Getting Started with EZ-PD™ CCG4

About this document

Associated Part Family

CYPD4x25

Associated Software

EZ-PD™ CCGx SDK

Scope and purpose

AN210771 introduces the dual USB Type-C EZ-PD™ CCG4 controller. It helps to understand the CCG4 architecture, evaluation kit, and development tools. It explains how to configure and modify the CCG4 firmware to develop a Dual Role Port (DRP) application such as a Type-C notebook. This application note also helps to find CCG4 resources to accelerate in-depth learning about CCG4.

Table of contents

1 Introduction

CCG4 is afourth-generation Type-C and PD controller, which includes two Type-C transceivers and termination resistors, Rp and Rd. CCG4 has integrated VCONN FETs, 128 KB of flash memory, 8 KB of SRAM memory, overvoltage and overcurrent protection, and 15-kV system-level ESD protection. CCG4 provides a complete solution for dual Type-C port notebook and power adapter designs.

This application note presents the basics of CCG4 including CCG4 features, and an overview of firmware development and debugging tools. It also demonstrates the basic DRP capability of CCG4 and provides a step by step guided tour of CCG4's DRP application design process using development and debugging tools.

1.1 Target Applications

The CCG4 device can be used in the following applications:

- Notebooks
- Power adapters
- Docking stations
- Power banks
- Monitors

1.2 EZ-PD CCG4 Features

As shown in **Figure 1**, the CCG4 device features include:

- 32-bit CPU subsystem
  - 48-MHz Arm™ Cortex™-M0 CPU
  - 128 KB of flash and 8 KB of SRAM
- Integrated digital blocks
  - Four integrated TCPWM blocks, which can each be configured as a timer, counter, or pulse width modulator
  - Four configurable serial communication blocks (I^{2}C/UART/SPI)
- Type-C support
  - Two integrated transceivers (baseband PHY) and support for two USB Type-C ports
  - Integrated UFP (R_D) and current sources for DFP (R_P) for both Type-C ports
  - Integrated dead battery detection for DRP applications
- Low power operation
  - 2.7-V to 5.5-V operation
  - Integrated VCONN FETs to power EMCA cables
  - Independent supply voltage pin for GPIO, which allows 1.71-V to 5.5-V I/O signaling
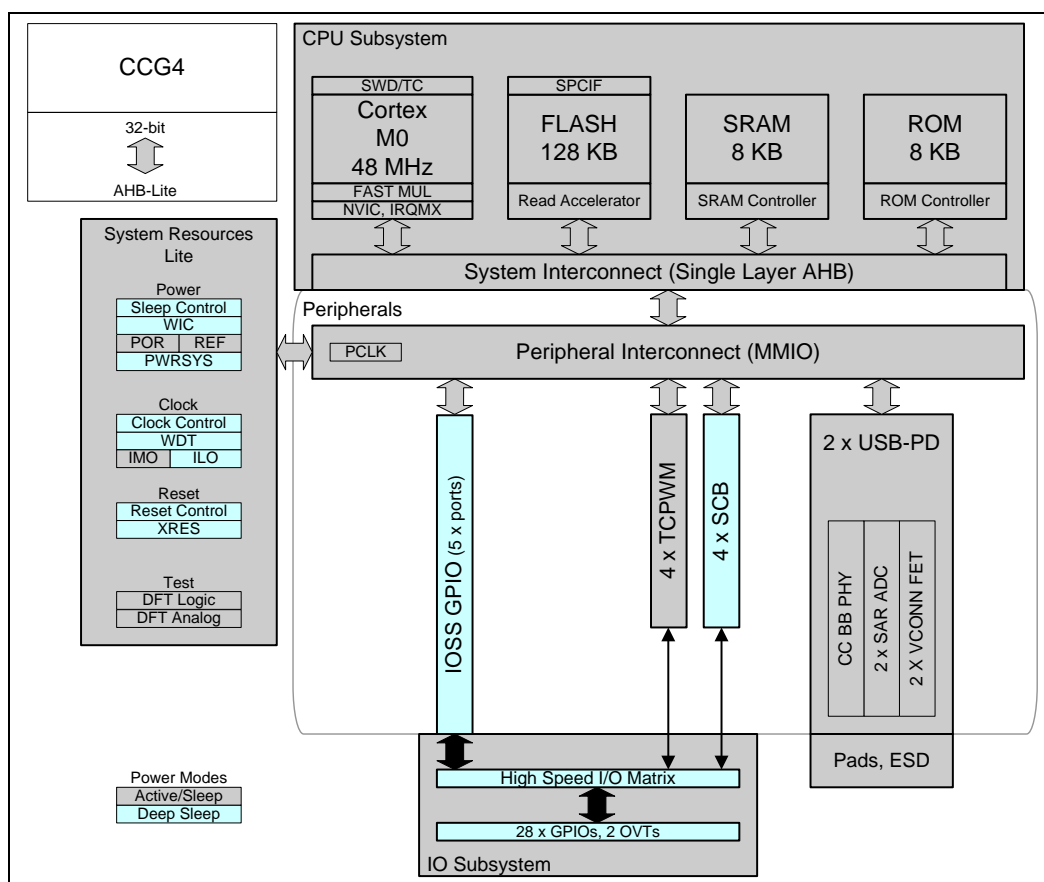 - DeepSleep current: 2.5 uA, Sleep current: 2.5 mA

Introduction

Figure 1   CCG4 Architecture

For in-depth information, see the **CCG4 datasheet**.

1.3   Prerequisites

Before getting started, the following hardware and software are required.

1.3.1   Hardware

- **CY4541 EZ-PD CCG4 Evaluation Kit (EVK)**
- PC/laptop with Windows XP (SP2 or later) [PC with at least two USB ports (USB 3.0 ports and Windows 7 or later is recommended)]
- **Type-C power adapter** that supports a 14 V or higher power profile

Note that the CCG4 device works with all Type-C power adapters supporting power profiles from 5 V to 20 V. This power profile requirement relates to the CY4541 CCG4 EVK architecture. The entire CY4541 CCG4 EVK is powered only when a Type-C power adapter that supports a 14 V or higher power profile is connected. (This is optional. It is required to test only Dead battery condition).

- **MiniProg3** (This is optional and is required only to program the .hex file format firmware)
- **CY4500 EZ-PD Analyzer Kit** (This is optional and is required only to debug the firmware)

1.3.2   Software

- **EZ-PD CCGx Software Development Kit (SDK)**
- **EZ-PD Configuration Utility**
- **PSoC Creator™ 3.3 SP1 or later** with **PSoC Programmer 3.24 or later** (only required if modifying the base firmware functionality)

2 CCG4 Resources

CCG4 design resources include datasheets, application notes, evaluation kits, reference designs, and firmware development and debugging tools. **Table 1** summarizes the resources.

Table 1 **CCG4 Design Resources**

Category	Available Resources	Where to Find Resources
Datasheet	CCG4 datasheet	**CCG4 Datasheet**
Hardware	Evaluation kit – schematic, board files, and documentation	**CY4541 CCG4 EVK**
Application Notes	Hardware design guidelines including recommendations for resistors, decoupling capacitors for power supplies, and PCB layout	**AN210403**
Programming Specifications Guide	The programming reference manual gives information necessary to program the nonvolatile memory of the CYPD4xxx devices	**Programming Specifications Guide**
Host PC Software	Software Development Kit	**EZ-PD CCGx SDK**
	GUI-based Windows application to help users configure the CCG4 controller	**EZ-PD Configuration Utility**
	Firmware development tool	**PSoC Creator 3.3 SP1 or later**
	Firmware programming tool	**PSoC Programmer 3.24 or later**
Debugging Tools	CY4500 EZ-PD Analyzer -- schematic, board files, documentation, and the EZ-PD Analyzer Tool	**CY4500 EZ-PD Analyzer**
Other Collaterals	Knowledge base articles	**Knowledge Base Articles for CCG4 Controller**

3 EZ-PD CCGx Portfolio

Infineon offers a portfolio of USB Type-C and Power Delivery (PD) controllers, including EZ-PD CCG1, EZ-PD CCG2, EZ-PD CCG3, and EZ-PD CCG4. These Type-C PD controllers are fully compliant with the **USB Power Delivery (PD) Specification Revision 2.0, Version 1.1** and **USB Type-C Cable and Connector Specification Revision 1.1** standards. **Table 2** describes the differences between these controllers.

The **USB Power Delivery (PD) Specification Revision 2.0, Version 1.1** defines power delivery up to 100 W (20 V at 5 A) over USB Type-C connector. The **USB Type-C Cable and Connector Specification Revision 1.1** details a new reversible and sub-3-mm, slim, connector design that supports 100 W of power along with USB and non-USB signals (for example, DisplayPort). Refer to **Introduction to Type-C videos** and **AN96527** for information on USB PD and Type-C protocols.

Table 2   Feature Comparison of USB Type-C and PD Controllers

Features	CCG1	CCG2	CCG3	CCG4
Number of Type-C and PD ports	1	1	1	2
Integrated ARM Cortex-M0 MCU at 48 MHz	Yes	Yes	Yes	Yes
Memory (flash, SRAM)	32 KB, 4 KB	32 KB, 4 KB	128 KB, 8 KB	128 KB, 8 KB
Integrated Type-C transceiver (number)	Yes (1)	Yes (1)	Yes (1)	Yes (2)
Integrated Type-C resistors	No	Yes (Ra, Rp, Rd)	Yes (Ra, Rp, Rd)	Yes (Rp, Rd)
Number of GPIOs	Up to 30	Up to 14	Up to 20	Up to 30
Number of serial communication blocks (I^2C/SPI/UART)	1	2	4	4
Number of TCPWM blocks (Each block can be configured as a timer, counter, or pulse width modulator)	2	6	4	4
Integrated USB Billboard Device Class Full-Speed USB 2.0 Device	No	No	Yes	No
Hardware Authentication Block (Crypto)	No	No	Yes	No
Integrated VCONN FETs	No	No	Yes (1 pair)	Yes (2 pairs)
Integrated VBUS discharge FETs	No	No	Yes	No
Integrated 20-V VBUS NFET/PFET gate drivers	No	No	Yes (2 pairs)	No
Integrated SBU/AUX analog switch	No	No	Yes	No
Supply voltage	1.8 V – 5.5 V	2.7 V – 5.5 V	2.7 V – 21.5 V	2.7 V – 5.5 V
VBUS overvoltage protection (OVP), undervoltage protection (UVP) and overcurrent protection (OCP)	Yes (Using external hardware circuitry)	Yes (Using external hardware circuitry)	Yes (Integrated)	Yes (Using external hardware circuitry)
Integrated ADCs for OVP, UVP, OCP detection, and other voltage or current measurements	1 channel (12-bit SAR)	1 channel (8-bit SAR)	2 channels (8-bit SAR)	4 channels (8-bit SAR)
USB Battery Charger (BC) Revision 1.2 and Legacy Apple Charger Detection and Emulation	No	No	Yes	No

EZ-PD CCGx Portfolio

Features	CCG1	CCG2	CCG3	CCG4
ESD protection	Yes (Up to 2.2 kV)	Yes (Up to ± 8-kV contact discharge and up-to ±15-kV air discharge)	Yes (Up to ± 8-kV contact discharge and ±15-kV air discharge)	Yes (Up to ± 8-kV contact discharge and ±15-kV air discharge)
Packages	40-QFN, 16-SOIC, 35-CSP	24-QFN, 14-DFN, 20-CSP	40-QFN, 42-CSP, 16-SOIC	40-QFN

4 CCG4 Hardware

This section introduces the CY4541 CCG4 Evaluation Kit (EVK), which enables designers to evaluate the functionality of CCG4.

4.1 CY4541 CCG4 EVK

The CY4541 EZ-PD CCG4 EVK consists of two CCG base boards and one CCG4 daughter card. The CCG4 daughter card is connected to the two CCG base boards to evaluate the CCG4's dual Type-C port capability, as shown in **Figure 2**.
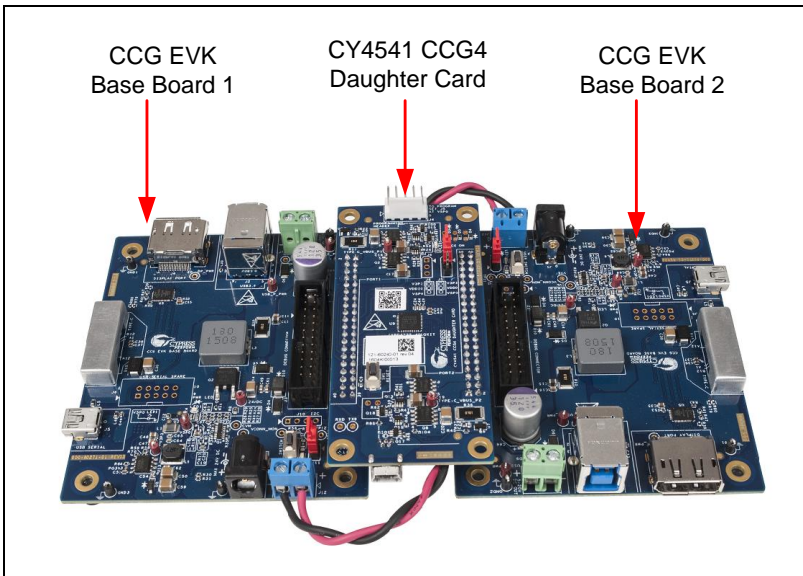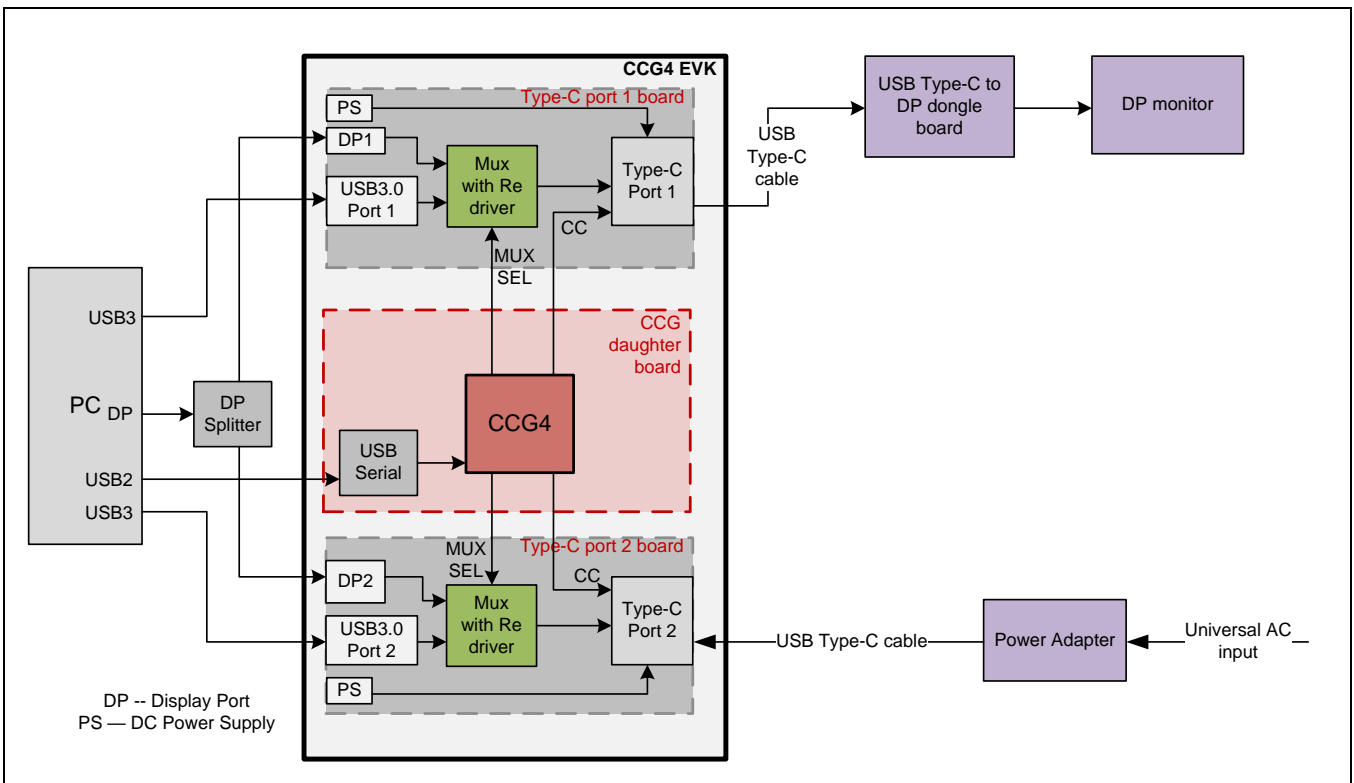

Figure 2   CY4541 CCG4 EVK Setup

Figure 3   CY4541 CCG4 EVK Architecture with an Example Use Case

Figure 3 shows the architecture of the CY4541 CCG4 EVK and an example use case. See the **CY4541 EZ-PD CCG4 Evaluation Kit Guide** for additional use cases. The CCG base board consists of a mux with re-driver, a DisplayPort, a SuperSpeed Type-B port, and a Type-C port. The CCG4 daughter card consists of the CCG4 device and a USB-Serial Bridge Controller device to provide a USB interface for downloading the firmware into the CCG4 device. The CCG4 daughter card is capable of interfacing to two CCG base boards to support dual Type-C ports.

When the CCG4 daughter card is plugged onto the two CCG base boards, the CC lines of the CCG4 are connected to the Type-C port on the respective CCG base board. The CCG4 controls the mux with re-driver, which transfers USB SuperSpeed or DisplayPort signals to the Type-C port. The EVK has power provider and consumer path control circuitry, which is controlled by the CCG4 to showcase its ability to switch the power role from a provider to a consumer and vice-versa. This EVK has overvoltage and overcurrent protection circuitry for VBUS and VCONN. It also supports CCG4 programming over SWD and I^{2}C interfaces.

A notebook or PC with two USB 3.0 ports and a DisplayPort along with the CY4541 CCG4 EVK is equivalent to a PD-enabled dual Type-C port notebook. Refer to section 5.1 ("Power Supply Connections") of the **CY4541 EZ-PD CCG4 Evaluation Kit Guide** to learn more about the Type-C notebook application using the EVK.

4.2   Powering CY4541 EZ-PD CCG4 EVK

Each CCG base board consists of a power output and power input header, which need to be connected by wires for powering the EVK. Refer to section 5.1 ("Power Supply Connections") of the **CY4541 EZ-PD CCG4 EVK Guide** for in-depth information about power supply connections between the power provider and power consumer headers.

