

USB Type C Power Delivery Source with TPS65987D and TPS55288

Steffen Graf

ABSTRACT

Contents

1	Introduction	1
2	Implementation	2
3	Test Results	7
4	References	8

List of Figures

1	Enable I2C Master	3
2	Configure I2C Address	4
3	Configure Binary Indices	5
4	Configuring I2C Master Events	6
5	Block Diagram	7
6	Startup behavior	7

List of Tables

Trademarks

All trademarks are the property of their respective owners.

1 Introduction

For USB type C the USB-PD (power delivery) specification is a rather new and powerful extension. The benefits of USB-C are not only the small and reversible connector, but also the wide range of connections that can be substituted by USB-C. Not only USB 2 and USB 3 can be used, but also DisplayPort, Thunderbolt or PCIe. Together with USB-PD a total power of 100 W can also be transferred. This makes it a good interface for smartphones, Laptop PCs and in addition for cameras and industrial PCs.

To achieve the high power levels, USB-C PD supports multiple voltage levels. Beside the well-known standard from legacy USB ports of 5 V any voltage up to 20 V is supported. With USB-PD PPS (Programmable Power Supply) the output voltage of a source can be programmed by the connected sink in 20 mV steps. This enables simplification and heat reduction in the connected sink.

2 Implementation

This document describes the steps that are necessary to implement a USB-PD source with four different voltage levels using TPS65987D as PD controller and TPS55288 as buck-/boost converter.

When implementing a USB-PD source as part of a system, the problem of the right power supply comes up. When the typical voltage levels of 5 V, 9 V, 15 V and 20 V should be supported and a wide input voltage range is needed a buck/boost topology is a good choice to go in that case. To switch the output voltage of any converter different methods can be used. Simple adjustable regulators allow changing the feedback network or injecting a voltage there. Other devices have a communication interface like I2C that allows changing the feedback ratio or the reference voltage to change the output voltage.

Regulators with an internal feedback, that use a resistor for setting the voltage often sample the resistance value only once at power up and cannot be used as there is no way to change the voltage during runtime.

The TPS55288 used here has an I2C interface that allows changing the feedback divider or the reference voltage.

Before explaining the commands that have to be sent to the TPS55288, let's have a closer look at the USB-PD TPS65987D and USB-C.

As mentioned before USB-C PD can support multiple voltages and has a reversible connector. So it is necessary to have some mechanism for negotiating the voltage and detect how it got connected, so the signal can be multiplexed the right way.

Therefore the CC-line (communication channel) is introduced. This line is present two times as CC1 and CC2. These two signals are physically located on the connector in a way so reversing the connector will lead in reversing the CC lines. In normal case, CC1 will connect to CC1 and same with CC2. When connected in reverse, CC1 will connect to CC2 and vice versa.

The USB PD controller detects this and can report this. But on these CC-lines also the negotiation about the power takes place.

When connecting a device a VBus voltage of 5 V gets supplied before any communication takes place over the CC-line. The source then advertises its capabilities (Source PDOs: Power Delivery Objects) and the sink selects one and requests it. When the voltage level is negotiated, the source switches the voltage to the selected level.

The TPS65987D needs a way to signal to a voltage regulator the voltage level that has been negotiated. The successful negotiation creates an internal event, which can be used to trigger actions. These actions can for example be used to set GPIOs or send I2C messages.

2.1 GPIOs for setting voltage

One way to implement this is to map the events for the source PDOs to GPIOs that interact with the voltage regulator. An example implementation of this is done on the TPS65987D evaluation module. Therefore it is not described further here.

2.2 I2C for setting voltage

Another way is setting the configuration through I2C. The TPS65987D can act as I2C master and send out messages at certain events. When using the "Application Customization Tool" there are several steps necessary to set this up.

