Power Sharing algorithm for USB Type-C™ Power Delivery dual port adapter

Introduction

USB Type-C™ Power Delivery specification has been updated with the extension of the output range from standard power range (SPR), up to 100 W, to extended power range (EPR), up to 240 W, allowing a wide flexibility of output variation to any compliant solution. This implies that a single port solution, which supports a wide output power range, needs an effective management of the internal power. It also needs a certain amount of input power to provide the maximum number of PDOs to the connected devices. The management of the internal power is crucial as it must balance the power fixed at the input and, for multiport solutions, dynamically distribute it among different output stages. So, it is fundamental to identify a strategy to balance the power managed by the solution and share it among the ports, according to the requests received by the attached USB PD Sink devices.

This technical note describes an algorithmic method designed for a USB Type-C™ Power Delivery dual port solution to share and optimize the power between the two ports. It aims at exposing the maximum available power, taking into account the input power and the requests from the device attached to the other port.

The algorithm is embedded as software IP in the STSW-2STPD01 software package. It is programmed to run on the STEVAL-2STPD01 USB Type-C™ Power Delivery dual port adapter kit and to demonstrate the effectiveness of the power-sharing strategy in the application.

RELATED LINKS

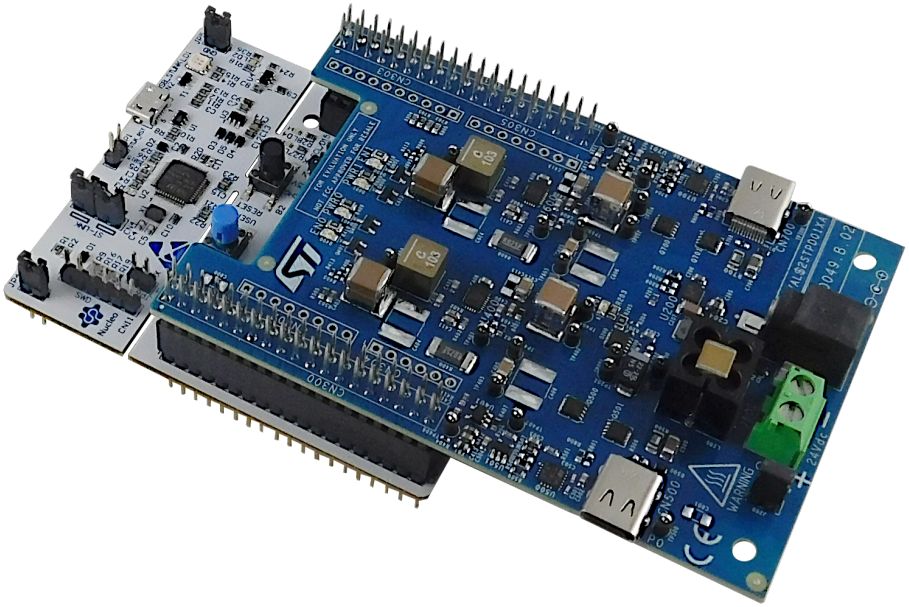
TA0357: Overview of USB Type-C and Power Delivery technologies

1 Hardware setup

The STEVAL-2STPD01 USB Type-C™ Power Delivery dual port adapter kit is the platform suitable to demonstrate the efficiency of the Power Sharing IP.

It is an evaluation kit composed of the NUCLEO-G071RB STM32 Nucleo-64 development board and an expansion board integrating two STPD01 programmable buck converters for USB Power Delivery as well as two TCPP02-M18 USB Type-C™ protections for source application, offering two Type-C ports.

Figure 1. STEVAL-2STPD01 evaluation kit



The kit exploits the characteristics of the STPD01PUR buck regulator programmable, controlled through an I²C interface and a UCPD peripheral embedded in the STM32G071RBT6 microcontroller with a dedicated firmware stack, to implement a USB Type-C and PD dual port source adapter with power-sharing capability, providing the voltages required by the systems compliant with the USB PD specification.

The STEVAL-2STPD01 evaluation kit can provide up to 120 W (max. 60 W for each port), to the USB PD Sink devices attached to its ports. It is able to expose all the possible PDOs (5 V, 9 V, 15 V, and 20 V at 3 A maximum current) on both ports, when its input is connected to a DC power supply able to furnish more than 144 W, with a V_{IN} > 21 V.

When a DC supply with lower power is connected to the input, the solution can expose a small set of PDOs.

The Power Sharing algorithm acquires the power supply data to optimize and distribute the power between the two ports, without exceeding the maximum power furnished by the power supply.

The user can modify a set of module parameters, such as boundaries of voltage and current, power, efficiency ratio, etc., to change the operating conditions.

The algorithm characteristics and functionalities, and the application example where the IP is integrated have been developed in compliance with the Type-C and Power Delivery specification.

The algorithm has been included as a compiled module in the STSW-2STPD01 software package that contains the application, source code, and libraries designed to demonstrate the capabilities of the STEVAL-2STPD01.

Note: For further information on the STEVAL-2STPD01, see the hardware and software user manuals (UM2880 and UM2882 freely available on www.st.com).

2 Algorithm overview

The algorithm takes into account the input power. It operates when the nominal input power is not enough to provide the widest output PDO range to each port (that is P_{IN} < 144 W with 21 V at least).

When launched, the algorithm configures each port to expose the set of PDOs that correspond to the maximum available power (15 W to 60 W). For every USB PD event occurred on one port, the algorithm adapts the power available in the other port by changing the available PDO set.