5 CCG4 Software and Tools

Infineon provides a complete DRP application firmware and software solution for the CCG4 device to integrate single or dual Type-C interfaces in any application. The **SDK** comes with a DRP bootloader and application firmware, and documentation required to use the firmware. Refer to section 1.3 ("CCGx SDK") and chapter 2 ("SDK Installation") of the *CCGx_FW_UserGuide.pdf* for information on SDK structure, installation, and tool dependencies. You can find the *CCGx_FW_UserGuide.pdf* at the location in the SDK:
<Install_Directory>\Cypress\EZ-PD CCGx SDK\CCG3-CCG4\Documentation

5.1 CCG4 Firmware Architecture Overview

The SDK provides a USB Type-C and USB-PD specification-compliant firmware stack along with the application firmware required to implement specific USB Type-C applications using the CCG4 controller as shown in **Figure 4**.
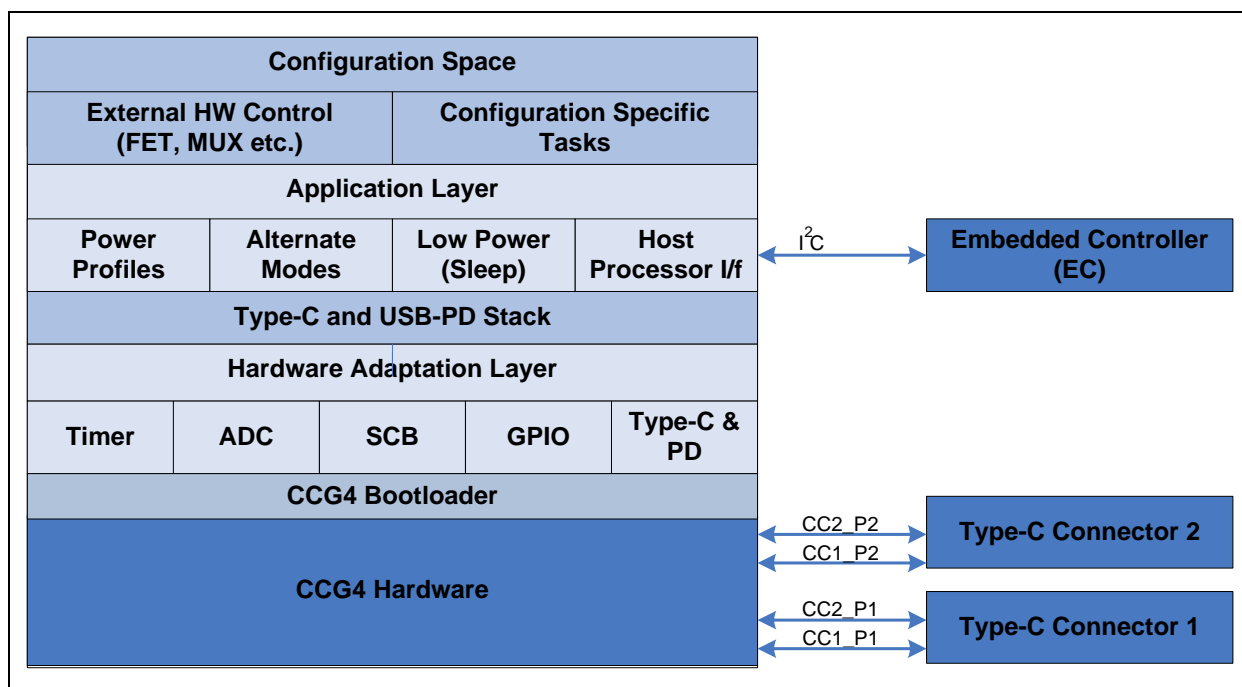

Figure 4 CCG4 Device's Firmware Modules

The CCG4 firmware solution consists of the following modules:

- Boot-loader segment, which is the starting point for firmware execution.
- Hardware adaptation layer to control the in-built ADC, timer, SCBs (I^2C/UART/SPI), GPIOs, and Type-C and PD-compliant PHY.
- USB Type-C and USB-PD specification compliant PD stack.
- Application firmware, which controls various hardware blocks of the CCG4 and provides the required USB-PD application functionality.
- Implementation of a host processor interface (HPI) that allows an external embedded controller to monitor and control the CCG4 device operation.
- Implementation of Alternate mode (for example, DisplayPort) that allows the transfer of non-USB signals over the Type-C data lanes.
- Configuration space, which provides the USB-PD configuration information used by the application firmware.

Refer to section 5 ("Firmware Architecture") of the *CCGx_FW_UserGuide.pdf* for more details about the firmware architecture. You can find the *CCGx_FW_UserGuide.pdf* at the location in the SDK: *<Install_Directory>\Cypress\EZ-PD CCGx SDK\CCG3-CCG4\Documentation*

The CCG4 bootloader allows users to upgrade the application firmware at any time in the system design process. This is described in **section 5.1.1**. Also, in a notebook system design, the CCG4 transmits and receives PD messages from the embedded controller through an HPI event to control the system PD policy. This is described in **section 5.1.2**.

5.1.1 CCG4 Bootloader

The CCG4 device is pre-programmed with a bootloader that allows firmware programming onto the device flash through an I^{2}C interface.

Because CCG4 device has 128 KB of flash, the firmware architecture allows maintaining two copies of the complete DRP firmware application in the flash memory. The redundant firmware storage allows users to perform a fail-safe firmware upgrade without the risk of device failure in case of a flashing error while programming.

The CCG4 firmware is designed to allow firmware and configuration updates through I^{2}C while the USB-PD ports are in operation. This allows users to perform a firmware update without affecting the device operation.

5.1.2 Host Processor Interface

The HPI is provided such that a local embedded controller (EC) or host processor can optionally monitor and control the runtime operation of the CCG4 device. CCG4 implements the HPI using I^{2}C interface, and the HPI interrupt is implemented using a GPIO. In a CCG4 notebook application, the EC may communicate with the CCG4 to negotiate the current with the connected Type-C device based on the charge level of the internal battery. CCG4 provides this functionality using commands, responses, events, and asynchronous messages modeled as registers as shown in **Figure 5**.

CCG4 supports two USB-PD ports, which can be independently configured and used. A two-byte addressed register space is used for these devices, so that the implementation allows for the addition of new registers and provides independent sets of registers to manage each USB-PD port.
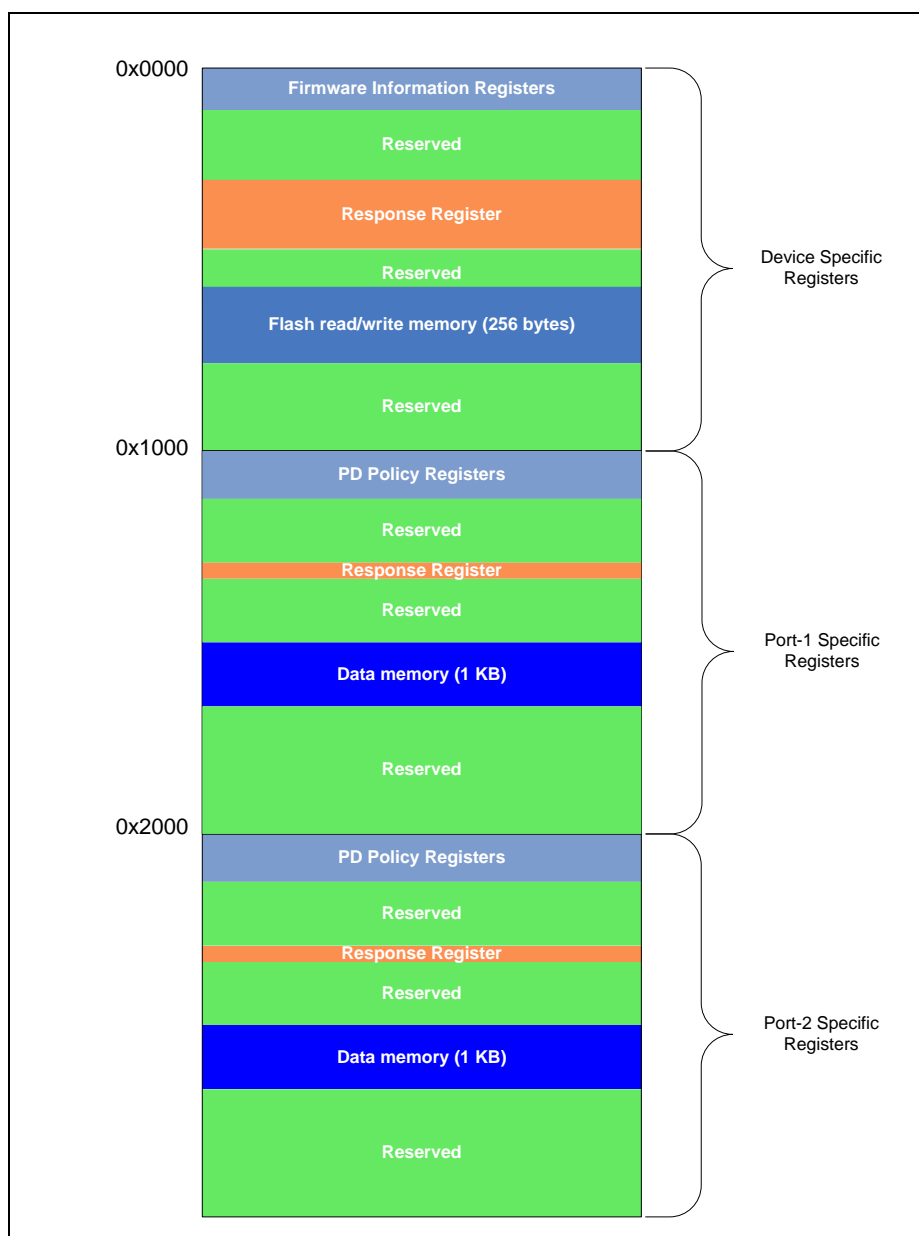
CCG4 Software and Tools

Figure 5   HPI Overview

The HPI register space is divided into three sections as follows:

- Device-Specific Registers (0x0000-0x0FFF): This space contains all command registers and status registers common to the CCG4 device (and unrelated to specific USB-PD ports). It also contains a Response register, Data memory, and Flash Read/Write memory.
  - Status registers: These registers provide information about the operating mode (such as firmware and bootloader) of the CCG4 device.
  - Command registers: These registers are used to send device level commands such as flash read/write to the CCG4 firmware.
  - Response register: This space is used to respond to device-specific commands and notify device-specific events to the EC.
  - Flash Read/Write memory: This space is used as a buffer to read or write a complete flash row.

- Port Specific Registers for PORT_1 (0x1000-0x1FFF): This space contains all command registers, status registers, and PD policy registers related to Type C port 1. It also contains a Response register similar to the device-specific registers and the Data memory.
- Port Specific Registers for PORT_2 (0x2000-0x2FFF): These registers are related to Type C port 2 on the CCG4 device, and have the same structure as the PORT_0 registers.

Contact **the technical support team** for detailed information on Host Processor Interface.

5.2 Firmware Development and Debugging Tools

Following is the list of CCG4 firmware development and debugging tools:

- Firmware Development and Programming Tools:
  - **EZ-PD Configuration Utility** – EZ-PD Configuration Utility is used to read, modify, and update the configuration parameters of the CCG4 device using the I^{2}C interface. It is also used to update the application firmware of the CCG4 device.
  - **PSoC Creator** – PSoC Creator is used to modify, debug, and program the firmware into the CCG4 device. This option is only required if base functionality of the CCG4 firmware is being modified from the standard firmware.
- Debugging Tool:
  - **EZ-PD Analyzer Utility** – The EZ-PD Analyzer utility is used along with a CY4500 EZ-PD Analyzer kit (as described in **section 5.3**) to log PD messages between CCG4 and an attached Type-C device while debugging the firmware.

5.2.1 EZ-PD Configuration Utility

The EZ-PD Configuration Utility is a Microsoft Windows application that configures the parameters stored in the internal flash memory of the CCG4. These parameters can be configured based on the customer specific application or system requirements. The EZ-PD Configuration Utility GUI (Graphical User Interface) allows users to intuitively select and configure the parameters for their applications.

You can download and install the EZ-PD Configuration Utility from **here**. The EZ-PD Configuration Utility can be executed from the following location: **Windows** > **Start** > **All Programs** > **Cypress** > **EZ-PD Configuration Utility** > **EZ-PD Configuration Utility**.

Figure 6 shows the EZ-PD Configuration Utility for the CCG4 device.
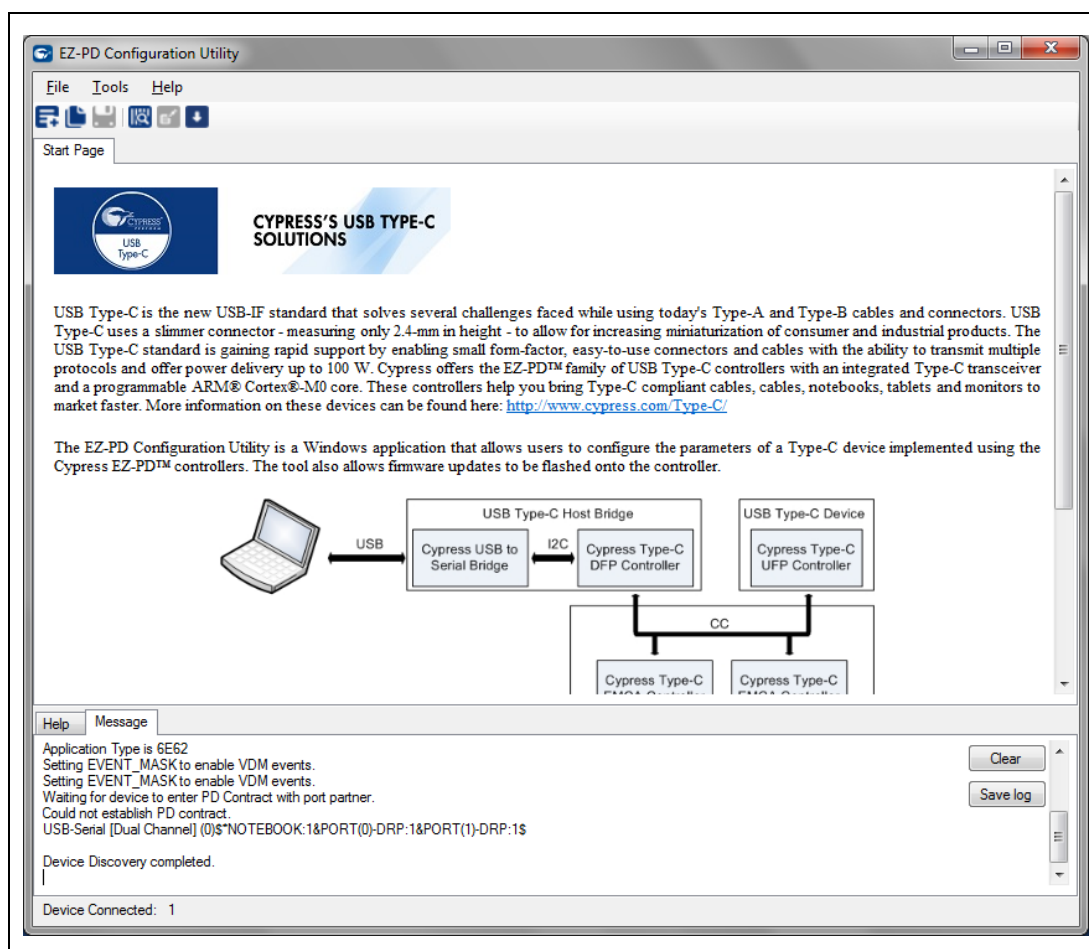
CCG4 Software and Tools

Figure 6   EZ-PD Configuration Utility Start Page

The EZ-PD Configuration Utility can be used to update application firmware and configure the CCG4 device. The utility shows the target application for the CCG4 device as a DRP notebook. The utility usage flow for configuration of the device is completed in three stages:

- Select Parameters – Select parameters available for the target application such as a **DRP notebook.**
- Create Configuration – Create a configuration from the File menu of the utility.
- Device Configuration – Program the device flash using the **Configure Device** option.