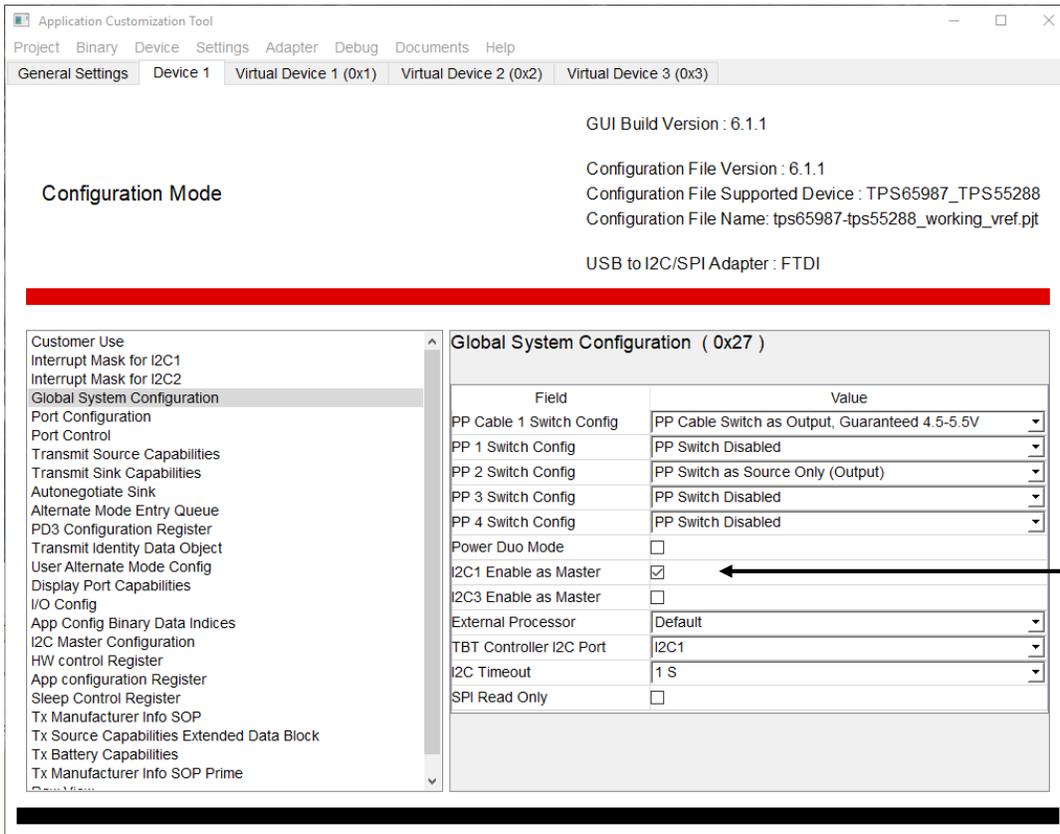


Figure 1. Enable I2C Master

First it is necessary to enable one of the I2C ports as master. This is done through the “Global System Configuration” by enabling “I2C1 Enable as Master”. In [Figure 1](#) it is shown how this can be done in the customization tool, when using I2C1 for communication with a voltage regulator.

