EZ-PD™ configuration utility user manual

About this document

Scope and purpose

This document describes the usage of EZ-PD™ configuration utility, a Microsoft Windows application that guides through the process of configuring and programming CCGx devices.

Intended audience

This document is intended for embedded developers using EZ-PD™ CCGx power/host software development kits.

Table of contents

1 Introduction

1.1 EZ-PD™ device family

EZ-PD™ controllers are fully programmable USB Type-C port controllers that are suited for all USB-PD applications such as cables, dongles, notebooks, monitors, and power adapters.

Currently the CCG1, CCG2, CCG3, CCG4, CCG3PA, CCG3PA-N, CCG3PA2, PAG1S, CCG5, CCG5C, CCG6, ACG1F, CCG6DF, CCG6SF, CCG7D, and CMG1 generations of EZ-PD™ controllers are available. These controllers are collectively called CCGx controllers in this document.

1.2 EZ-PD™ configuration utility

The EZ-PD™ CCGx controller is a highly configurable and programmable solution. The chip can be configured using parameters stored in the internal flash memory. These parameters are to be chosen and programmed by customers according to their use cases and requirements.

The EZ-PD™ configuration utility is a Microsoft Windows application that guides the CCGx user through the process of configuring and programming the chip. The utility works in tandem with the supplied hardware, which hosts the CCGx controllers along with a USB interface.

The Graphical User Interface (GUI) of the EZ-PD™ configuration utility allows you to intuitively select and configure the parameters for your application.

1.3 Software pre-requisites

Table 1 lists the software pre-requisites needed to run the configuration utility.

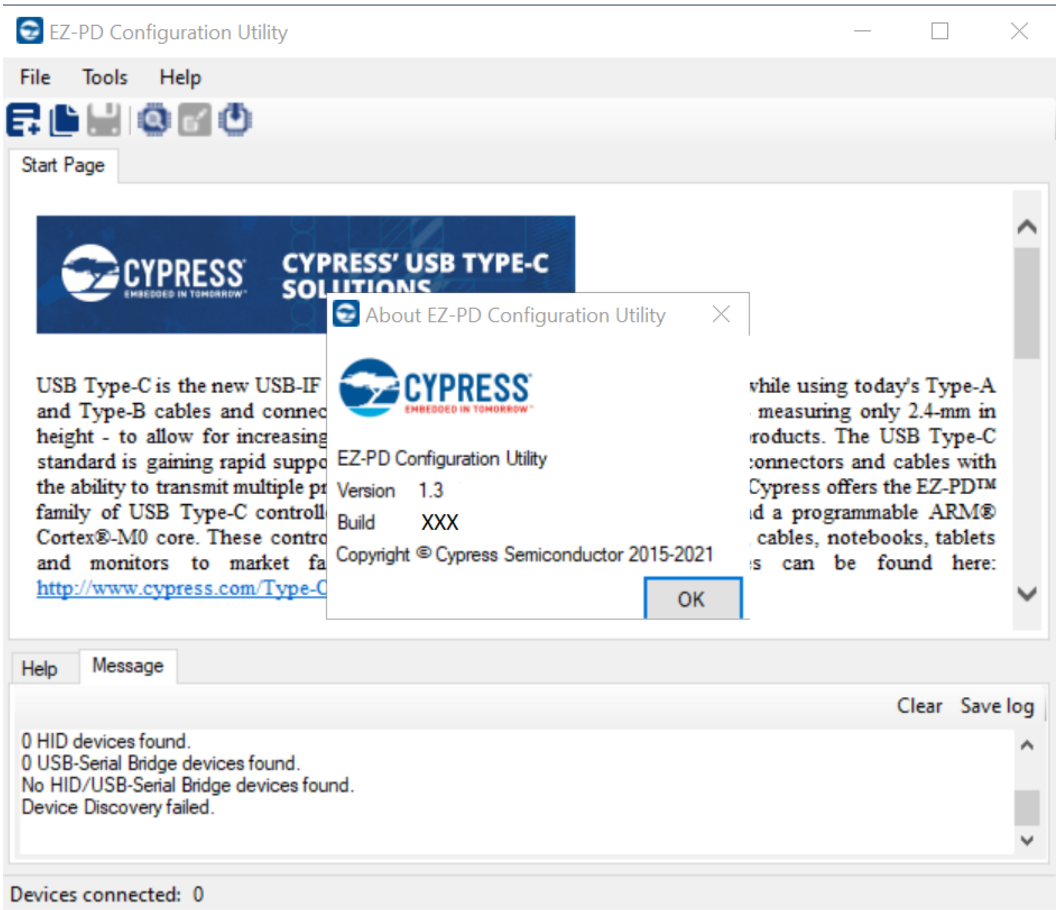
Table 1 Pre-requisites

#	Software	Version
1	Operating System	Microsoft Windows 7 or later
2	.NET Framework	.NET Framework 2.0 or later

Note: Microsoft .NET framework can be downloaded and installed from ***Microsoft.com****.*