Refer to the EZ-PD Configuration Utility User Manual for more details on firmware update and configuration of the device. The user manual can be opened by clicking **Help > User Manual** in the EZ-PD Configuration Utility as shown in **Figure 7**.
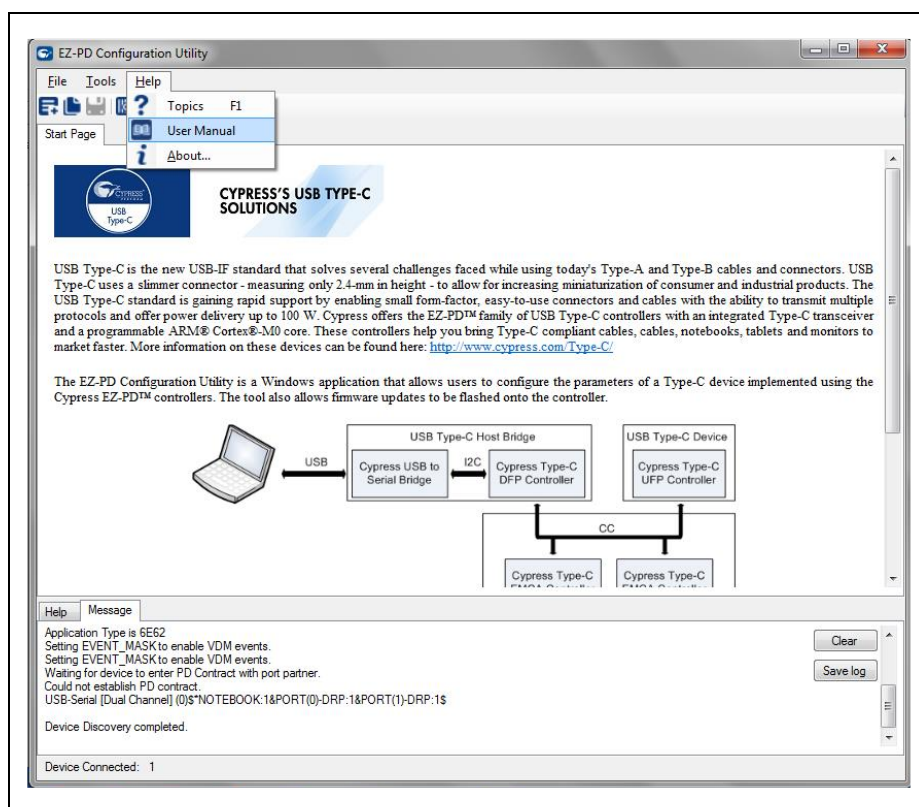
Figure 7   User Manual for EZ-PD Configuration Utility

5.2.2   CCG4 Firmware Build Environment (PSoC Creator)

PSoC Creator is a free Windows-based integrated design environment (IDE) as shown in **Figure 8**. It enables you to develop, modify, and debug the CCG4 firmware using the **Build** and **Debug** options as shown in **Figure 8**. Refer to the "Using the Reference Projects" section of the *CCGx_FW_UserGuide.pdf* for the procedure to open the reference projects. You can find the *CCGx_FW_UserGuide.pdf* at the location in the SDK: *<Install_Directory>\Cypress\EZ-PD CCGx SDK\CCG3-CCG4\Documentation.*
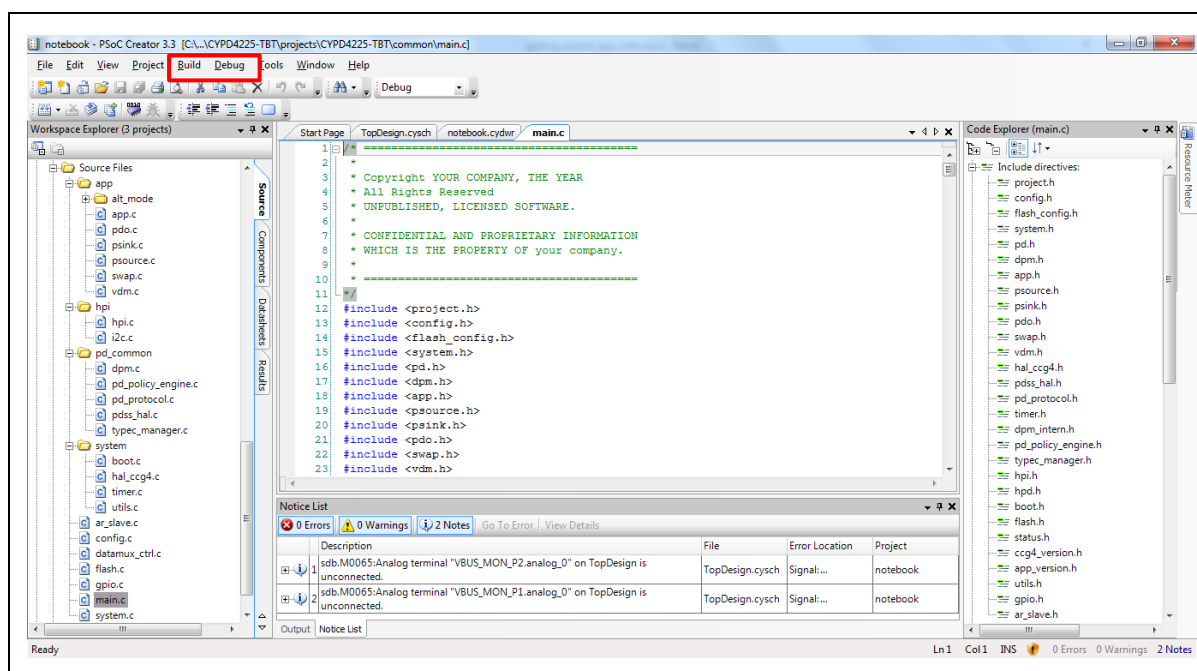

Figure 8   PSoC Creator IDE

5.2.3 PSoC Creator Help

Visit the **PSoC Creator webpage** to download and install the latest version of PSoC Creator (3.3 SP1 or later). The webpage also contains links to video training as well as other documentation. In PSoC Creator, help is available from the following items:

- **Quick Start Guide:** Choose **Help > Documentation > Quick Start Guide**. This guide gives you the basics for developing PSoC Creator projects.
- **System Reference Guide:** Choose **Help > System Reference Guide**. This guide lists and describes the system functions provided by PSoC Creator.
- **Document Manager:** PSoC Creator provides a document manager to help you to easily find and review document resources. To open the document manager, choose **Help > Document Manager**.

5.3 Introduction to Debugging Tools

This section describes the CY4500 EZ-PD Analyzer Kit and EZ-PD Analyzer Utility.

The CY4500 EZ-PD Analyzer kit supports protocol analysis of the **USB-PD** and **USB Type-C** specifications. It performs non-intrusive probing and captures accurate protocol messages on the CC line. This analyzer consists of a programmable MCU (PSoC 5LP), which monitors data on the CC line and sends this data to the host application over the USB interface. The Type-C plug and Type-C receptacle on this analyzer provide a pass-through for CC messages transmitted between the Type-C PD connections. The processor MCU (PSoC 5LP) taps these CC messages using an I^{2}C interface and transfers them over the USB interface to a PC running the host application. **Figure 9** shows the connections between the CY4500 EZ-PD Analyzer kit and the CY4541 CCG4 EVK.
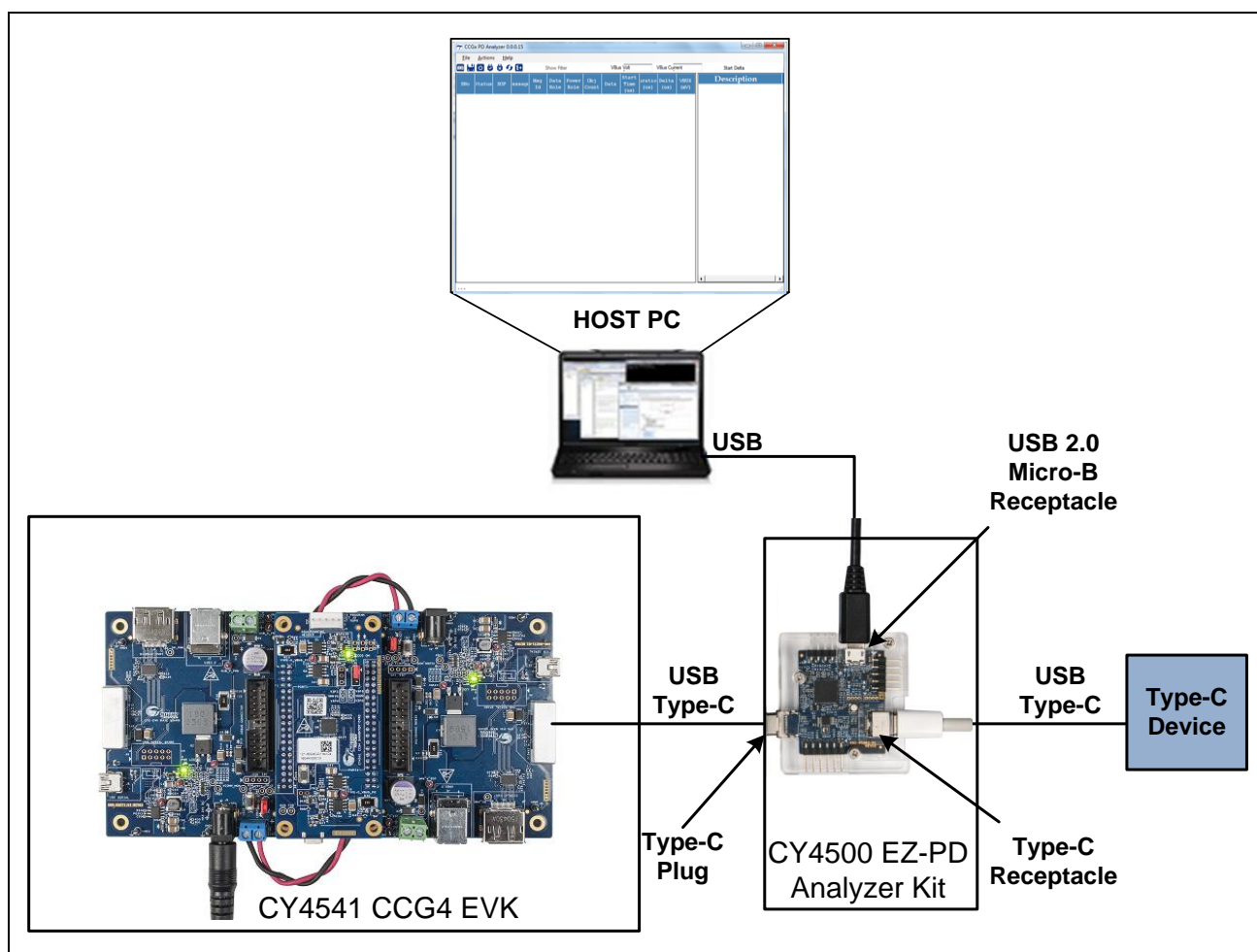

Figure 9   Connections between CY4500 EZ-PD Analyzer Kit and CY4541 EZ-PD CCG4 EVK

CCG4 Software and Tools

There is a Windows utility **EZ-PD Analyzer Utility.exe** to log messages as shown in **Figure 10**. Download and install the application and required drivers from **here**. The analyzer utility shows PD messages over CC (for example, PR_SWAP, DR_SWAP, and PDOs) while the CCG4 device is establishing a PD contract with the connected device.
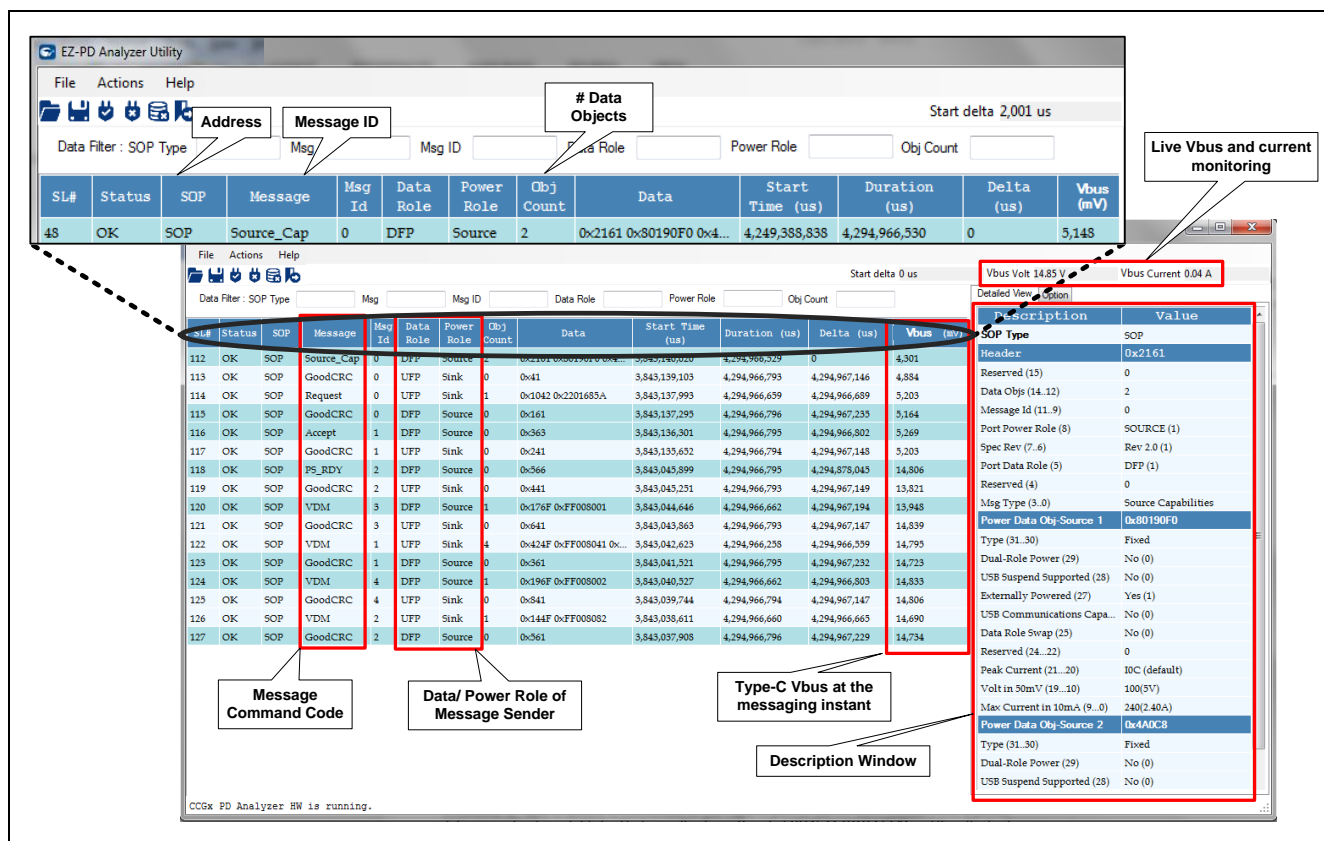

Figure 10   EZ-PD Analyzer Utility

Whenever a Type-C device is connected to the Type-C port of the CY4541 CCG4 EVK, a power delivery contract is established between the CCG4 and the attached Type-C device. The flow of CC messages as well as source and sink Power Data Objects1 (PDOs) between the DFP and UFP (or between power port partners) can be monitored on the PC by using the CY4500 EZ-PD Analyzer kit and host application. This information helps to debug the CCG4's firmware if the power contract is not correctly established between the CCG4 and the attached Type-C device.

---

1 Power Data Object is used to expose a source port's power capabilities or a sink port's power requirements as part of the Source_Capabilities or Sink_Capabilities message, respectively.

6 CCG4 DRP Demonstration and Configuration Example

This section demonstrates the dual role capability of the CCG4 for both Type-C ports using the CY4541 CCG4 EVK. It provides the procedure for modifying the configuration parameters of the CCG4 device (for example, source or sink PDOs, dead battery enable) using the EZ-PD Configuration Utility. It also gives steps to modify and debug the CCG4 DRP notebook firmware using PSoC Creator and the CY4500 EZ-PD Analyzer kit and EZ-PD Analyzer Utility.

According to the USB-PD specification, the DFP, by default, is a USB host and power source. The UFP, by default, is a USB device and power sink. PD-enabled USB products (such as a notebook with a Type-C port) can switch their power role from provider to consumer and vice-versa. The USB-PD specification refers to such USB Type-C ports as DRPs. Refer to the **USB-PD Specification** for more details.

A typical DRP can perform the roles listed in **Table 3**. DRP devices have the capability to detect the presence of the Rp and Rd resistors on the CC lines.

Table 3   DRP Device Roles

No.	Data Port Role (USB Host or Device)	Power Port Role (Power Provider or Power Consumer)
1	DFP (connect Rp and disconnect Rd)	Source (Power Provider)
2	DFP (disconnect Rp and connect Rd)	Sink (Power Consumer)
3	UFP (connect Rp and disconnect Rd)	Source (Power Provider)
4	UFP (disconnect Rp and connect Rd)	Sink (Power Consumer)

6.1 DRP Capability and Dead Battery Charging Demo

This section demonstrates the DRP capability of both Type-C ports on the CCG4 device using the CY4541 CCG4 EVK as shown in **Figure 11**. It emulates a dual Type-C port notebook design in which each Type-C port can either be a power provider or power consumer. In a CCG4-enabled notebook design, the power adapter can be connected to one Type-C port to charge the internal battery of a notebook. At the same time, a USB pen drive, display monitor, or external hard disk can be connected to the other Type-C port.

A PC with two USB 3.0 ports, a DisplayPort, and the CY4541 CCG4 EVK emulates a PD-enabled dual Type-C port PC, as shown in **Figure 11**. Refer to chapter 4 ("DRP Kit Operation") of the CY4541 CCG4 EVK Guide for details on hardware connections of this setup.

Scenario 1: Type-C Port 1 as power consumer and Type-C Port 2 as power provider

- As **Figure 11** shows, when the Type-C power adapter is connected to Type-C port 1 of the CY4541 EZ-PD CCG4 EVK, the CCG4 starts consuming power from the Type-C power adapter.
- This can be verified by measuring the voltage on the power output header (J7) of the CCG base board 1 (where the Type-C power adapter is connected) using a multimeter.
- This emulates the charging of a CCG4-enabled Type-C notebook where Type-C port 1 of the notebook consumes power from the power adapter to charge its internal battery.
- Connect a USB pen drive to Type-C port 2 of the CCG4-enabled Type-C notebook; the CCG4 device provides power to the connected USB pen drive from the Type-C power adapter.
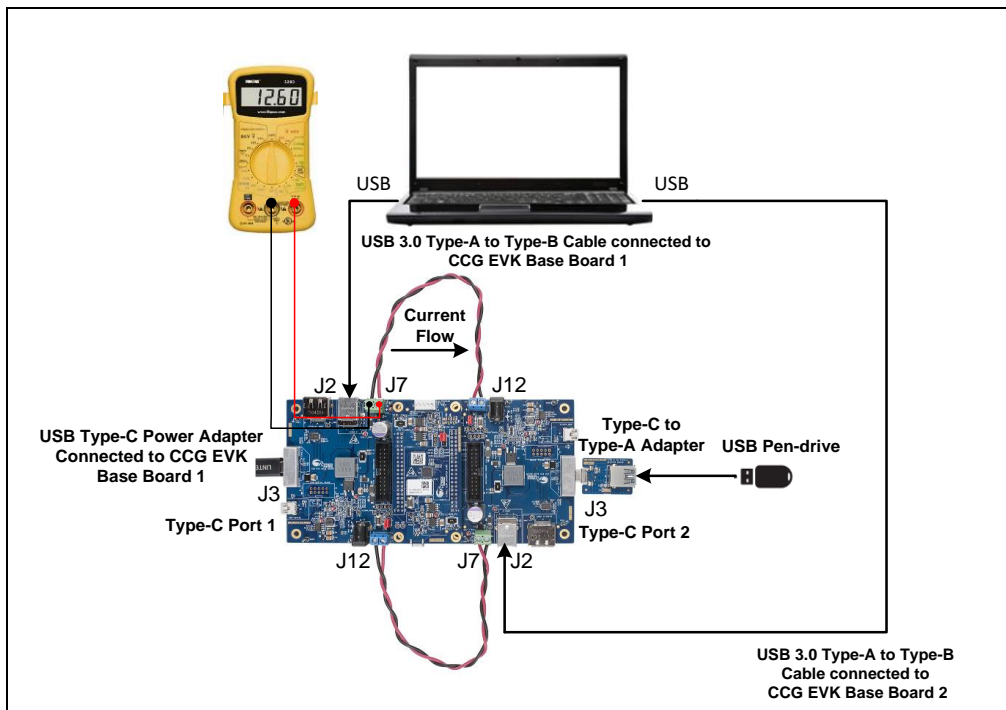
CCG4 DRP Demonstration and Configuration Example

Figure 11   DRP and Dead Battery Charging Demo Setup (Scenario 1)

Scenario 2: Type-C Port 1 as power provider and Type-C Port 2 as power consumer

- The Type-C power adapter and a USB pen drive can be interchanged between Type-C port 1 and Type-C port 2, as shown in **Figure 12**.
- In this scenario, Type-C port 1 of the CCG4-enabled notebook provides power to the USB pen drive and Type-C port 2 consumes power from the Type-C power adapter.