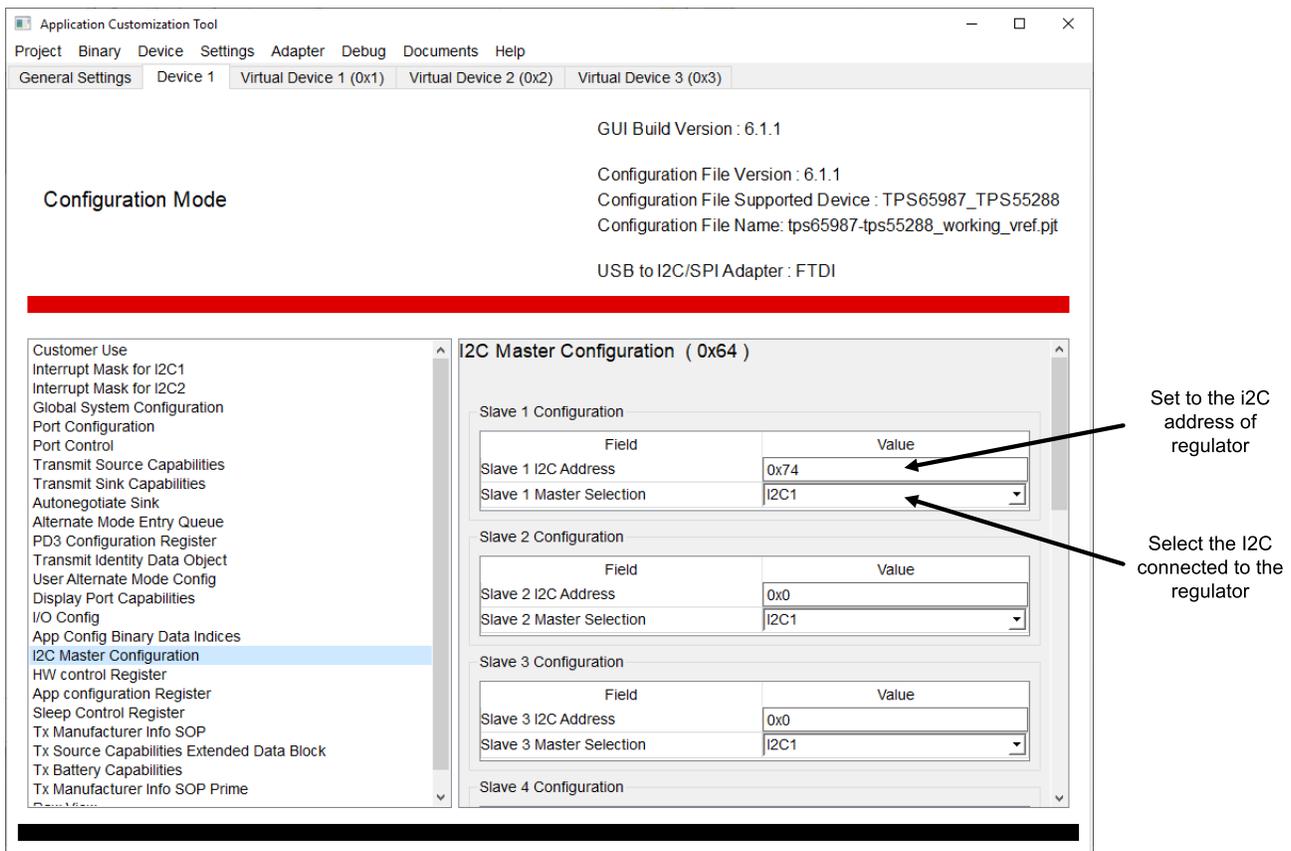


Figure 2. Configure I2C Address

Next necessary step is as shown in [Figure 2](#) to configure the address of the connected slave for the I2C port by setting this in “I2C Master Configuration”. Here for the first I2C slave it has to be configured to use I2C1 and the address of the connected regulator, in this case 0x74.

Application Customization Tool

Project Binary Device Settings Adapter Debug Documents Help

General Settings Device 1 Virtual Device 1 (0x1) Virtual Device 2 (0x2) Virtual Device 3 (0x3)

Configuration Mode

GUI Build Version : 6.1.1
 Configuration File Version : 6.1.1
 Configuration File Supported Device : TPS65987_TPS55288
 Configuration File Name : tps65987-tps55288_working_vref.pjt
 USB to I2C/SPI Adapter : FTDI

Field	Value
Country Codes Start Index	0
Country Codes Number of Indices	0
Common I2C Record Start Index	0
Common I2C Record Number of Indices	0
Port 1 I2C Record Start Index	1
Port 1 I2C Record Number of Indices	8
Port 2 I2C Record Start Index	0
Port 2 I2C Record Number of Indices	0

Customer Use
 Interrupt Mask for I2C1
 Interrupt Mask for I2C2
 Global System Configuration
 Port Configuration
 Port Control
 Transmit Source Capabilities
 Transmit Sink Capabilities
 Autonegotiate Sink
 Alternate Mode Entry Queue
 PD3 Configuration Register
 Transmit Identity Data Object
 User Alternate Mode Config
 Display Port Capabilities
 I/O Config
App Config Binary Data Indices
 I2C Master Configuration
 HW control Register
 App configuration Register
 Sleep Control Register
 Tx Manufacturer Info SOP
 Tx Source Capabilities Extended Data Block
 Tx Battery Capabilities
 Tx Manufacturer Info SOP Prime

Annotations:
 - First index in I2C Master Events table (points to Port 1 I2C Record Start Index = 1)
 - Number of entries in table (points to Port 1 I2C Record Number of Indices = 8)

Figure 3. Configure Binary Indices

In the “App Config Binary Data Indices” shown in [Figure 3](#) it is needed to set the “Port 1 I2C Record Start Index” to the first index used (here it is one) and the “Number of Indices” to the amount of configured I2C messages. Make sure to set useful and correct values here, otherwise it might not work as expected.