2 EZ-PD™ configuration utility overview

You can install the EZ-PD™ configuration utility through a single click installer. After successful installation, you can start the application from the desktop icon for Windows 7, or from the Apps menu in case of Windows 8 or Windows 10.



USB Type-C is the new USB-IF … while using today's Type-A and Type-B cables and connec… measuring only 2.4-mm in height - to allow for increasing… roducts. The USB Type-C standard is gaining rapid suppo… connectors and cables with the ability to transmit multiple pr… Cypress offers the EZ-PD™ family of USB Type-C controll… d a programmable ARM® Cortex®-M0 core. These contro… cables, notebooks, tablets and monitors to market fa… s can be found here: http://www.cypress.com/Type-C

EZ-PD Configuration Utility
Version 1.3
Build XXX
Copyright © Cypress Semiconductor 2015-2021

```
0 HID devices found.
0 USB-Serial Bridge devices found.
No HID/USB-Serial Bridge devices found.
Device Discovery failed.
```

Devices connected: 0

Figure 1 Welcome screen with version information

2.1 Usage flow

The utility usage flow typically consists of three steps:

1. **Creating configuration**
2. **Selecting parameters**
3. **Configuring device**

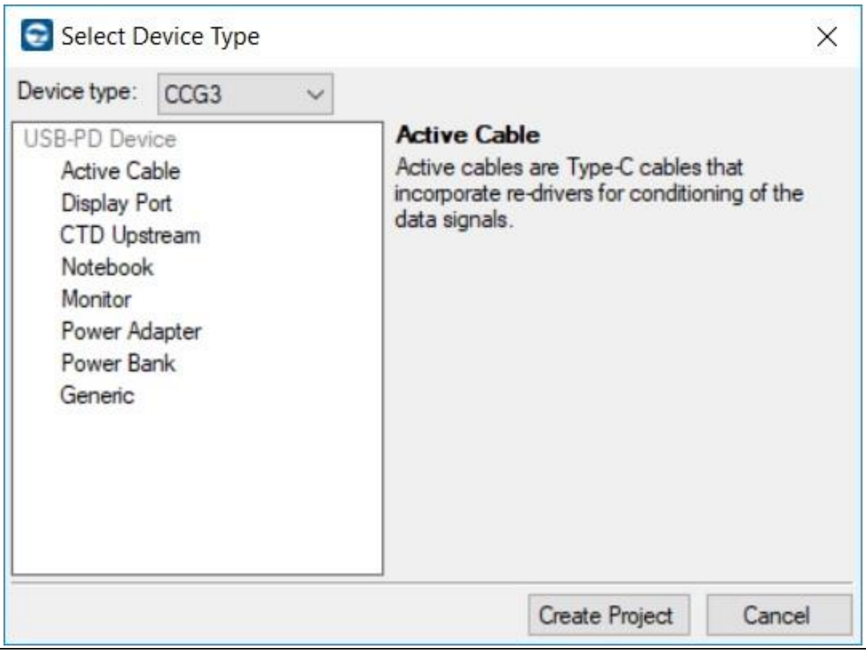
In the first stage, you select the target application to be configured. The applications are categorized by device family. To select an application that is available in a specific device family, select the corresponding family from the drop-down menu.

After selecting the device type, you move to the **Selecting parameters** stage. All configurable parameters of the active application and their default values are shown on the **Configuration** tab of the GUI. You can view, modify, and save the parameters from the **Configuration** tab.

After selecting and saving the parameters, you move to the final **Configuring device** stage. In this stage, the saved configuration is programmed onto the internal flash memory on the EZ-PD™ CCG controller.

2.1.1 Creating configuration

To start a new configuration, follow the menu path **File** > **New**. In the Select Device Type dialog, select the CCGx device from the **Device type** drop-down list. The USB-PD applications supported by the selected CCG device are listed.