This demonstrates that the CCG4 can switch its power role from provider to consumer and vice-versa and it can be implemented on both the Type-C ports.
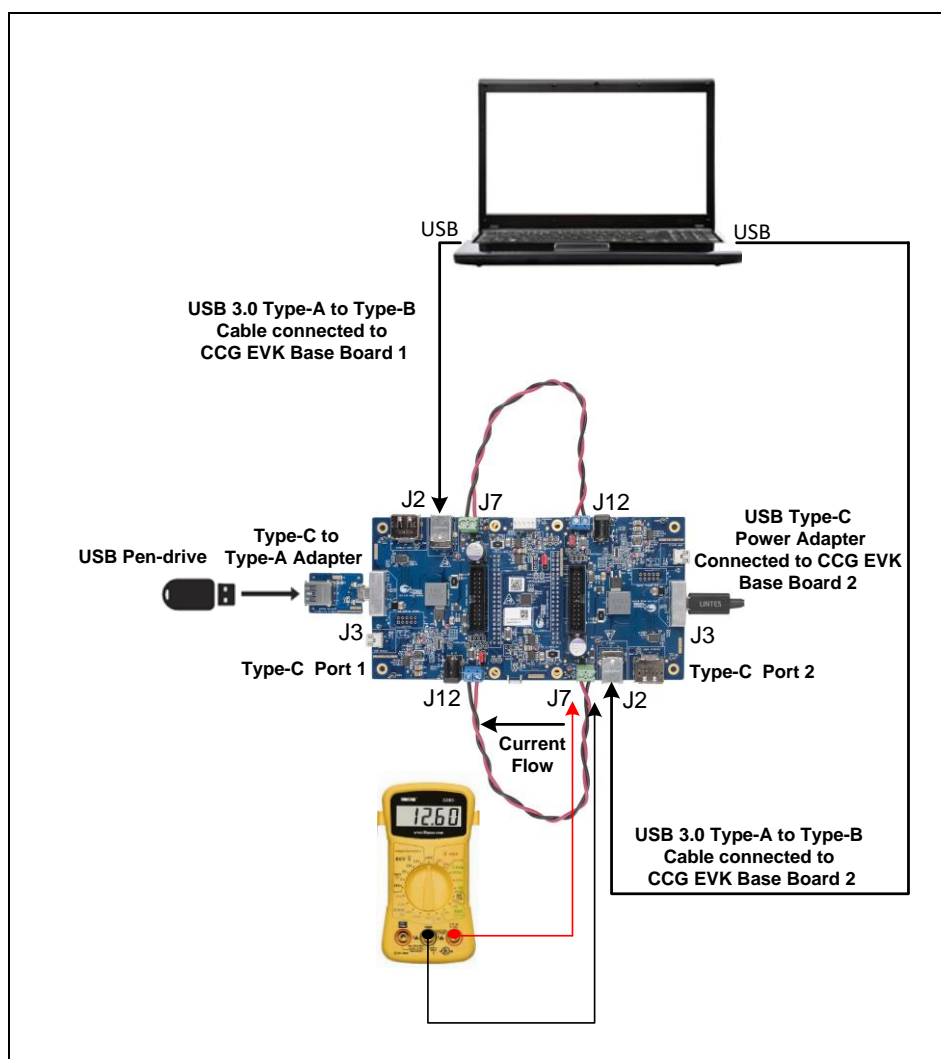
Figure 12   DRP and Dead Battery Charging Demo Setup (Scenario 2)

6.1.1   Dead Battery Charging Demo

Dead battery condition is emulated by not connecting the DC power adapter to the EVK on J12, the onboard CCG4 is not powered, which emulates a dead battery scenario. The CCG4 can be powered by connecting a Type-C power adapter to one of the EVK's Type-C ports, as shown in **Figure 11**. When the CCG4 in the EVK is powered, it establishes a power contract with the Type-C power adapter and starts consuming power. This can be verified by connecting a multimeter to the power output header (J7) of the CCG base board (where the Type-C power adapter is connected) to measure the output voltage in the dead battery charging scenario. This demonstrates that a CCG4-enabled Type-C notebook can be charged even from a dead battery condition.

6.2   Modifying CCG4 Configuration Parameters

Various configuration parameters of the CCG4 device are stored in the device's internal flash memory. These configuration parameters may need to be changed based on the system requirements. This can be demonstrated by a simple example of connecting a custom Type-C power adapter to the CY4541 CCG4 EVK. This Type-C power adapter (For example, **Apple Type-C power adapter**) supports a custom PDO. The power contract between the Type-C power adapter and the CCG4 device with the negotiated PDO will be established only if the CCG4 device supports the custom PDO. The EZ-PD Configuration Utility allows the user to modify parameters such as sink or source PDO without changing the base CCG4 firmware.

Following are the steps to modify the CCG4 configuration parameters for establishing a power contract with custom Type-C adapter:

1. Update the configuration parameters (removal of default sink PDO such as 14.8 V, 900 mA) of CCG4 device using the EZ-PD Configuration Utility

2. Test the CY4541 CCG4 EVK setup with a Type-C power adapter

3. Update the configuration parameters (sink PDOs to support custom PDO, such as 14.8 V, 2 A) in the EZ-PD Configuration Utility.

4. Re-test the CY4541 CCG4 EVK setup with a Type-C power adapter.

6.2.1 Step1: Update the configuration parameters of CCG4 device using the EZ-PD Configuration Utility

This section describes the steps to configure the CY4541 CCG4 EVK setup.

As shown in **Figure 11**, connect the wires between the power input header (J12) and power output header (J7) of the CCG base boards. Ensure that the voltage selection jumper (J5) is set to 5 V (pins 2 and 3 of jumper J5 on the CCG4 daughter card are shorted).

1. Wait for driver detection and binding for the USB-Serial controller on the CY4541 CCG4 EVK. The driver for this controller can be obtained by searching on Windows Update. Once the driver installation is successful, a "USB-Serial (Dual Channel) Vendor 1" device will be listed under Universal Serial Bus Controllers in the Device Manager window. See **Figure 13** for the expected device listing.
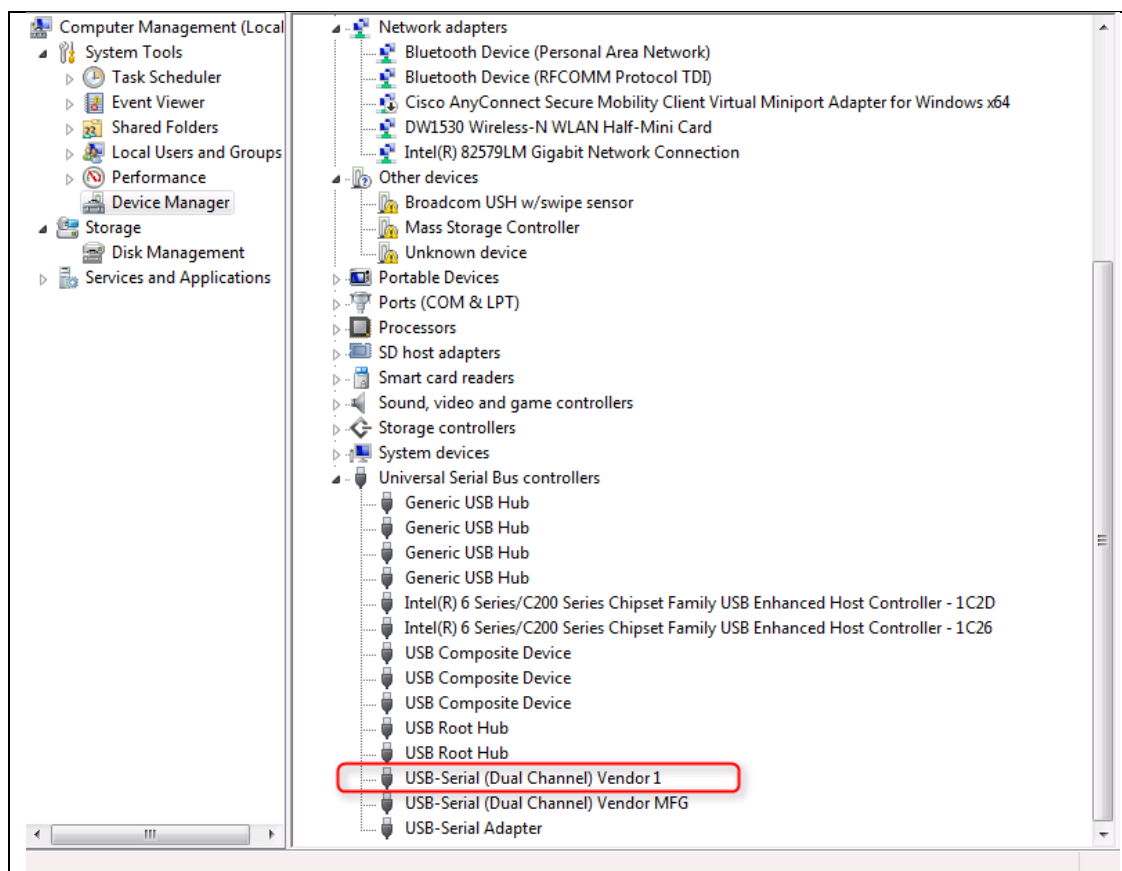

Figure 13   Device Manager View Showing USB-Serial Bridge Device

CCG4 DRP Demonstration and Configuration Example

2. Launch the EZ-PD Configuration Utility from **Windows > All Programs > Cypress > EZ-PD Configuration Utility > EZ-PD Configuration Utility**. If the device driver binding is successful, the GUI should report one device connected on the status bar of the GUI as shown in **Figure 14**.
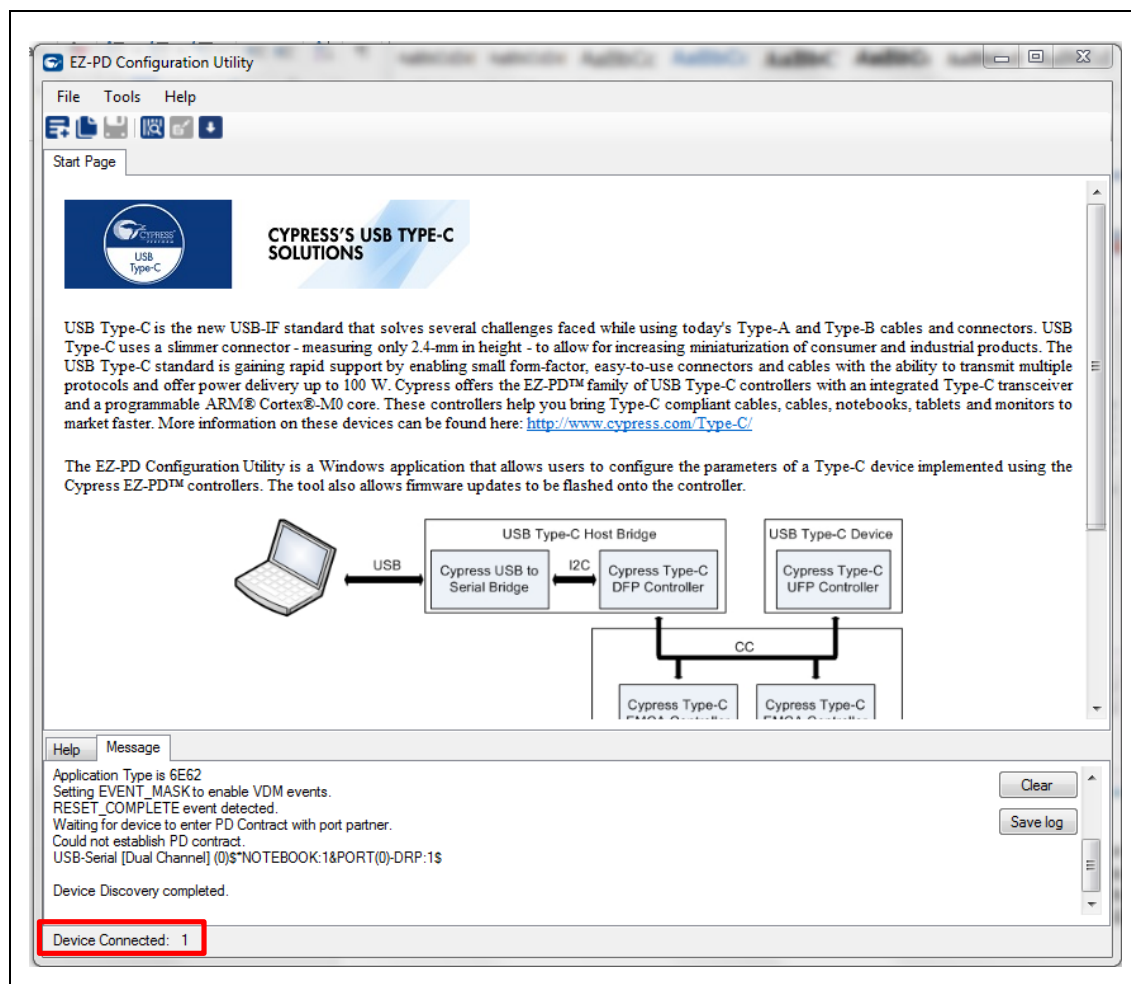

Figure 14   Configuration Utility showing valid connection to CY4541 CCG4 EVK

3. Select **File > Read From Device**, select **NOTEBOOK** in the USB-PD Device list as shown in **Figure 15**. Select the **Bootloader Read** checkbox and click on the **Read** button in the popup window to read the existing configuration as shown in **Figure 15**.

    *The existing configuration will now be read, and will be displayed in the utility in a tree under **CCGx Configuration** in the left panel as in **Figure 15**. Device configuration parameters are classified into Device Parameters, Port Parameters, and User Parameters. For multi-port devices there will be multiple copies of the port parameter entries. Port parameters are further classified and grouped into a hierarchical tree structure.*
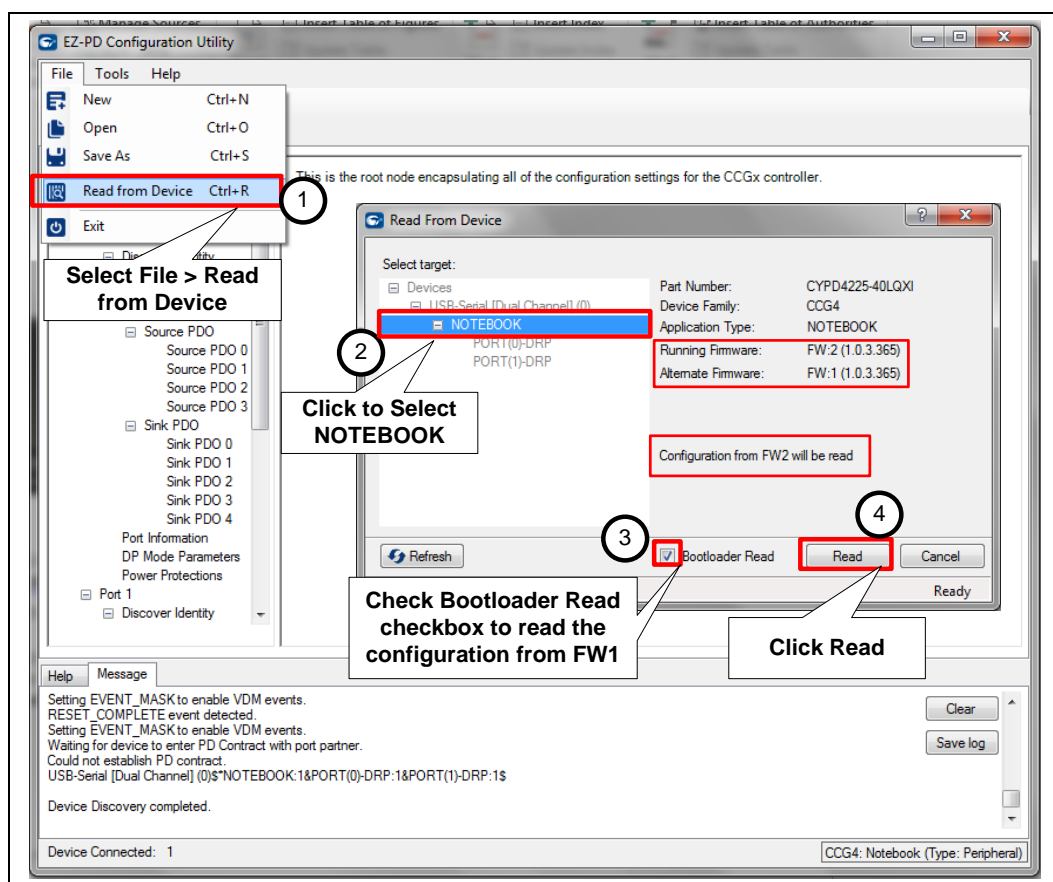
Figure 15   Reading the existing configuration table of CCG4 device