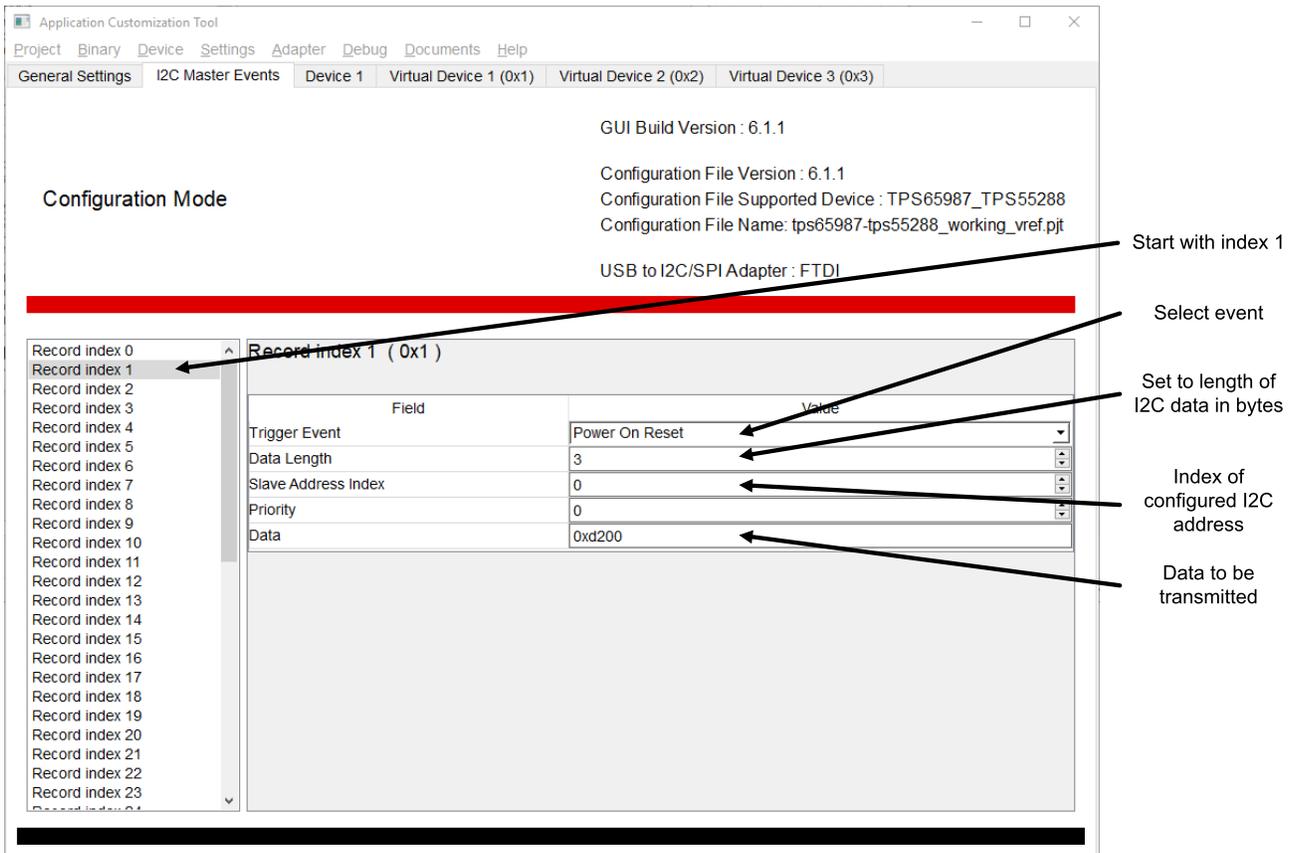


Figure 4. Configuring I2C Master Events

With this the data that has to be sent out can be configured in the “I2C Master Events” tab, this tab is accessible from Settings -> Show I2C Master Events Table. In [Figure 4](#) this screen is shown, make sure to start with index one and set the right data length and address. The TPS55288 needs some initial configuration after power up. It is possible to use the “Power On Reset” event for initial configuration of the buck-/boost regulator. If multiple messages have to be sent out, simply use several indices with the same event. The messages will then be sent out one after another. The first configuration is shown in the screenshot above, the others have to be added in similar manner in the next indices.

For the TPS55288 three messages are defined with the “Power On Reset” event:

- Setting V_{Ref} to 0.282 mV for 5 V output, Data 0xd200 with a length of 3
- Selecting the divider 0.0564, Data 0x0304, Length 2
- Enabling the output, Data 0xa006, Length 2

This will make sure that the TPS55288 always starts up with 5 V at the output. Next step is to configure the different voltages. The supported voltages that are advertised are configured in “Transmit Source Capabilities” and will trigger a “Source PDO x Negotiated” event.

