3]Request	SOP H=0x1682 0x43825896 CRC=0x11FE5DC1 EOP
Source:DFP +	> + [3]GoodCRC	SOP H=0x0761 CRC=0xA35BDDBA EOP
Source:DFP +	> 1 [0]Accept	SOP H=0x01A3 CRC=0xB3F4CD43 EOP
Sink:UFP 1	> * [0]GoodCRC	SOP H=0x0041 CRC=0xA8BB6CBB EOP
Source:DFP +	> + [1]PS_RDY	SOP H=0x03A6 CRC=0x208D582A EOP
Sink:UFP 1	> * [1]GoodCRC	SOP H=0x0241 CRC=0x46B50D97 EOP

Figure 2-15. PD Trace Showing PDOs Corresponding to Single and Dual Port Connections

Figure 2-16 shows Port-0 (channel 1) and Port-1 (Channel 2) Vbus voltage transitions captured in parallel with the PD trace. Channel 2 shows Port-1 Vbus transition to 20 V.



lek Stop] Nois	e Filter Off
	evbus.	of already connected port	
Port-0 Vbus			
Port-0 Vous			
	Port-1 Vbus		
2		Second port establ without affecting alr	ished 20V contract -
Second port con	nected	without affecting and	eady connected port.
		<u>.</u>	
 5.00 V 5.0 	0 V (100ms	(2) / 4.00 V	14:59:23

Figure 2-16. Scope Capture of Vbus Transitions



3 References

- Texas Instruments, TPS65987D GPIO Events application report.
- Texas Instruments, Using I2C Master in TPS65987D and TPS65988 PD Controllers application report.
- Texas Instruments, Dual Port USB Type-C[™] and USB PD Controller with Integrated Power Switches data sheet.

4 Revision History

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

С	hanges from Revision * (May 2021) to Revision A (December 2023)	Page
•	Updated the numbering format for tables, figures, and cross-references throughout the document	1
•	Added author name	1

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 Copyright © 2023, Texas Instruments Incorporated