1. *The existing configuration will now be read, and will be displayed in the utility in a tree under **CCGx Configuration** in the left panel as in **Figure 16**. Device configuration parameters are classified into Device Parameters, Port Parameters, and User Parameters. For multi-port devices there will be multiple copies of the port parameter entries. Port parameters are further classified and grouped into a hierarchical tree structure.*
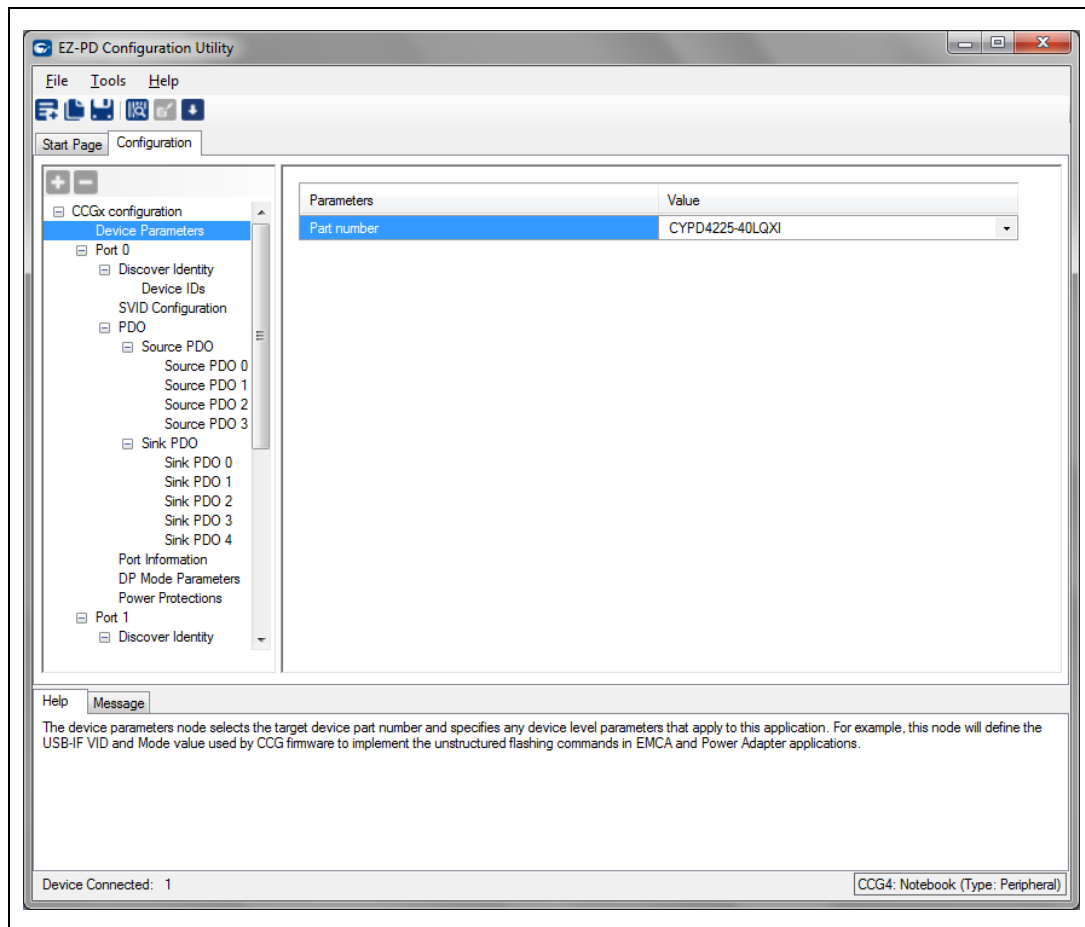
Figure 16   Existing Configuration Table of CCG4 device in CY4541 CCG4 EVK

2. Select the **Sink PDO** option in the CCGx configuration list for Port 0 (corresponds to the Type-C port of CY4541 CCG4 EVK). It can be seen that the CCG4 supports five Sink PDOs listed as Sink PDO 0 – 4 as shown in **Figure 17** and the window on the right will show the values of voltage and current corresponding to the selected PDO. For example, clicking on Sink PDO 2 will show the corresponding voltage of 14.8V, 900mA (expressed in units of 50mV, 10mA) in the main window. That is, 296 * 50mV = 14.8V and 90 * 10mA = 900mA.
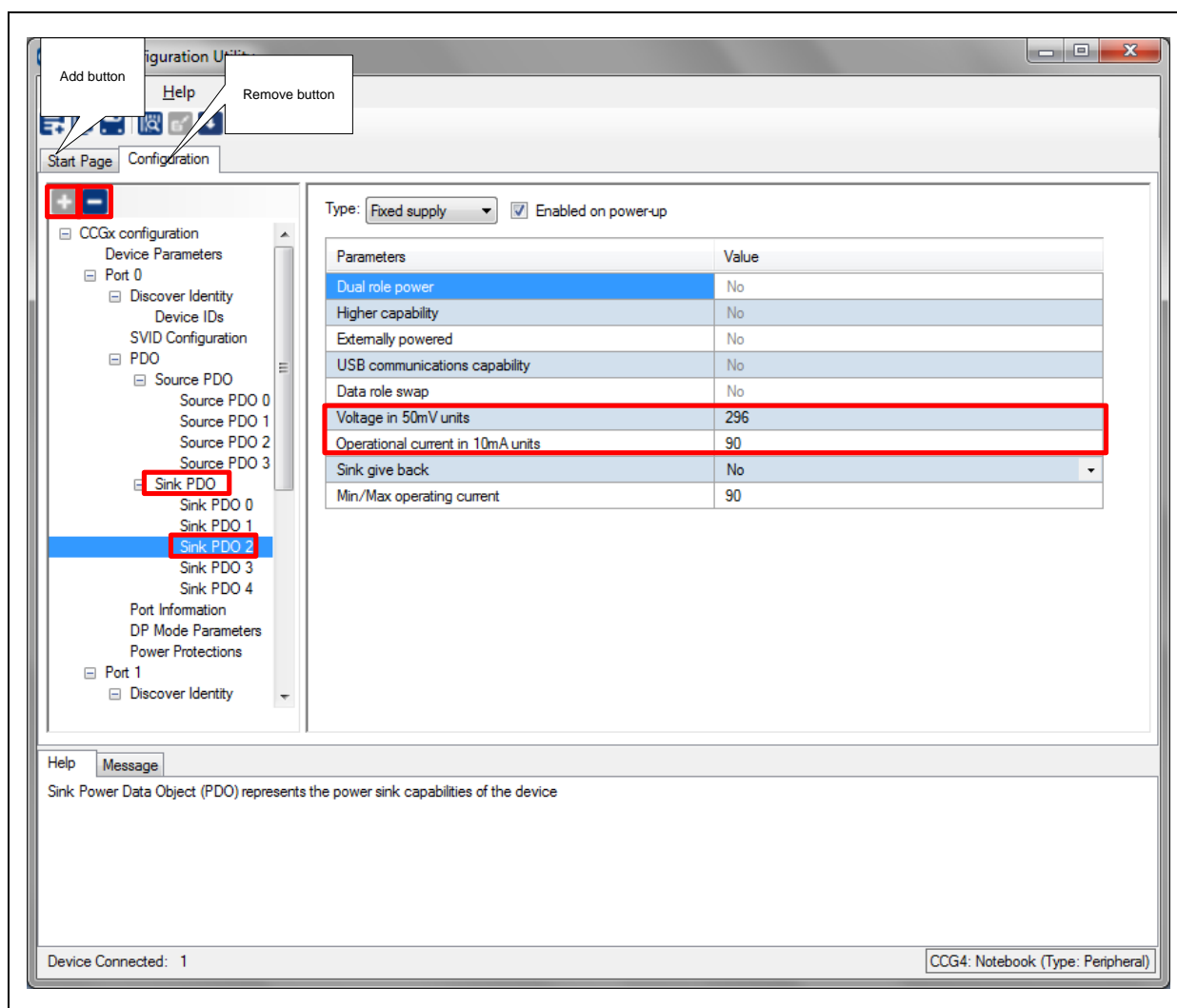
Figure 17   Sink PDOs supported by CCG4 device of CCG4 EVK

3.   Select the Sink PDO 2 and click on Remove (-) button to delete it.

> **Note**: *Add (+) button can be used to add more Sink PDOs to the list and Remove (-) button can be used to delete any selected PDO.*

Click on the **Save As** icon to save the modified configuration in any convenient location (say in Desktop) as shown in **Figure 18**. The modified configuration is saved as an XML file. A .cyacd and a .c file will be saved in the same location.
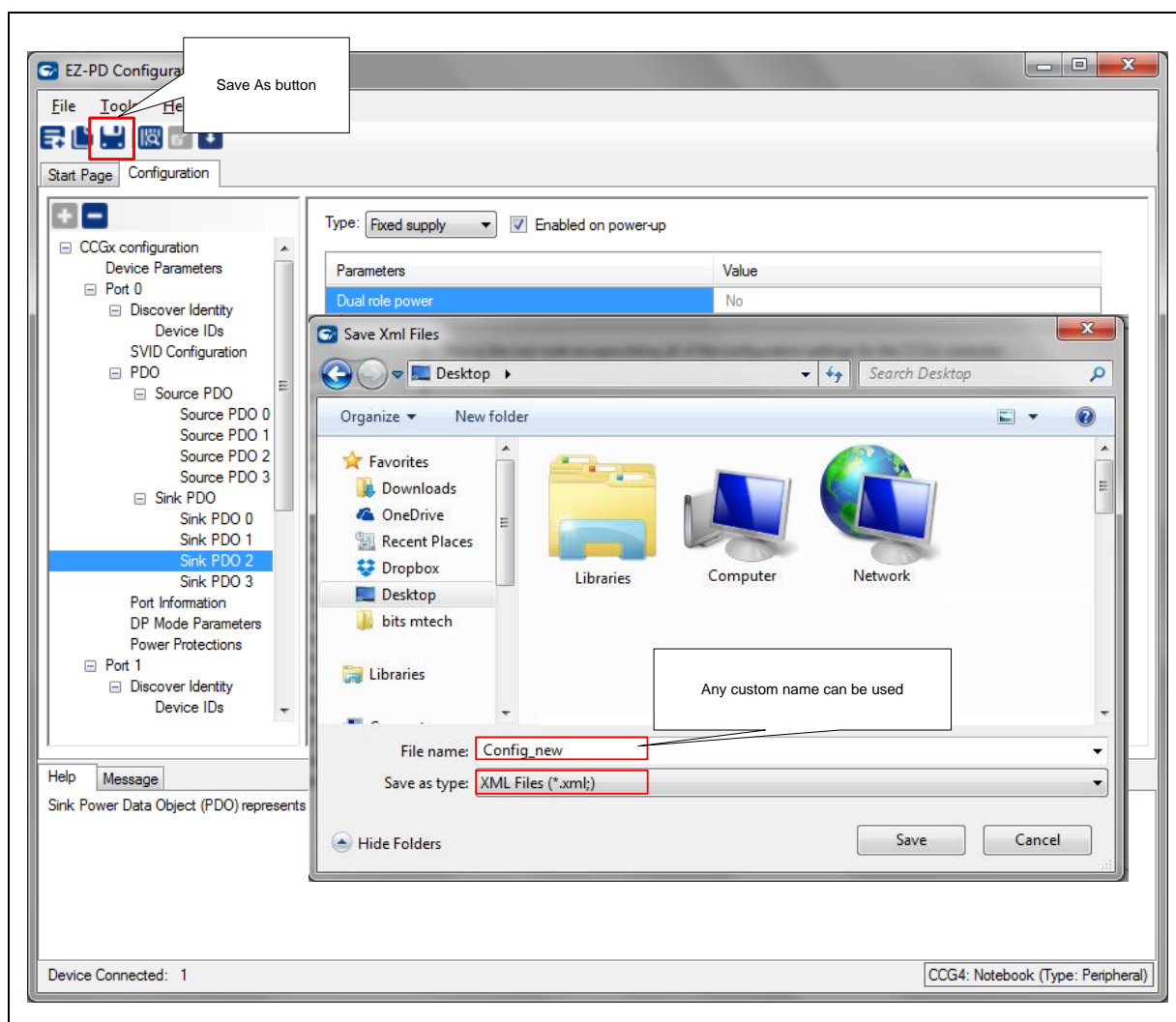
Figure 18   Save the modified Configuration

4.   Select **Tools > Configure Device** as shown in **Figure 19**. This brings up the Configure Device Dialog box where the target device to be programmed can be selected.
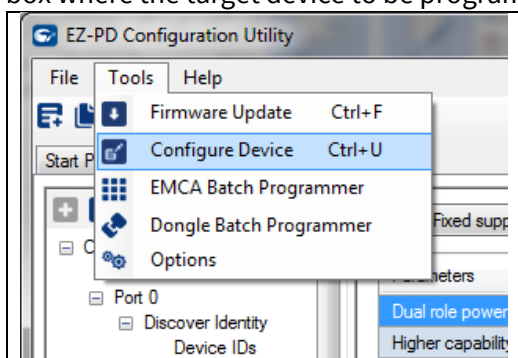
Figure 19   Select Configure Device Menu

5. Select "**Notebook**" as the target device, select the appropriate "cyacd" file to be programmed, then click on the Program button as shown in **Figure 20**.

Note: The selected configuration file updates the configuration parameters of the firmware that is not running, i.e. the alternate firmware.
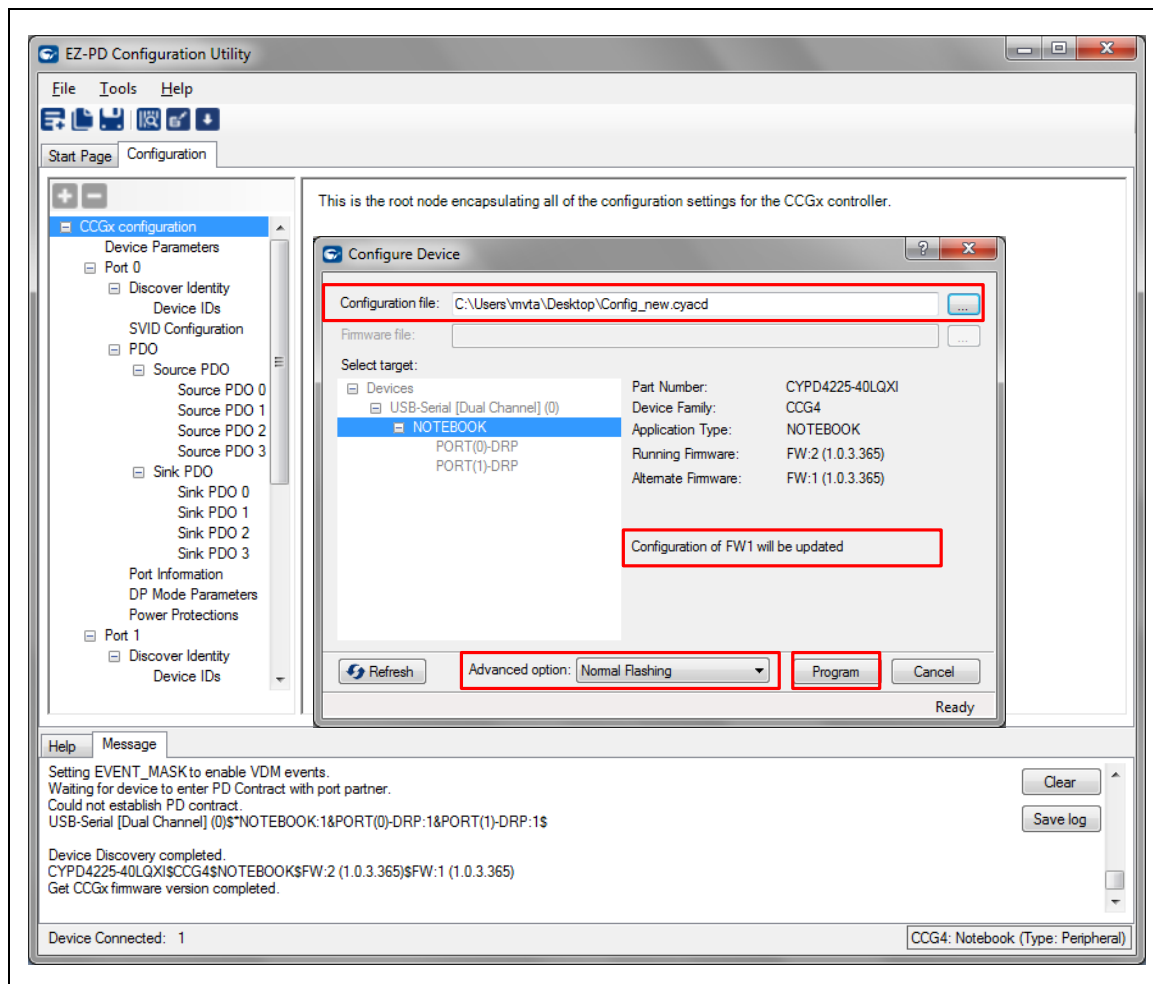
Figure 20   Updating Configuration Table using EZ-PDTM Configuration Utility

6. The successful programming of the new configuration file will be indicated by a message box with the message **Flashing Configuration Succeeded.** The **Message** window at the bottom of the utility also logs messages during each operation as shown in **Figure 21**.
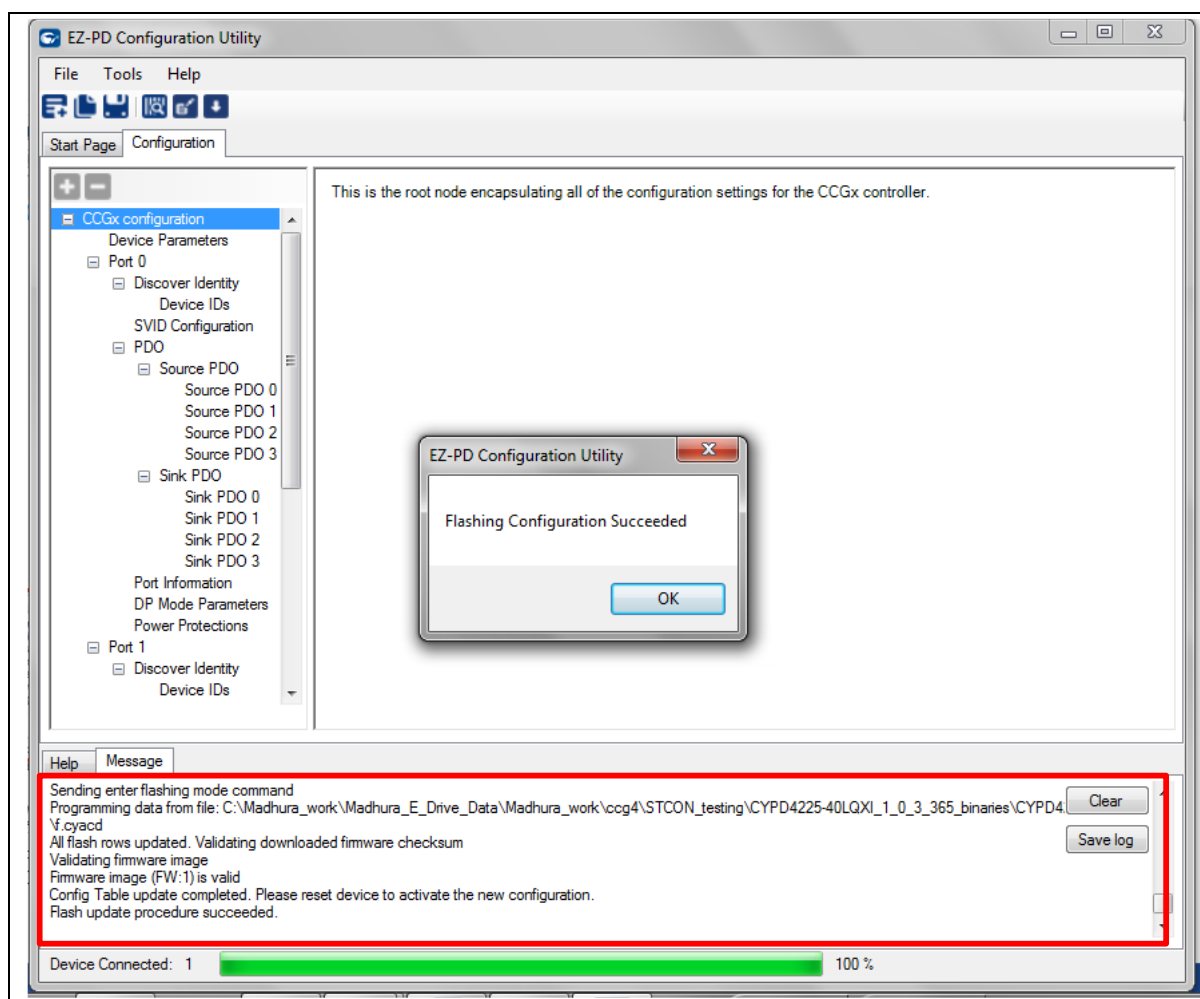
Figure 21   Message Showing Successful Update of Configuration Parameters

7. *With this, we have successfully removed one of the Sink PDOs (14.8V, 900mA) from the CCG4 device's configuration table. Power cycle the CY4541 CCG4 EVK or press the reset switch SW1 on the CCG4 Daughter card to switch to the newly updated configuration on the CCG4.*

6.2.2   Step 2: Test the CY4541 CCG4 EVK Setup with Custom Type-C Power Adapter

This section describes the steps to test the CY4541 CCG4 EVK setup.