These events have to be configured in the “I2C Master Events”. In this example the V_{Ref} is changed according to the selected PDO:

- Source PDO 1 (Set to 5 V): Data 0xd200, Length 3
- Source PDO 2 (9 V): Data 0x19a00, Length 3
- Source PDO 3 (15 V): Data 0x2c500, Length 3
- Source PDO 4 (20 V): Data 0x3bf00, Length 3

With this configured, the TPS65987D will change the output voltage as negotiated when connecting a power sink. In addition to that, one more event should be configured to make sure, that the voltage will be at 5 V in the beginning. Therefore the detach event should send the same data as the PDO 1 event. Otherwise the output of the TPS55288 might be at higher voltage when connecting a sink.

3 Test Results

The overall test setup consisting of the TPS55288 and the TPS65987D EVM is shown in [Figure 5](#).

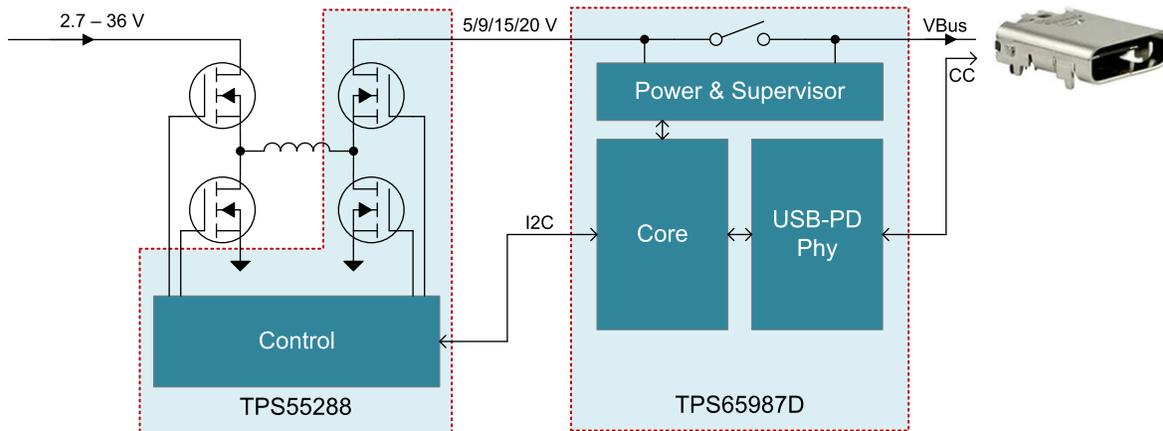


Figure 5. Block Diagram

This setup allows easy evaluation of this combination.

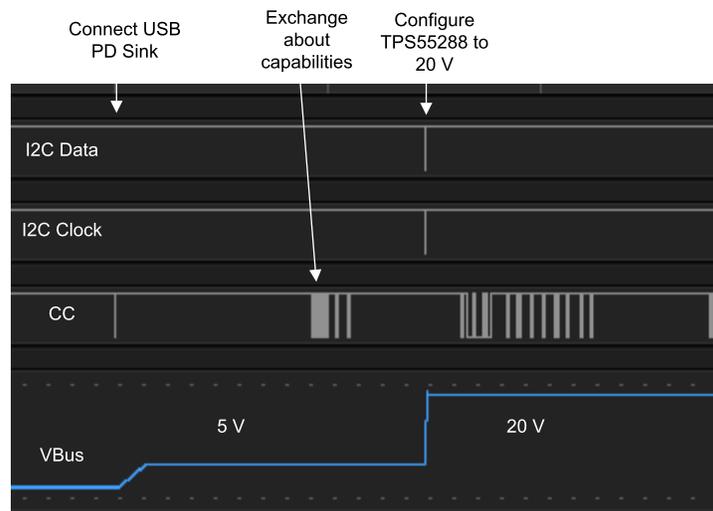


Figure 6. Startup behavior

In [Figure 6](#) is shown how the system behaves when a device gets connected. It can be seen before a device is connected, the V_{BUS} line is unpowered. With a first reaction on the CC line, the voltage on V_{BUS} gets enabled but is still at 5 V. During the voltage is at 5 V, communication on the CC line happens. After this exchange about capabilities and negotiating PDO 4 with 20 V, the TPS65987D sends out the configured message to switch the voltage to 20 V. It can be seen that the VBus voltage increases to 20 V.

4 References

- [USB Type-C and power delivery 101 – Power delivery protocol](#)
- [I2C Master: Using I2C Master in TPS65987D and TPS65988 PD Controllers](#)
- [TRM: TPS65987DDH and TPS65988DH Host Interface Technical Reference Manual](#)
- [EVM: TPS65987 Evaluation Module](#)
- [DS: TPS65987D USB Type-C and USB PD Controller with Integrated Power Switches](#)

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 © 2022, Texas Instruments Incorporated