To let the user set the power budget available as input from the power supply, the IP includes a mechanism that exploits the serial communication of the embedded STM32 microcontroller to save the data on its memory. You can insert the characteristics of the input power supply (nominal power as voltage, current pair) through STM32CubeMonitor-UCPD GUI. Starting from these characteristics and after fixing the system efficiency, the algorithm calculates the disposable power to provide to each port. According to this disposable power, the algorithm generates a new list of PDOs in line with the power balance and presents it to the connected devices.

Note: Refer to UM2880, freely available on www.st.com, for further details on how to set the input power with STM32CubeMonitor-UCPD GUI.

The Power Sharing algorithm is composed of three main parts identified by the following specific functions:

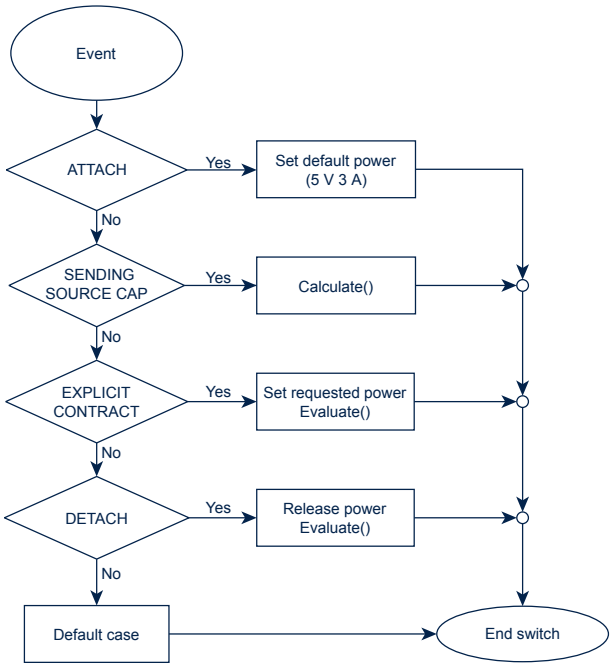
- **Event Notification**: to gather the occurred events and consequently update the power-sharing status parameters, before calling the **Calculate** or **Evaluate** functions, or both;
- **Calculate**: to compute a valid PDO set for the target port according to the input power supply and the status of the other port;
- **Evaluate**: to evaluate if it is necessary to notify the updated PDO list to the other port.

Note: The power sharing algorithm keeps track of the solution power efficiency. In order to calculate the amount of the actual power available to each port, it subtracts the amount of power the board consumes from the input power.

2.1 Event Notification function

The Event Notification function is the entry point of the power algorithm since it drives the solution on the basis of the USB PD stack messages to which it is latched to. The following figure describes the flow diagram of the Event Notification function.

Figure 2. Event Notification flow diagram



The Event Notification is synchronized on four main events, to address the instances that might change the power balance.

1. When the algorithm detects an Attach event, it reserves 15 W (the default PDO is 5 V, 3 A) of the initial power budget, waiting for further messages that might vary the port status.
2. Latching on the Source Capabilities message sent before, the Event Notification calls the Calculate function to define the power to be presented by a port as PDO list.
3. When an Explicit Contract is detected, the Event Notification calls the Evaluate function to estimate whether the remaining power is sufficient to enable the other port.
4. On a Detach event, the Event Notification calls the Evaluate function to update the balance, after the port has released the engaged power.

2.2 Calculate function

Before dispatching the Source Capabilities message to a connected Sink, the Power Sharing algorithm must identify the quantity of power available in the system and the maximum voltage supplying the solution to assess the PDOs list to expose to each port.

This goal is achieved through the Calculate function which allows calculating the complete power balance of the solution, that is the system must know the input power supply characteristics and the power value that the other port contracts, in terms of voltage and maximum current, when a Sink is connected.

This calculation is repeated every time a new Source Capabilities message has to be provided.

2.3 Evaluate function

The Evaluate function is called only when an Explicit Contract is reached or a Detach event occurs, as for these two events a variation of the power requested by the system can be verified.

This function decides to generate or not a Source Capabilities message on the basis of the information on the power balance provided by the system.

When a new power request is generated in a port through one of the events mentioned above, there is a variation of disposable power for the other port. This might imply a change in PDOs, which consists in sending a different PDO list to the connected Sink.

After the first Explicit Contract occurrence (that is, nothing has been attached to both ports until that moment), the Evaluate function calculates the reduction of the available power and saves the data in the microcontroller memory. In the subsequent new negotiations, the other port offers the remaining power through a regenerated PDO list.

In case a disposable power variation occurs, the Evaluate function calculates the difference between the input power and the new power employed. If this difference is bigger than the maximum power deliverable by a port (60 W for the STEVAL-2STPD01), it means that the available power was greater than the maximum one even before the new request, so all the possible PDOs were already available to the other port. This implies no change is applied to the PDO list for the other port and no Source Capabilities message will be sent.

In case a Detach event occurs, the function estimates the variation of the available power. The disposable power estimated by the Evaluate function is saved and offered to the other port: this could lead to an increase of the power expanding the number of PDOs or of the maximum current in the PDOs where it was limited.

The algorithm can also deactivate a port, if the maximum disposal power is lower than the minimum PDO (5 V, 3 A for the STEVAL-2STPD01).

3 Applicative examples

The following paragraphs describe three applicative cases of the STEVAL-2STPD01 USB Type-C™ Power Delivery Dual Port adapter running the Power Sharing algorithm:

1. when the input is connected to a DC power supply able to supply 144 W (24 V and 6 A);
2. when it is supplied by a limited DC power of 90 W (19 V and 4.74 A);
3. when the input power is the lowest one, that is 18.5 W (8.1 V and 2.3 A).