As shown in **Figure 11**, connect the wires between the power input header (J12) and power output header (J7) of the CCG base boards. Ensure that the voltage selection jumper (J5) is set to 5 V (pins 2 and 3 of jumper J5 on the CCG4 daughter card are shorted).

When the Type-C power adapter is connected to Type-C port 1 of the CY4541 CCG4 EVK, the CCG4 will not be able to establish a power contract with the Type-C power adapter; this is because the CCG4 is not configured for the custom PDOs supported by the Type-C power adapter. This can be verified by measuring the voltage on the power output header (J7) of CCG base board 1 (where the Type-C power adapter is connected) using a multimeter. The output voltage on this header will be 5 V, which is vSafe5V (the default) instead of the negotiated voltage (for example, 14.8 V) supported by the Type-C power adapter.

CCG4 DRP Demonstration and Configuration Example

The analyzer explained in **section 5.3** can be used for debugging. Connect the analyzer to the CY4541 CCG4 EVK, as shown in **Figure 9**. Connect the Type-C port 1 or Type-C port 2 of CY4541 CCG4 EVK to the Type-C plug of CY4500 EZ-PD Analyzer kit. Connect USB 2.0 Micro-B cable between PC's USB port and USB 2.0 Micro-B receptacle on CY4500 EZ-PD Analyzer kit. Follow these steps to take the CC trace for further debugging.

- − Download and install the EZ-PD Analyzer Utility from **here**.
- − Open the EZ-PD Analyzer Utility and click the **Start Capturing** icon to take the CC trace as shown in **Figure 22**.
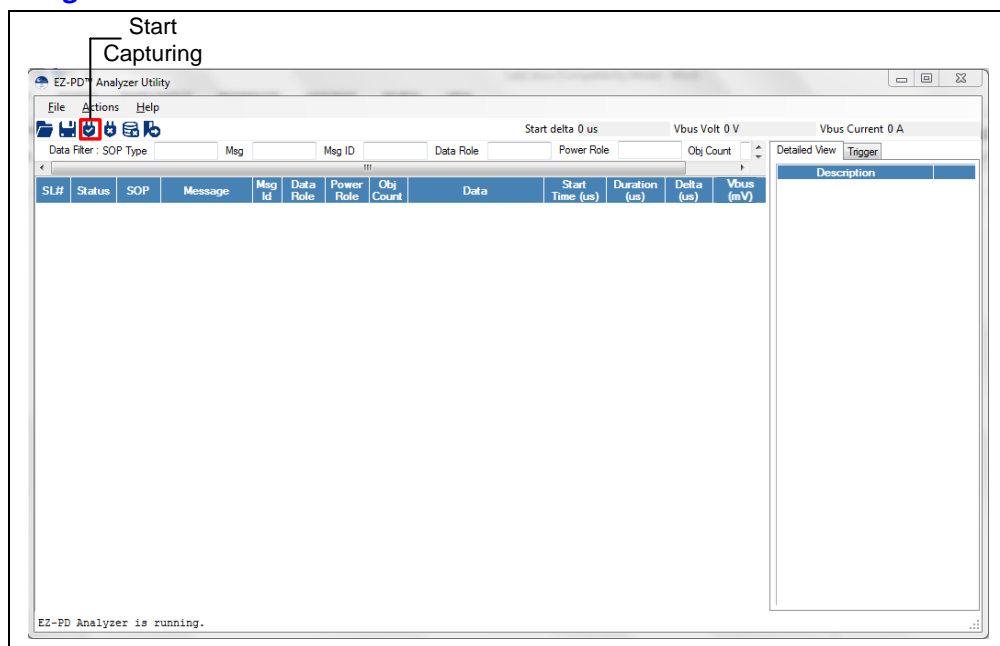
Figure 22   Connect the EZ-PD Analyzer to PC

- − Connect the Type-C power adapter to Type-C receptacle of CY4500 EZ-PD Analyzer kit.
- − The CC trace in **Figure 23** shows the PDOs supported by the Type-C power adapter (see boxed area). Based on this trace, you can identify the PDOs supported by the Type-C power adapter. This PDO can be added to the CCG4 device's application firmware to establish a power contract between the CCG4 and the Type-C power adapter. The CC trace in **Figure 24** shows that a power contract for 14.8 V is not established and the Type-C power adapter is providing 5 V to the device. During the power negotiation phase, power provider device (here, Type-C power adapter) sends the "Source Capability" message to the attached power consumer device (here, CCG4). The Source Capabilities message in **Figure 24** shows the custom PDOs (Power Data Obj Source 1 and Power Data Obj Source 2) supported by the Type-C power adapter. The "Request" message after the "Source Capabilities" message is always sent by a power consumer device to request the power. Since the CCG4 device is not able to establish a power contract for 14.8 V, it requests for Power Data Object position 1 (boxed area), which corresponds to the 5 V PDO. Refer to the **USB Power Delivery (PD) Specification Revision 2.0, Version 1.1** for more details on PD messages.
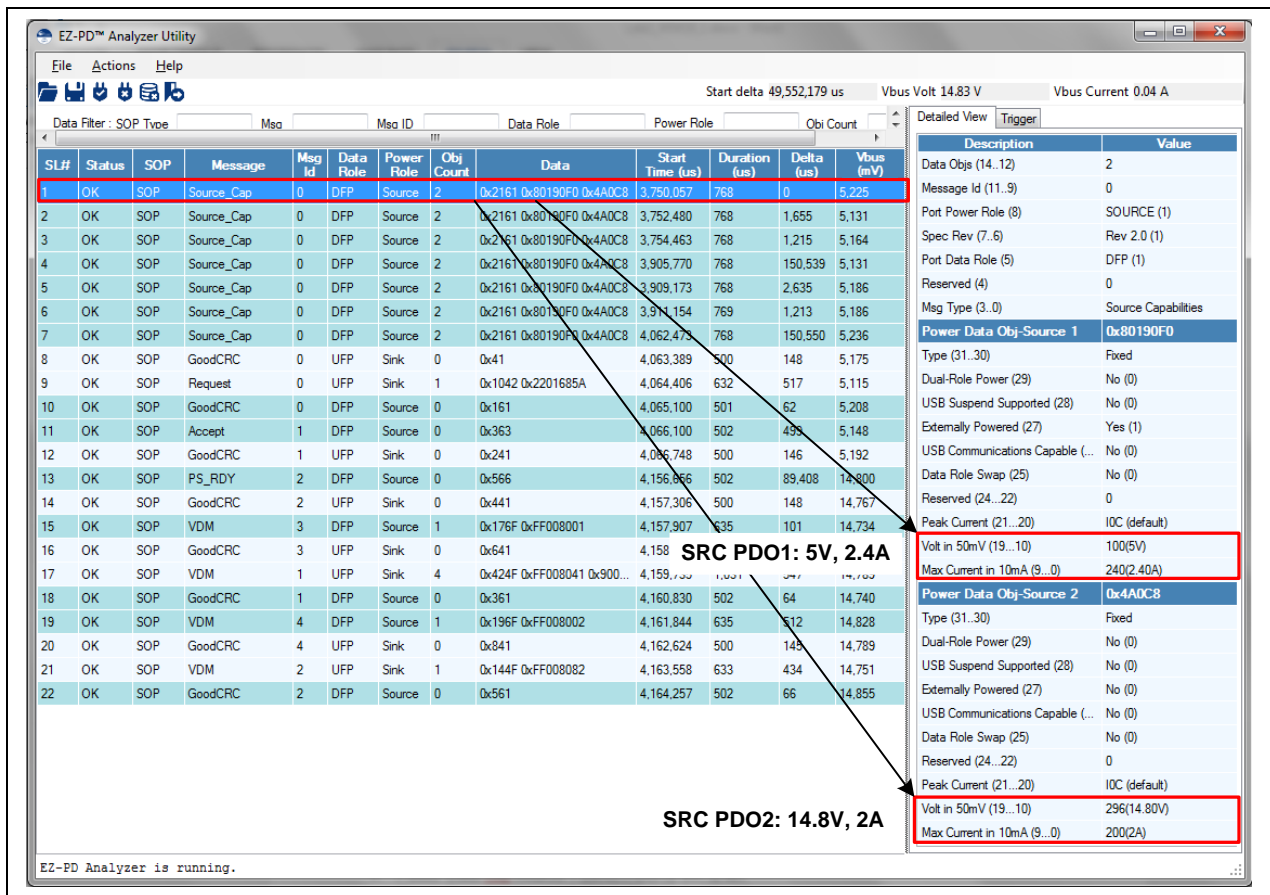
CCG4 DRP Demonstration and Configuration Example

Figure 23   Sink PDOs of Type-C Power Adapter
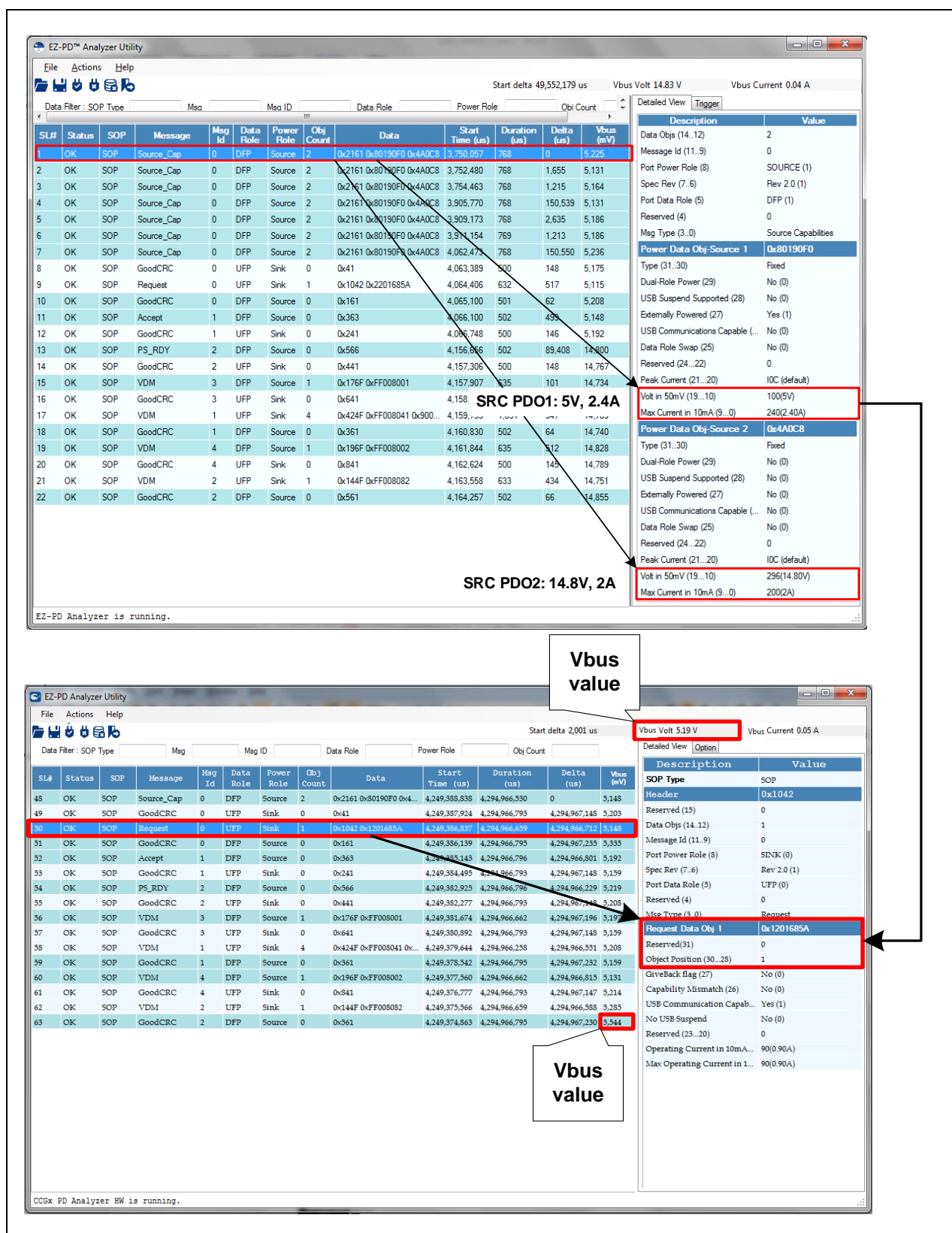
CCG4 DRP Demonstration and Configuration Example

Figure 24   Debugging Using EZ-PD Analyzer Utility

–

6.2.3 Step 3: Update Configuration Parameters of the CCG4 Device

This section gives step-by-step instructions to update the CCG4 configuration parameters using the EZ-PD Configuration Utility.
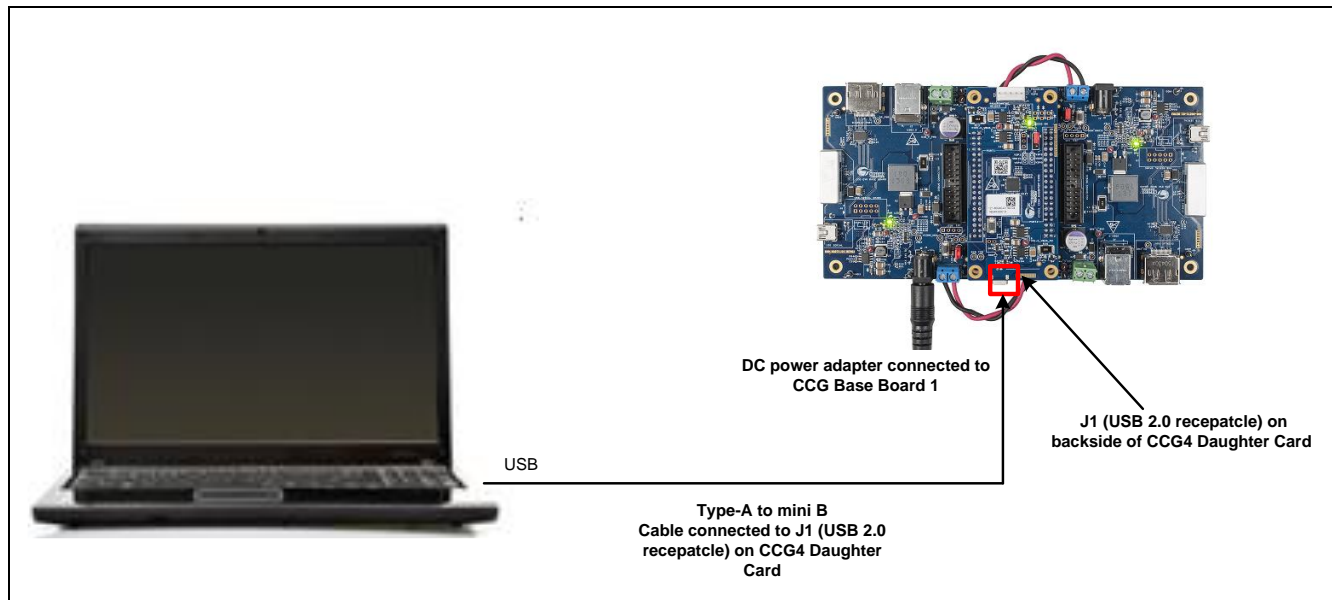

Figure 25   Updating the Configurable Table of CCG4 device

1. Download and install the EZ-PD Configuration Utility.

2. Start the EZ-PD Configuration Utility and choose File > New to create a new configuration.

3. Connect the boards and cables (USB 2.0 USB 2.0 Type-A to Mini-B cable, DC power adapter, and wires between the power input header (J12) on CCG EVK base board 1 and CCG EVK base board 2) as shown in **Figure 25.** Connect the DC power adapter to CCG EVK base board 1.

4. Select the CCG4 device in the CCG Type drop-down list. Select Notebook in the USB-PD Device list.

5. Select the Sink PDO option in the CCGx configuration list. Click the Add (+) button as shown in **Figure 26**. Sink PDOs 0 to 3 are already supported by the CCG4 device. This step will configure the CCG4 device with the new Sink PDO. EZ-PD configuration utility will display 5 Sink PDOs (from 0 to 4) after the addition of new Sink PDO.
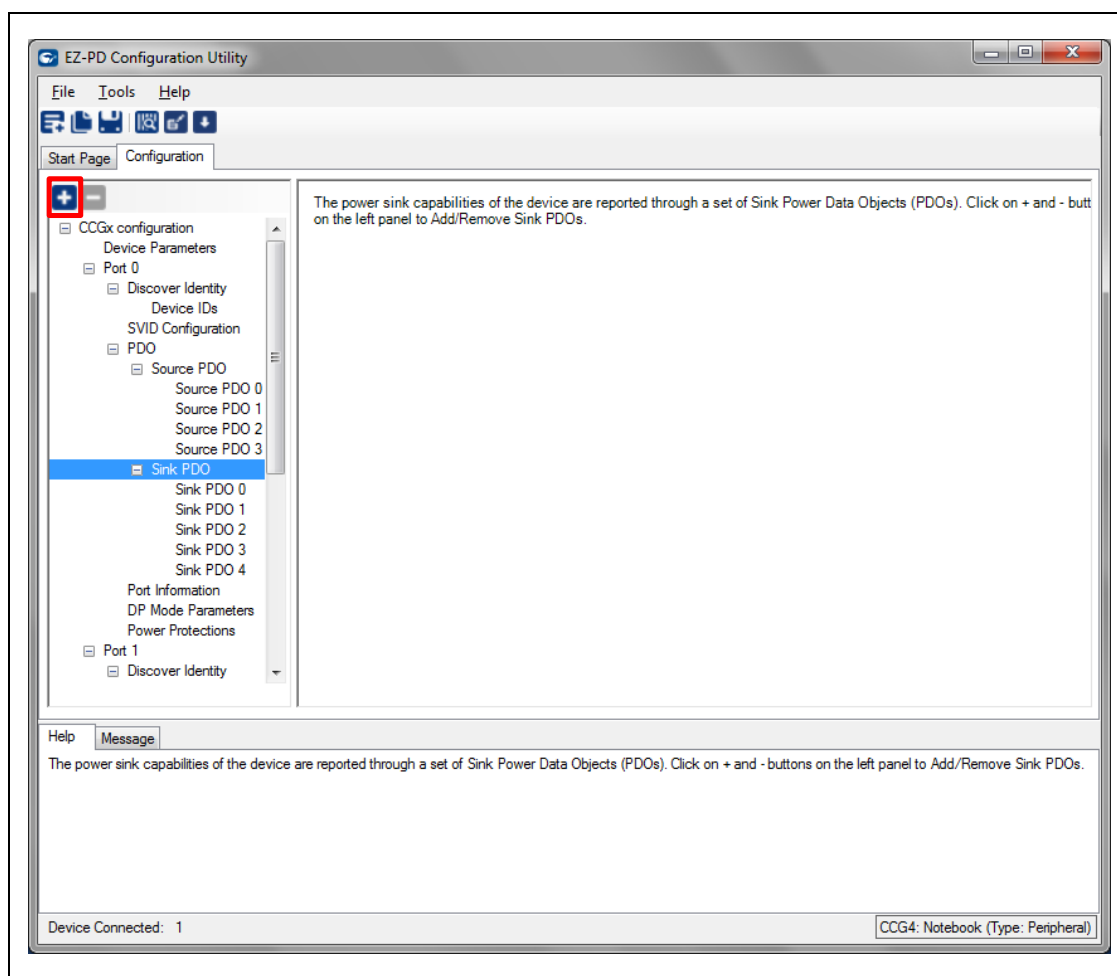
Figure 26   Adding Sink PDO in CCG4 Device Firmware

6. As the Type-C power adapter supports a custom (14.8 V, 2 A) power profile, update the Voltage in 50 mV units and Operational Current in 10 mA units widgets with 296 (as defined in **Table 4** for 14.8 V) and 200 (as defined in **Table 5** for 2 A) respectively as shown in **Figure 27**.

7. Follow steps 8 to 12 in step 1 to configure the device with this new configuration.

8. Refer to section 3.2 ("Updating CCGx Configuration") of the CCGx_FW_UserGuide.pdf for information on configuring the CCG4 device with custom sink PDOs. You can find the CCGx_FW_UserGuide.pdf at the location in the SDK: <Install_Directory>\Cypress\EZ-PD CCGx SDK\CCG3-CCG4\Documentation

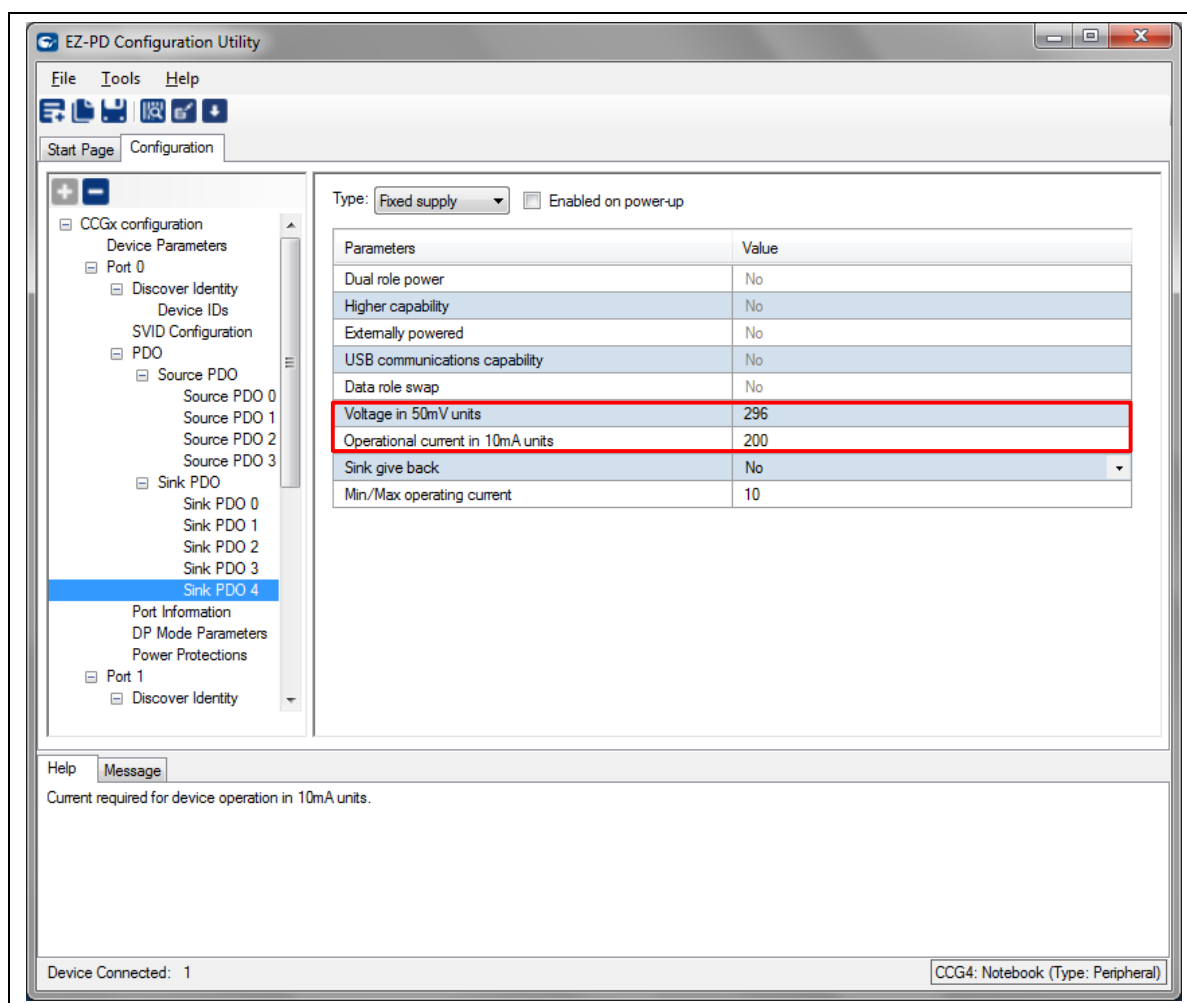**CCG4 DRP Demonstration and Configuration Example**

Figure 27   Updating Sink PDO with Required Voltage and Current

Table 4   Voltage in 50 mV for Corresponding PDO Voltage

Voltage (V)	Voltage in 50 mV units (mV)
5	100
12	240
14.8	296
20	400

Table 5   Minimum Operating Current for Corresponding PDO Current

Current (A)	Operational Current in 10 mA units (mA)
2	200
3	300
4	400
5	500

6.2.4 Step 4: Test the CY4541 CCG4 EVK Setup with the Custom Type-C Power Adapter

Make sure that wires are connected between the power input headers (J12) and the power output headers (J7) of the CCG base boards.

When the Type-C power adapter is connected to Type-C port 1 of the CY4541 CCG4 EVK, the CCG4 establishes a power contract with the Type-C power adapter. This can be verified by measuring the voltage on the power output header (J7) of CCG base board 1 (where the Type-C power adapter is connected) using a multimeter.

The output voltage on this header will be the negotiated voltage (for example, 14.8 V), which is the negotiated voltage between the CCG4 device and the Type-C power adapter. The CC trace in **Figure 28** shows that power contract for 14.8 V is established because the CCG4 device has requested the Power Data Object position 2 (boxed area), which corresponds to the 14.8 V PDO.

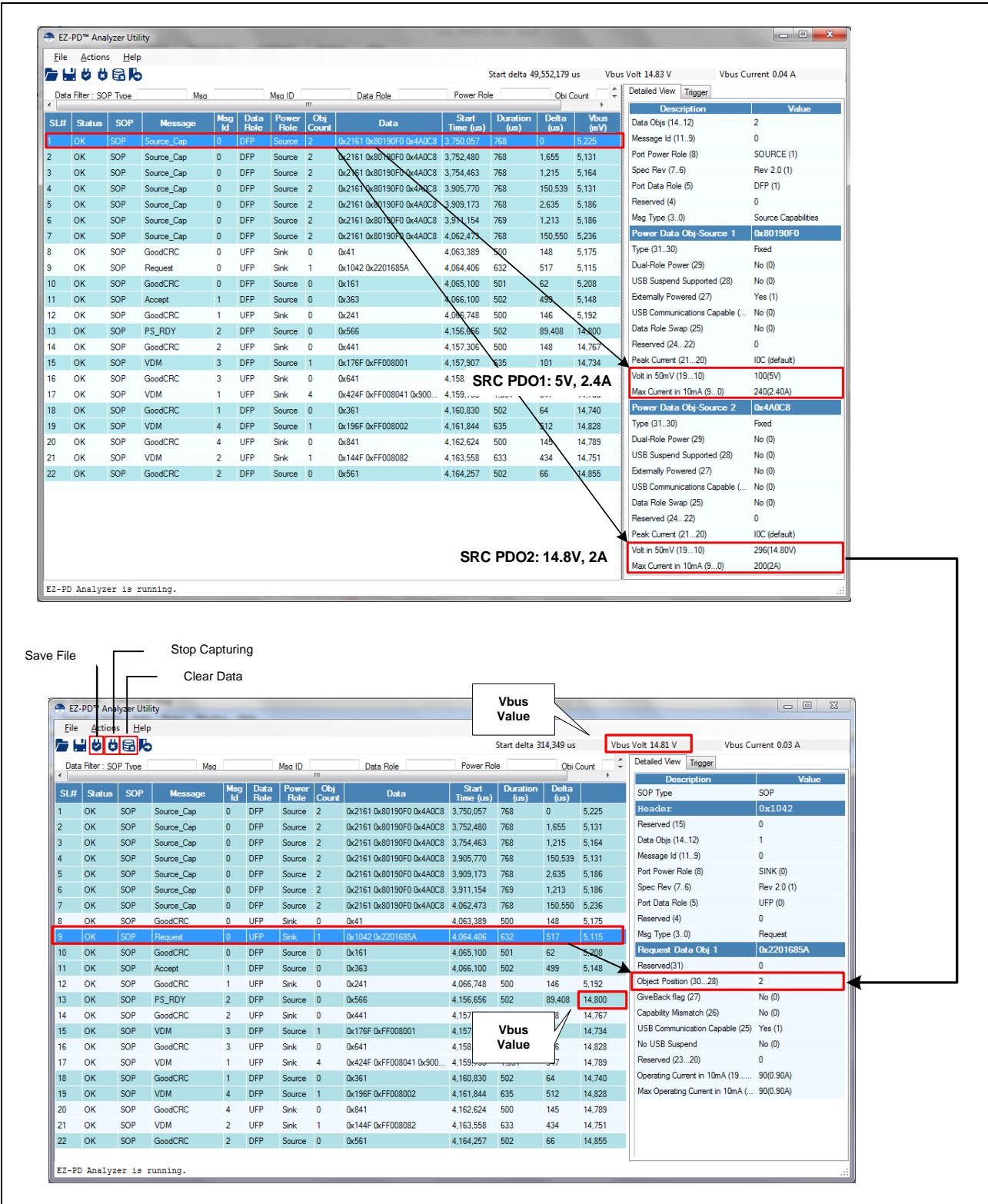## CCG4 DRP Demonstration and Configuration Example

Figure 28   Establishment of Power Contract for 14.8 V

6.3 Modifying CCG4 Firmware

This section demonstrates modifications of the CCG4 device firmware for a notebook application using PSoC Creator. It gives you a step-by-step process to modify the CCG4 firmware for a specific customer's system requirements such as data or display multiplexer settings and system overvoltage and overcurrent limits. The reference firmware for a CCG4 DRP notebook application is available.

6.3.1 Build Environment

PSoC Creator 3.3 SP1 or later is required to edit, compile, download, and debug the firmware for the CCG4 DRP application of the CCG4 device as shown in **Figure 29**. The compiler tool chain is ARM GCC (build 493, provided along with the PSoC Creator build).
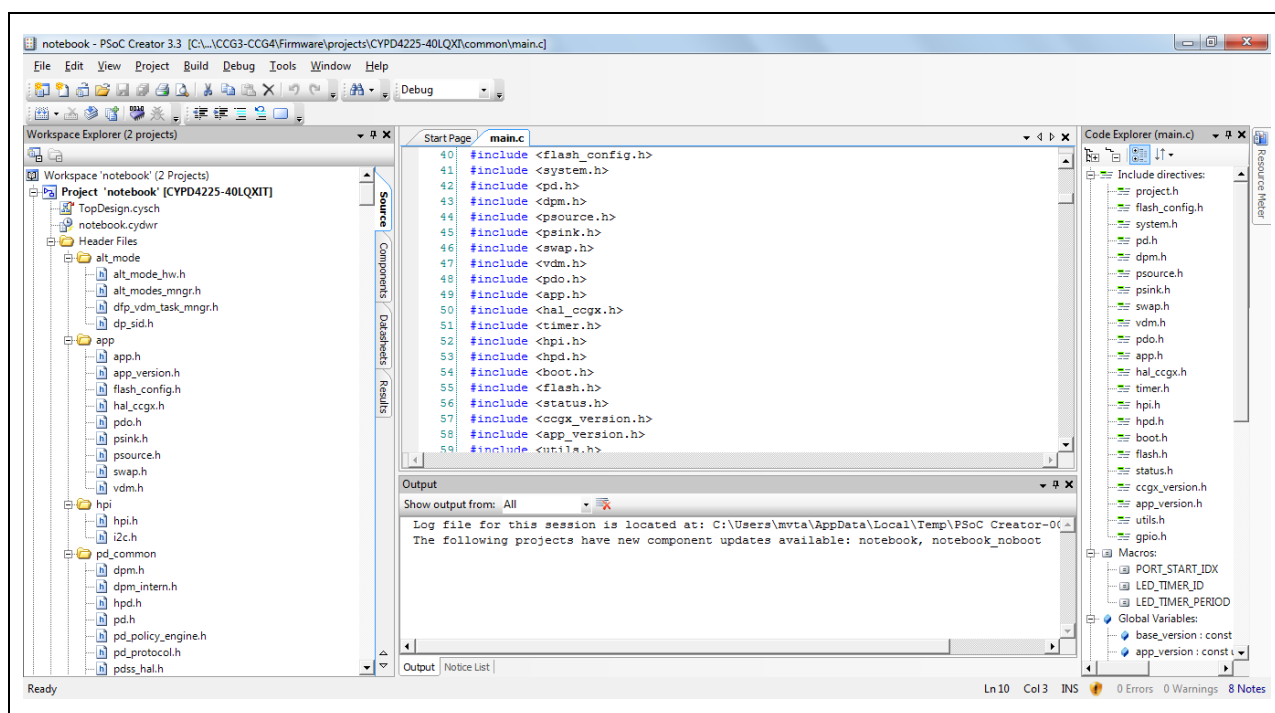
Figure 29   PSoC Creator IDE

6.3.2 Source Structure

- The SDK includes a reference firmware project for a notebook application of the CCG4 device. Refer to section 3.1 ("Using the Reference Projects") of the *CCGx_FW_UserGuide.pdf* for information on the procedure to open the reference projects. You can find the CCGx_FW_UserGuide.pdf at the location in the SDK: <Install_Directory>\Cypress\EZ-PD CCGx SDK\CCG3-CCG4\Documentation

6.3.3 PSoC Creator Workspace File Structure

Figure 30 shows the PSoC Creator workspace file structure for the notebook application using the CCG4 device.
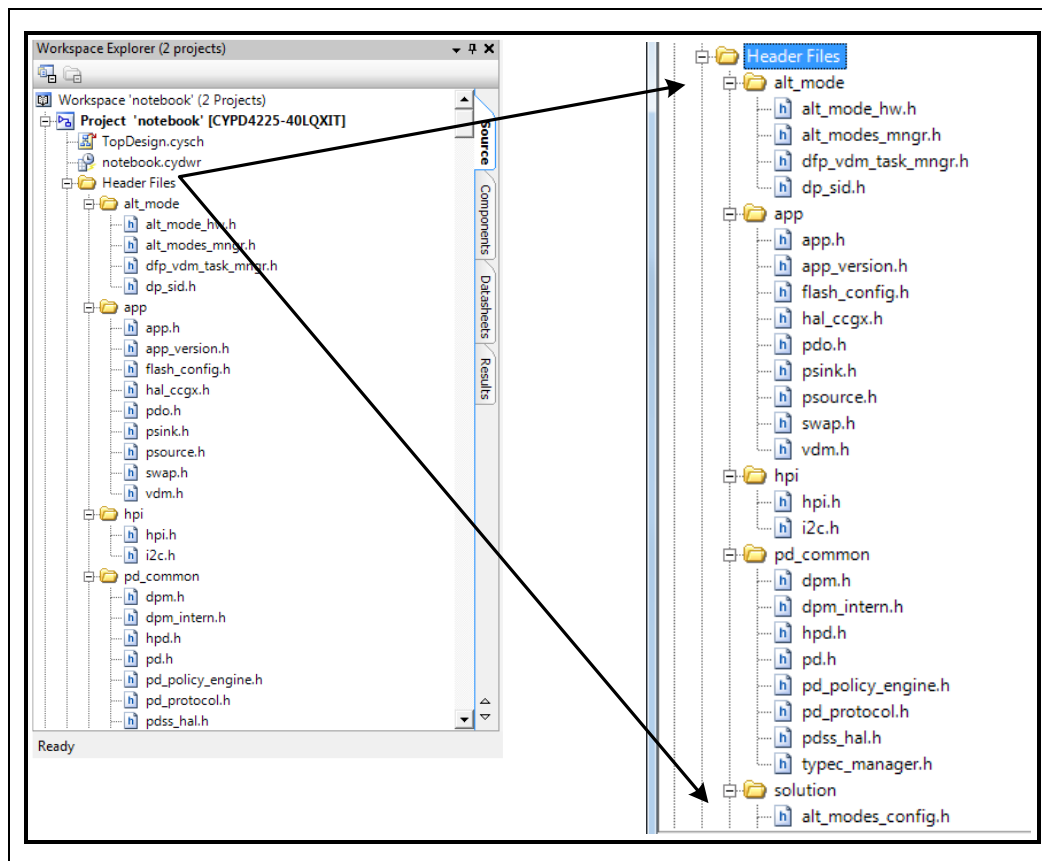

Figure 30   PSoC Creator Workspace Structure for Notebook Application

6.3.4 Build Output

PSoC Creator generates a bootloadable .cyacd file and a Cypress format *.hex* file after building the project.