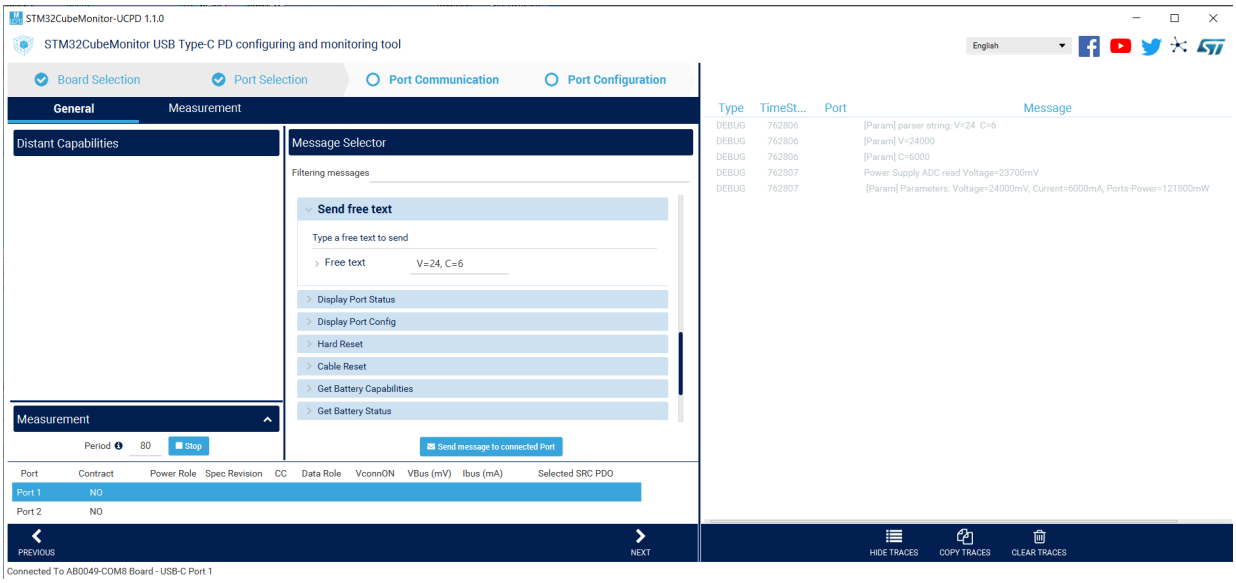
These examples use the *TRACES* window of the STM32CubeMonitor-UCPD GUI v1.1.x to monitor the Source Capabilities messages sent by each port of the solution. The *TRACES* messages report how the power assigned to a port (and therefore the PDO list) changes after each negotiation.

*Note: Refer to UM2880 for further details on how to set correctly the input power by using the **Send free text** function of the STM32CubeMonitor-UCPD GUI.*

3.1 Power Sharing IP operating when the input power is 144 W

When connecting a 144 W DC power supply to the STEVAL-2STPD01, use the STM32CubeMonitor-UCPD GUI to set the input power data at 24 V, 6 A, as shown below.

Figure 3. Setting the input power data at 24 V, 6 A

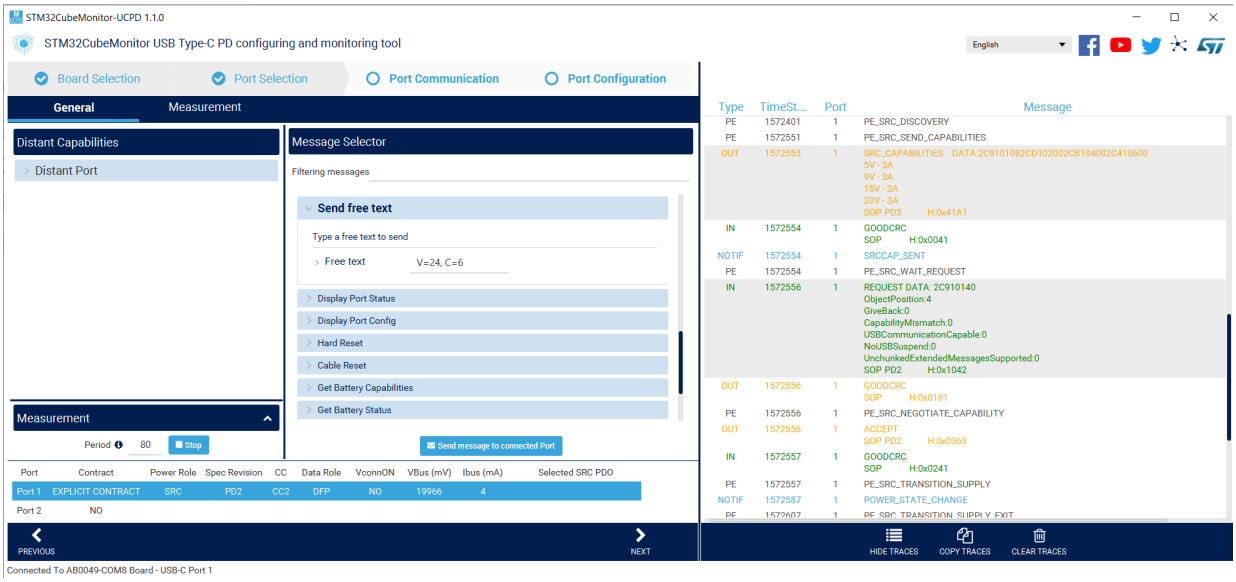


After setting the input, the *TRACES* window confirms the inserted input values and returns the maximum power available for both ports (indicated in Figure 3 as *Ports-Power = 121800 mW*), net of the power dissipated by the board.