6.3.5 PSoC Creator Schematic

- The system design of the notebook using the CCG4 device is located in the schematic associated with the PSoC Creator firmware project. This schematic is present in the *TopDesign.cysch* file, which is part of the PSoC Creator project. Refer to section 4 ("Customizing the Firmware Application") of the *CCGx_FW_UserGuide.pdf* for more information on PSoC Creator schematic. You can find the CCGx_FW_UserGuide.pdf at the location in the SDK: <Install_Directory>\Cypress\EZ-PD CCGx SDK\CCG3-CCG4\Documentation

6.3.6 Firmware Versioning

The firmware project source provided in the SDK has an application type string as well as application major and minor version numbers. This information is stored in *app_version.h*. The application string and version numbers can be modified by customers based on their requirements.

6.3.7 Firmware Feature Configurations

The notebook application firmware provided in the SDK has the following configurable features, which can be modified by individual users based on system requirements. These configurable parameters are present in the *config.h* file. **Table 6** provides the list of important pre-compile configurable features available in the notebook application firmware.

Table 6 Configurable Features of CCG4 Notebook Application Firmware (config.h)

Feature	Macro in Config.h	Explanation	Options	Default Setting
CCG4 DeepSleep Mode	`SYS_DEEPSLEEP_ENABLE`	After enabling this feature, the CCG4 device goes into the DeepSleep mode under idle condition to reduce power consumption.	Enable – 1 Disable – 0	Enable-1
Alternate Mode support when CCG4 acts as DFP	`DFP_ALT_MODE_SUPP`	After enabling this feature, the CCG4 device (configured as a DFP) supports various Alternate modes such as DisplayPort, and Audio.	Enable – 1 Disable – 0	Enable-1
Display Port support when CCG4 acts as DFP	`DP_DFP_SUPP`	After enabling this feature, the CCG4 device (configured as a DFP) supports DisplayPort Alternate mode.	Enable – 1 Disable – 0	Enable-1
OverVoltage Protection (OVP)	`VBUS_OVP_ENABLE`	This feature enables VBUS overvoltage protection if the Type-C VBUS exceeds the limit specified by VBUS_OVP_Margin for a negotiated PDO.	Enable – 1 Disable – 0	Enable-1
	`VBUS_OVP_MARGIN`	Sets the OVP percentage margin above the negotiated VBUS voltage	Custom value	20
OverCurrent Protection (OCP)	`VBUS_OCP_ENABLE`	This feature enables VBUS overcurrent protection if the Type-C VBUS current exceeds the system current.	Enable – 1 Disable – 0	Enable-1
Enable LED support	`APP_FW_LED_ENABLE`	This feature enables toggling of an LED to indicate that the CCG4 device is in Active mode.	Enable – 1 Disable – 0	Disable-0

6.3.8 Firmware APIs

- The major functionality of the application firmware can be modified using the EZ-PD Configuration Utility without changing the device firmware. However, changes in the hardware design require the firmware source to be modified. Refer to section 4.2 ("Updating Code to Match of the Schematic") of the *CCGx_FW_UserGuide.pdf* for more details about customizing the firmware application. You can find the CCGx_FW_UserGuide.pdf at the location in the SDK: <Install_Directory>\Cypress\EZ-PD CCGx SDK\CCG3-CCG4\Documentation

CCG4 DRP Demonstration and Configuration Example

Following are a few important firmware APIs that may need to be modified based on the system requirements to start the design:

- OVP settings – The OVP feature in a CCG4-enabled notebook application prevents system damage if VBUS exceeds the maximum voltage negotiated by CCG4. The default firmware sets the OVP trip voltage 20 percent higher than the negotiated voltage at the CCG4 VBUS_MON pin, but this can be varied using the macro in *config.h* as described above. A custom system specific VBUS overvoltage threshold value for either of the Type-C ports can be set by using the API given in **Table 7.** This API allows the user to vary the overvoltage threshold value dynamically with respect to the negotiated PDO. For example, overvoltage threshold value can be set to 5.5 V (1.2 times negotiated PDO voltage, which is 5 V). This "1.2 times" factor can be changed to any custom value (for example, "1.1 times") for higher negotiated PDO voltages such as 20 V.

- GPIO control – CCG4 GPIOs device can be connected to another controller for specific purposes. Toggling of the CCG4 device's GPIOs can be performed by using firmware APIs based on system requirements.

- Display or data multiplexer settings -- The display monitor can have two or four Main Link (Display Port) lanes. The two-lane Main Link configuration supports a raw bit rate up to 10.8 Gbps, and the four-lane Main Link configuration supports a raw bit rate up to 21.6 Gbps. Resolution of the display in the two-lane configuration is lower than that of the display in the four-lane configuration. However, SuperSpeed USB devices can be supported only in the two-lane mode. The system may decide to switch from a four-lane to a two-lane display to enable SuperSpeed device enumeration or it may continue in the four-lane mode. This is implementation-specific. The display mux controller needs to be configured either in two-lane on four-lane mode to support the particular display monitor and enumeration of SuperSpeed USB devices in the system.

Table 7 provides the list of firmware APIs that can be used to modify the settings above mentioned in the notebook application firmware based on system requirements.

Table 7   Firmware APIs

Function	Description	Parameter			Return
set_mux	Sets appropriate MUX configuration based on function parameters	Port	Type-C port number		True if MUX setting passed successfully; false if MUX setting is invalid or contains unacceptable fields.
		cfg	Required mux configuration	Explanation	
			`MUX_CONFIG_2_0`	`USB 2.0 configuration`	
			`MUX_CONFIG_SS_ONLY`	`USB SS configuration`	
			`MUX_CONFIG_DP_2_LANE`	`Two lane DP configuration`	
			`MUX_CONFIG_DP_4_LANE`	`Four lane DP configuration`	
			`MUX_CONFIG_AR_CUSTOM`	`AlpineRidge custom configuration`	

Function	Description	Parameter		Return
		Custom_data	Additional data in the case of custom AR mux configuration	
pd_get_vbus_adc_level	Dynamically determines the ADC level that corresponds to the VBUS OVP threshold value based on "Volt" and "Per" parameters	Port	Type-C port number	Returns the ADC level corresponding to the OVP threshold value
		Adc_id	ADC ID	
		Volt	Voltage in 50 mV	
		Per	Margin on VBUS voltage.	
gpio_set_value	Updates the output state of a GPIO pin to the required state	Port_pin	Pin to be updated	None
		Value	Value to drive on the pin: 0 = LOW, 1 = HIGH	
gpio_read_value	Gets the current GPIO state	Port_pin	Pin to be queried	Current state of the pin.

- Refer to *CCGx_FW_1_0_0_API_Guide.pdf* for details on the firmware APIs. You can find the CCGx_FW_UserGuide.pdf at the location in the SDK: <Install_Directory>\Cypress\EZ-PD CCGx SDK\CCG3-CCG4\Documentation

6.3.9 Building Firmware Using PSoC Creator

This section demonstrates a simple example of modifying the notebook application firmware. In this example, the VBUS _OVP_MARGIN parameter is modified from the default to a custom value, as explained in **Table 6**. The following steps describe how to build application firmware for CCG4 using PSoC Creator.

Before modifying the firmware using PSoC Creator, copy the entire source project from the default location (for example, *<Install_Directory>\Cypress\EZ-PD CCGx SDK\CCG3-CCG4\Firmware\projects\CYPD4225 40LQXI)* to another location of your choice.

Note: The default *<Install_Directory>* is *C:\Program Files* for a 32-bit Windows PC and *C:\Program Files (x86)* for a 64-bit Windows PC.

Make all files writable in the copy of the project.

Open the workspace by double-clicking on the *notebook.cywrk* file in the copy of the project.

Update the VBUS_OVP_MARGIN parameter from 20 to 25 as shown in **Figure 31**. This parameter is located in the *config.h* file, located at *fw/header_files/solution/config.h*.

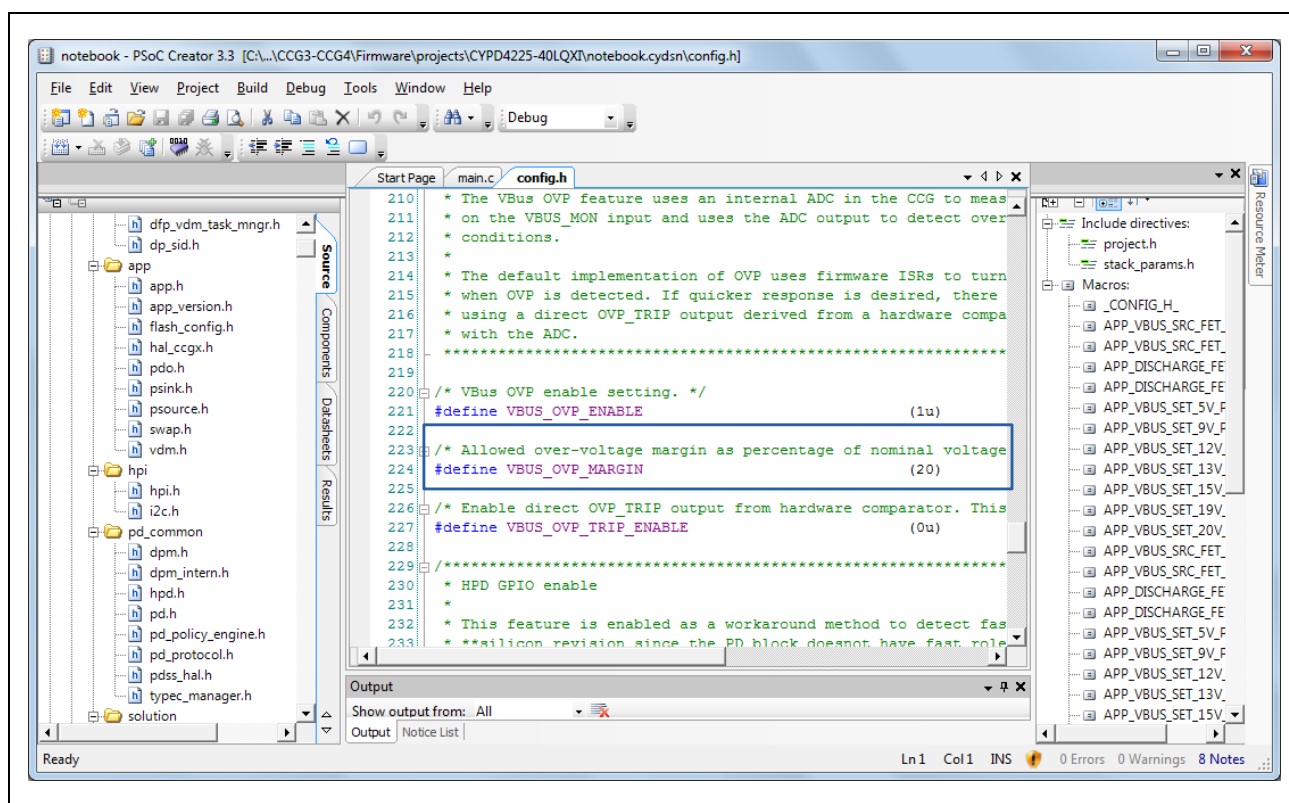## CCG4 DRP Demonstration and Configuration Example

Figure 31   Updating the VBUS_OVP_Margin Parameter

Select the Debug/Release build setting as shown in **Figure 32**. If firmware debug is in progress, select **Debug**. Select **Release** for final release of the firmware.
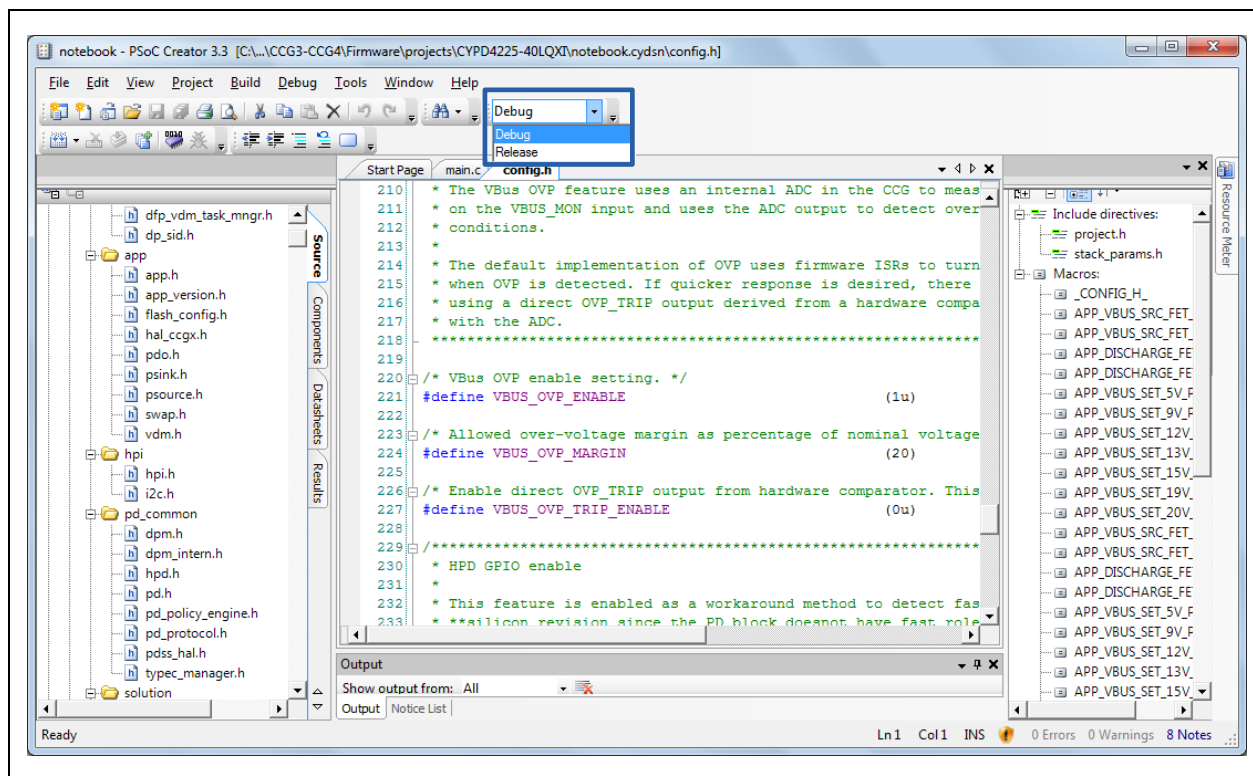

Figure 32   Figure 1. Selection of Debug/Release

Build the firmware by choosing Build > Build notebook as shown in **Figure 33** .
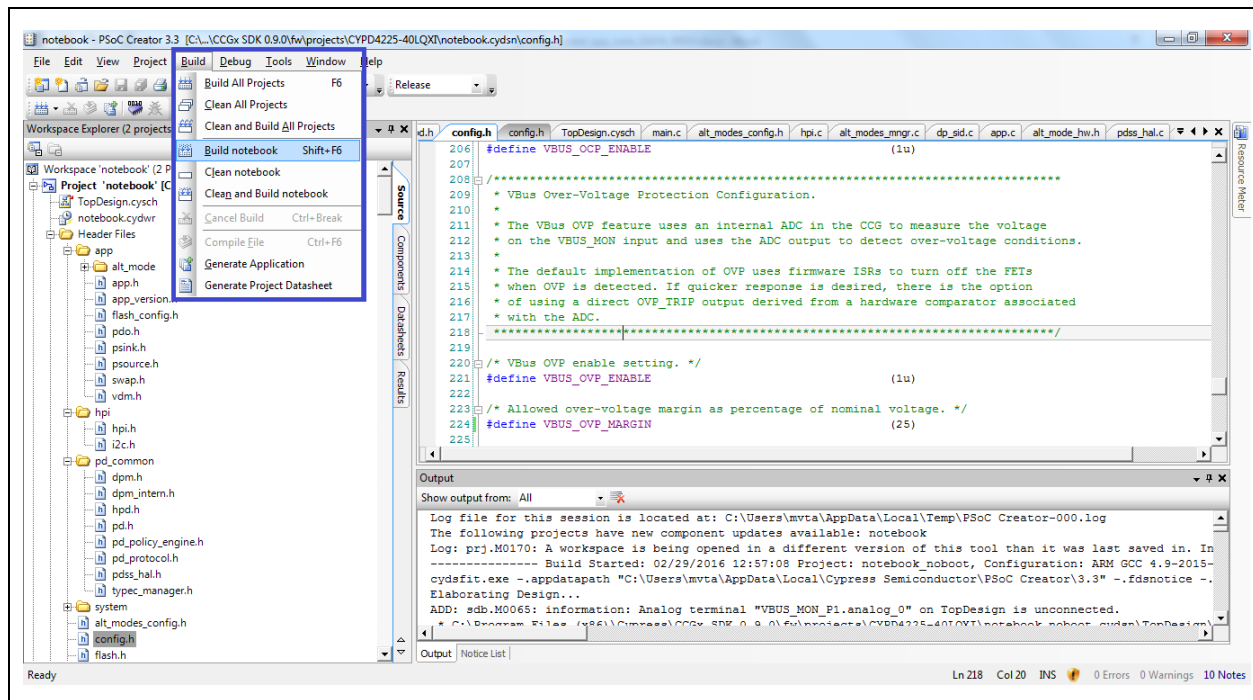
Figure 33   Building the Firmware

After the firmware is successfully built, PSoC Creator generates a *.cyacd* file and a Cypress format *.hex* file.

6.3.10   Programming Firmware in CCG4 Device

There are two methods to program the firmware into the CCG4 device:

- Using EZ-PD Configuration Utility (using the *.cyacd* file)
- Using MiniProg3 and PSoC Creator/Programmer (using the *.hex* file)

Refer to chapter 4 of the **CY4541 EZ-PD CCG4 EVK guide** for detailed information about firmware download the *.cyacd* file using the EZ-PD Configuration Utility.

Refer to the knowledge base article **KBA96477** for detailed information about downloading the firmware *.hex* file into the CCG4 device using PSoC Programmer or PSoC Creator.

Revision history

Document version	Date of release	Description of changes
**	2016-03-17	New application note
*A	2017-06-15	Updated Figures 4,9,10,13,15,19 Updated section 5.3 Updated template
*B	2017-04-19	Updated logo and Copyright.
*C	2021-03-15	Updated to Infineon template.

Trademarks
All referenced product or service names and trademarks are the property of their respective owners.