Supplying the board with a power supply equal or higher than 144 W guarantees the maximum power capability to each port (60 W), and the algorithm does not operate any reduction of the PDO list in neither port (see also UM2880).

After connecting a Sink platform to the STEVAL-2STPD01 port P0 (named Port 1 in the GUI), the *TRACES* window reports the USBPD messages exchanged by the two entities (Figure 4). At the same time, the Source Capabilities message, sent by the STEVAL-2STPD01 to the Sink, reports all the available PDOs (from 5 V, 3 A to 20 V, 3 A).

Figure 4. Source Capabilities on P0 (Port 1 for the GUI) exposing all PDOs (60 W max capability)



The figure above shows that the first Sink connected to port P0 selects the fourth PDO at 20 V 3 A.

Connecting another Sink to port P1 (named Port 2 in the GUI), the Source Capabilities message, sent by the STEVAL-2STPD01, contains all the available PDOs related to this port (Figure 5). In particular, the Sink connected to this second port initially requests PDO 1 at 5 V 3 A (Figure 5). Later on, it sends a second request for PDO 2 at 9 V 3 A (Figure 6).

Figure 5. Source Capabilities on Port 2 exposing all PDOs (60 W max capability)

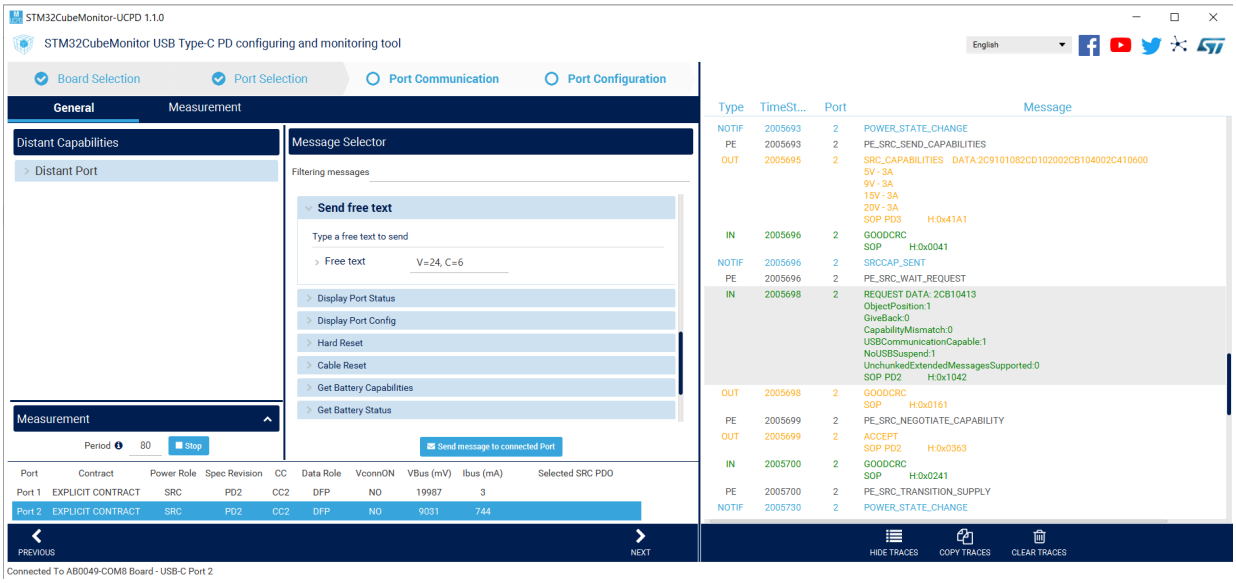
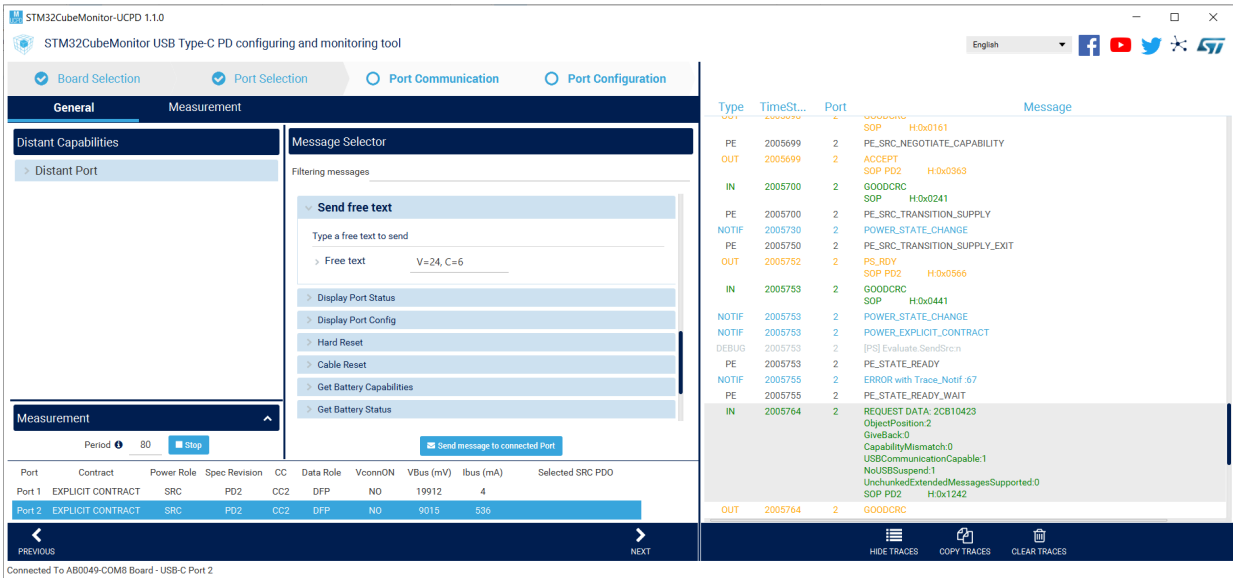


Figure 6. Sink contracting PDO2 on the second port



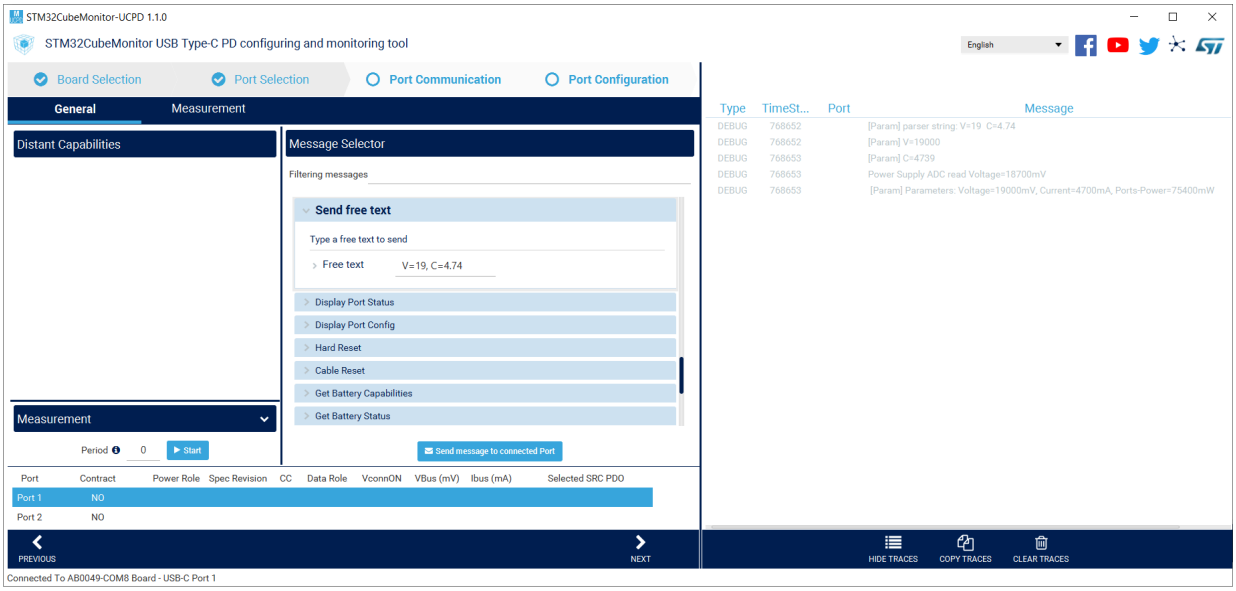
At the end of these operations, both ports reach stable negotiations, and the related Sink platforms are charging.

Thus, when the input power is sufficient to guarantee both ports to operate at the maximum power capability (60 W each), the Power Sharing algorithm is not enabled.

3.2 Power Sharing IP operating when the input power is 90 W

When connecting a 90 W DC power supply to the STEVAL-2STPD01, use the STM32CubeMonitor-UCPD GUI to set the input power data at 19 V, 4.74 A, as shown below.

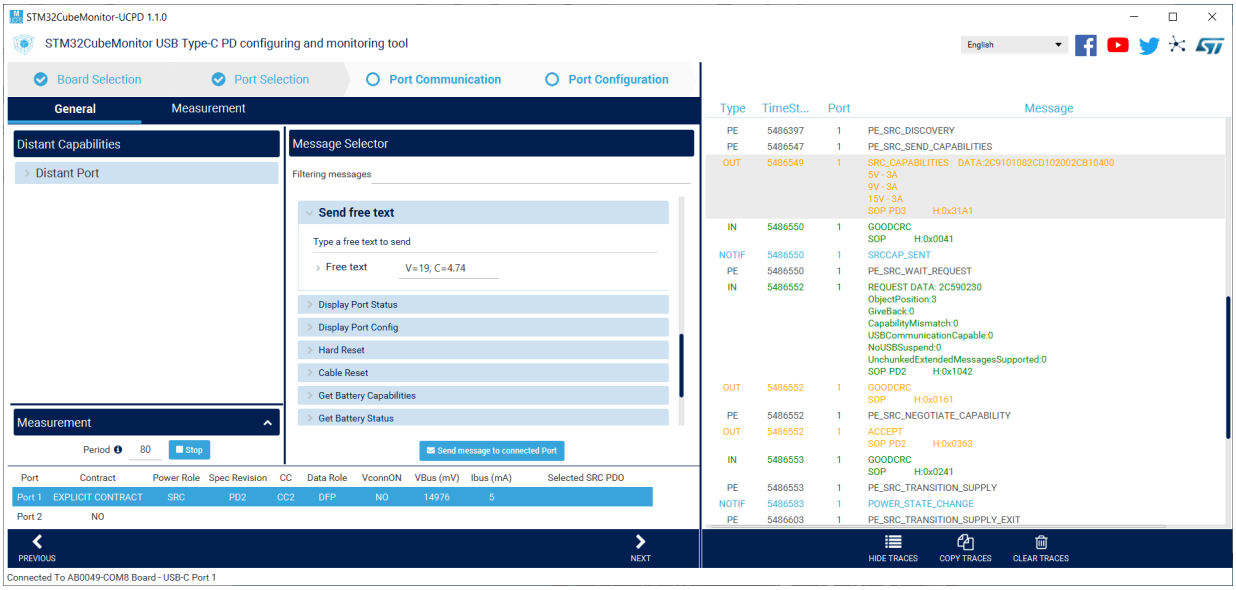
Figure 7. Setting the input power data at 19 V, 4.74 A



In this second example, the *TRACES* window confirms the inserted input values and returns the value of the maximum power calculated by the algorithm (75400 mW) that must be shared between the two ports: the characteristics of the algorithm perfectly match the operative conditions of this second use case.

Since the total available power is minor than the one provided in the previous case, the algorithm enables the ports to present, in the Source Capabilities message, only the PDO profiles in line with that value. Connecting a Sink platform to the STEVAL-2STPD01 port P0, the *TRACES* window reports that the Source Capabilities message sent to the Sink presents one PDO less than the previous case: 5 V 3 A, 9 V 3 A and 15 V 3 A (Figure 8). This is the consequence of a lower input voltage that it does not allow to provide the higher PDO.

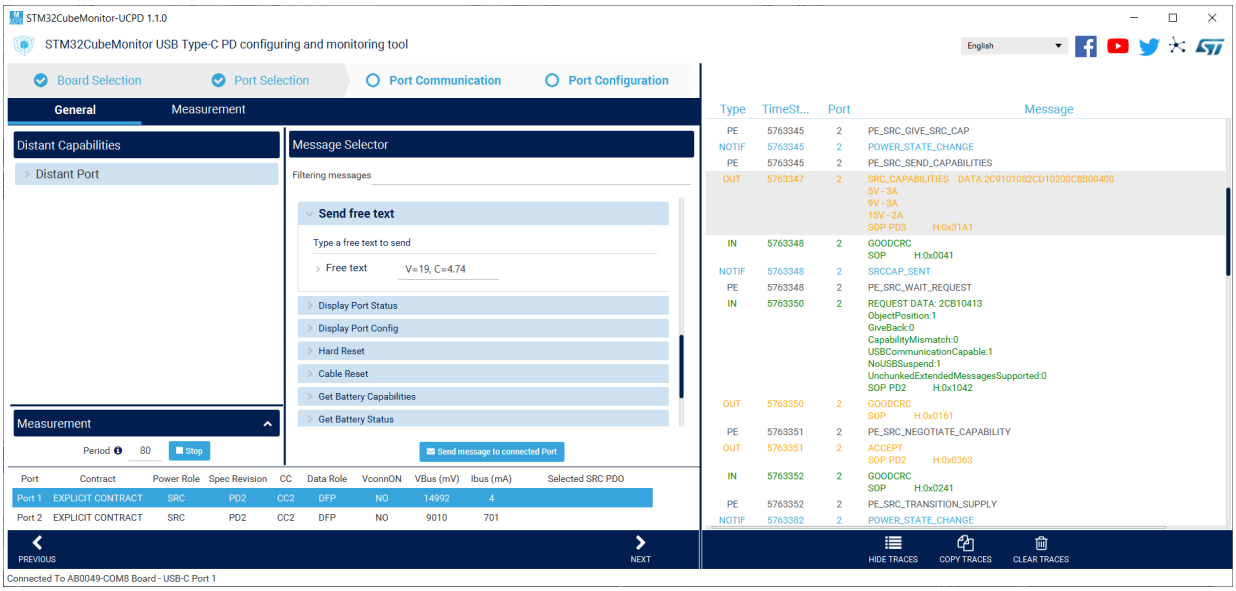
Figure 8. Source Capabilities on Port P0 (Port 1 for the GUI) exposing all PDOs



After the first Sink (attached to Port P0) requests the 45 W PDO (15 V 3 A), the available power for the other port (Port P1) is reduced (lower than 30 W). So, when another Sink is attached on the second port, the algorithm elaborates a new PDO list taking into consideration the remaining power available for this port.

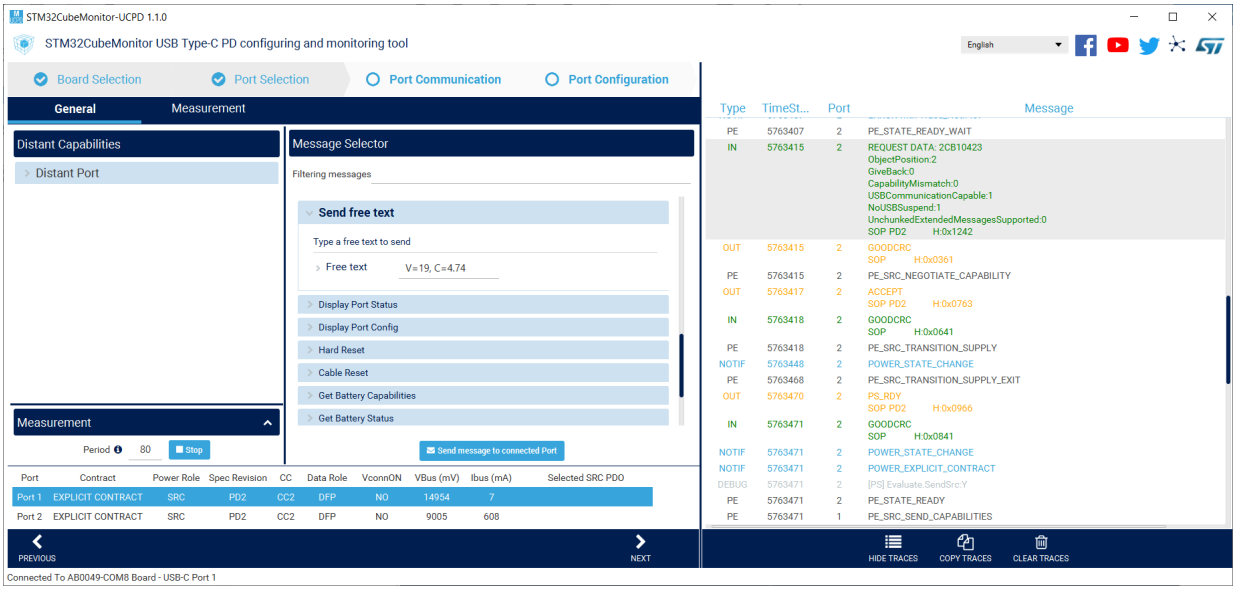
As shown in the figure below, the Source Capabilities message, sent by the solution to the Sink connected to the second port, has the third PDO as 15 V 1.8 A (not 3 A as before), as a consequence of the reduction of the available power (max. 27 W).

Figure 9. Source Capabilities on Port P1 (Port 2 for the GUI) exposing all PDOs, with Sink negotiating PDO1



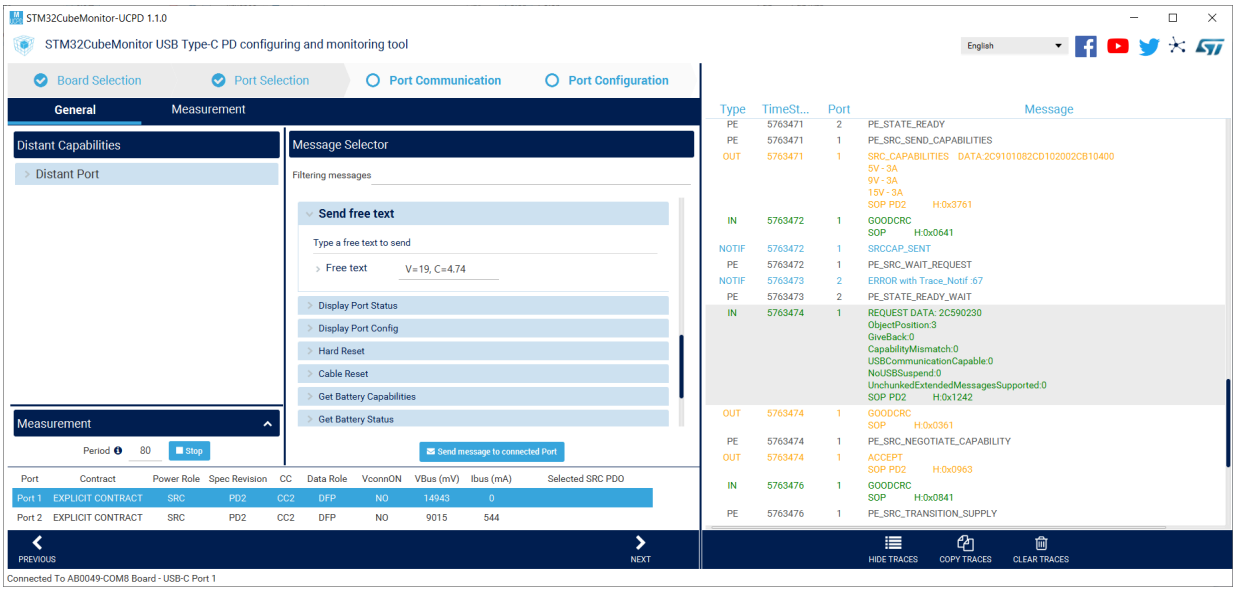
In this specific case, the Sink first negotiates the PDO1 at 5 V 3 A (Figure 9) and then re-negotiates for the higher PDO2 at 9 V 3 A (Figure 10).

Figure 10. Sink on Port P1 (Port 2 for the GUI) negotiating PDO2



After the Sink on the second port reaches the negotiation stability, the algorithm updates the Sink connected to first port with a new Source Capabilities message that reports the remaining power. This message aims at providing a further opportunity to the first Sink to improve its charging condition. In this specific case, the Sink confirms the selection of PDO3 (15 V 3 A), as shown in the figure below.

Figure 11. Source Capabilities on Port P0 (Port 1 for the GUI) with Sink confirming the 15 V 3 A PDO

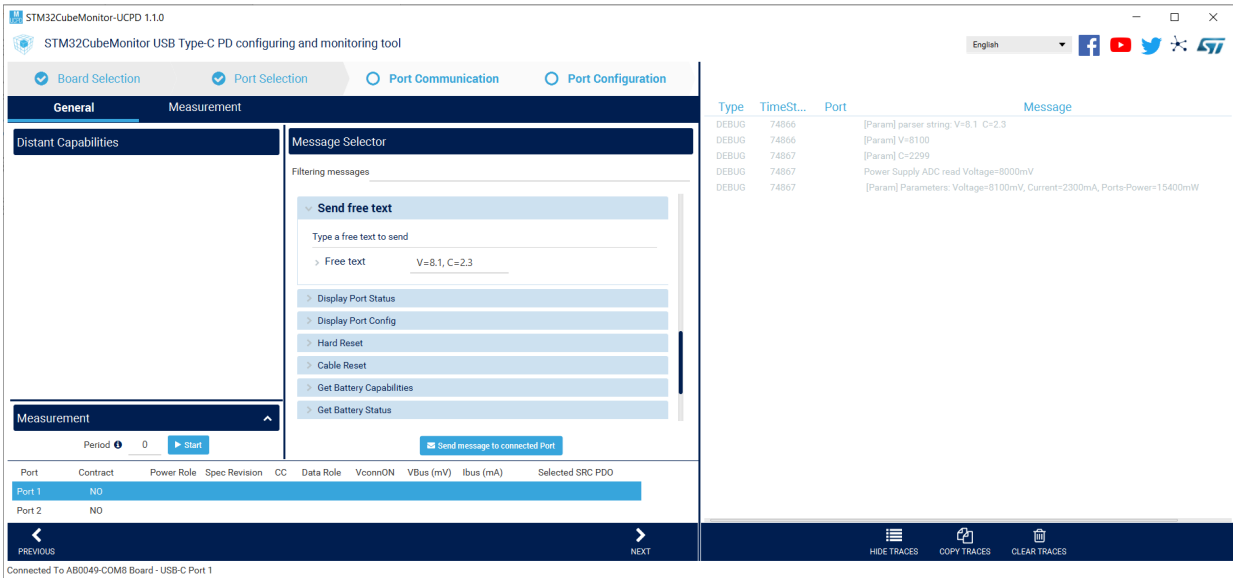


3.3 Power Sharing IP operating when the input power is 18.5 W

This is the worst possible use case in terms of operative condition, since it refers to the minimum input power.

When connecting a 18.5 W power supply to the STEVAL-2STPD01, use the STM32CubeMonitor-UCPD GUI to set the input power data at 8.1 V, 2.3 A, as shown below.

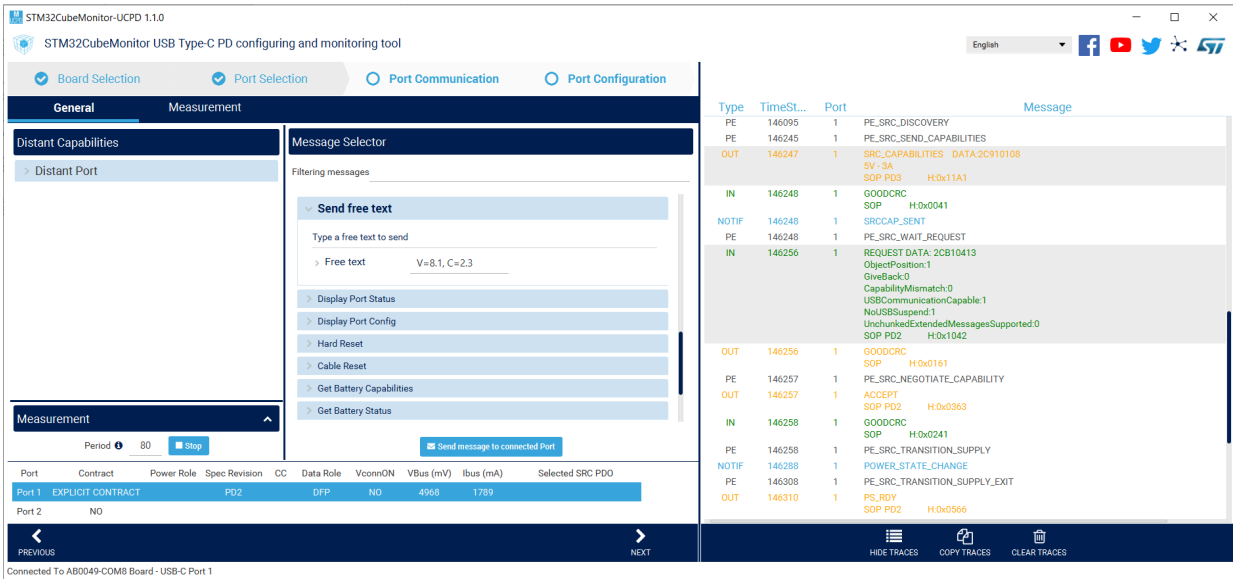
Figure 12. Setting the input power data at 8.1 V, 2.3 A



The TRACES window confirms the inserted input values, and the algorithm calculates a value of 15400 mW as the maximum available power to be shared by the two ports: this value permits the solution to expose only one power profile of 15 W (5 V 3A) that the algorithm addresses to the port that has the first attach event.

Indeed, when a sink is connected to the Port P0 (Port 1 for the GUI), the solution sends the Source Capabilities message composed only of the 5 V 3 A PDO, as shown in the figure below.

Figure 13. Source Capabilities on Port P0 (Port 1 for the GUI) and Sink request of 5 V 3 A PDO



When the Sink concludes the PDO1 negotiation, the algorithm disables the other port as the remaining power is not sufficient to generate any further PDO in line with the USB PD specification power rules.

Note: For any input power value, if the residual power on the solution is lower than 15 W, after successfully reaching the explicit contract on the first port, the algorithm disables the remaining port.

4 References

- Universal Serial Bus Type-C Cable and Connector Specification R2.1 (May 2021)
- Universal Serial Bus Power Delivery Specification R3.1 (May 2021)
- Freely available on www.st.com:
  - TA0357: Overview of USB Type-C and Power Delivery technologies
  - UM2880: "Getting started with the STEVAL-2STPD01 USB Type-C™ Power Delivery dual port adapter based on the STPD01 programmable buck converter"
  - UM2882: "Getting started with the software package for the STEVAL-2STPD01 USB Type-C™ Power Delivery dual port adapter kit"

Revision history

Table 1. **Document revision history**

Date	Revision	Changes
06-Dec-2021	1	Initial release.

Contents

List of tables

List of figures