Active Cable

Active cables are Type-C cables that incorporate re-drivers for conditioning of the data signals.

Figure 2 Creating a new project

Since the USB-PD applications supported vary across the product families, select the CCG product family first before selecting the type of USB-PD application. The utility filters the applications available for selection based on the product family selected.

Table 2 lists the CCGx applications supported by the current version of the utility.

Table 2 Supported CCGx applications

	Application			
Product family	**Electronically marked Type-C cables (passive or active EMCA)**	**Type-C to display (DisplayPort, HDMI or VGA) dongles**	**Type-C port controllers for Notebooks**	**Type-C power applications (power adapter, power bank, car charger, and so on)**
CCG1	✓	✓	✓	✓
CCG2	✓	✓	✓	✓
CCG3	✓	✓	✓	✓
CCG4			✓	✓
CCG5/CCG5C			✓	
CCG6/CCG6DF/CCG6SF			✓	
CCG3PA				✓
ACG1F			✓	

	Application			
Product family	**Electronically marked Type-C cables (passive or active EMCA)**	**Type-C to display (DisplayPort, HDMI or VGA) dongles**	**Type-C port controllers for Notebooks**	**Type-C power applications (power adapter, power bank, car charger, and so on)**
PAG1S				✓
CCG3PA2	✓			✓
CMG1	✓			
CCG3PA-N				✓
CCG7D				✓

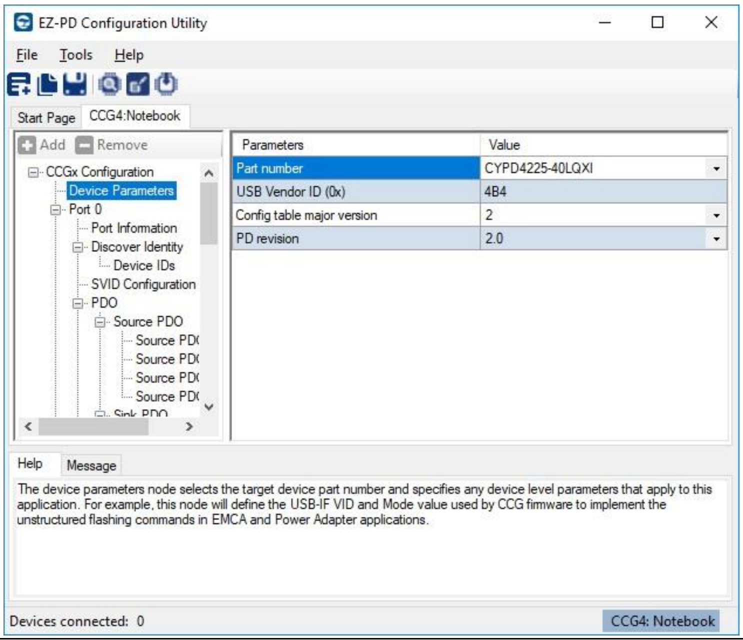
The leaf nodes on the tree represent devices that can be configured. Select the device, from the left pane, and click **Create Project** to configure the device as shown in **Figure 2**.

Since the USB-Serial devices are used to implement the “Billboard controller”, the application type ‘Billboard controller’ is available when the **USB Serial** device family is selected.

2.1.2 Selecting parameters

The **Configuration** tab appears only after the device type for the configuration is selected. The left pane of the **Configuration** tab displays parameter groups available for the configuration of the selected device, in a tree structure. For devices with multiple Type-C ports, configure each port independently by entering parameter values under the nodes of each ports.

You can select a group, from the left pane, to view the parameters and the corresponding values on the right pane. To modify the parameters, you can either select the values from the drop-down list or enter the values in the textbox. The utility performs validity checks on the values entered in the textbox.



Parameters	Value
Part number	CYPD4225-40LQXI
USB Vendor ID (0x)	4B4
Config table major version	2
PD revision	2.0

The device parameters node selects the target device part number and specifies any device level parameters that apply to this application. For example, this node will define the USB-IF VID and Mode value used by CCG fimware to implement the unstructured flashing commands in EMCA and Power Adapter applications.

Figure 3 Selecting device parameters

To view the description of a parameter, hover the mouse over the parameter name. A tool-tip will display the description. You can also click a parameter name and view the description in the **Help** tab at the bottom of the dialog.

After updating the parameters, click **Save As** and save the configuration in the specified folder. The modifications will be saved as an XML file.

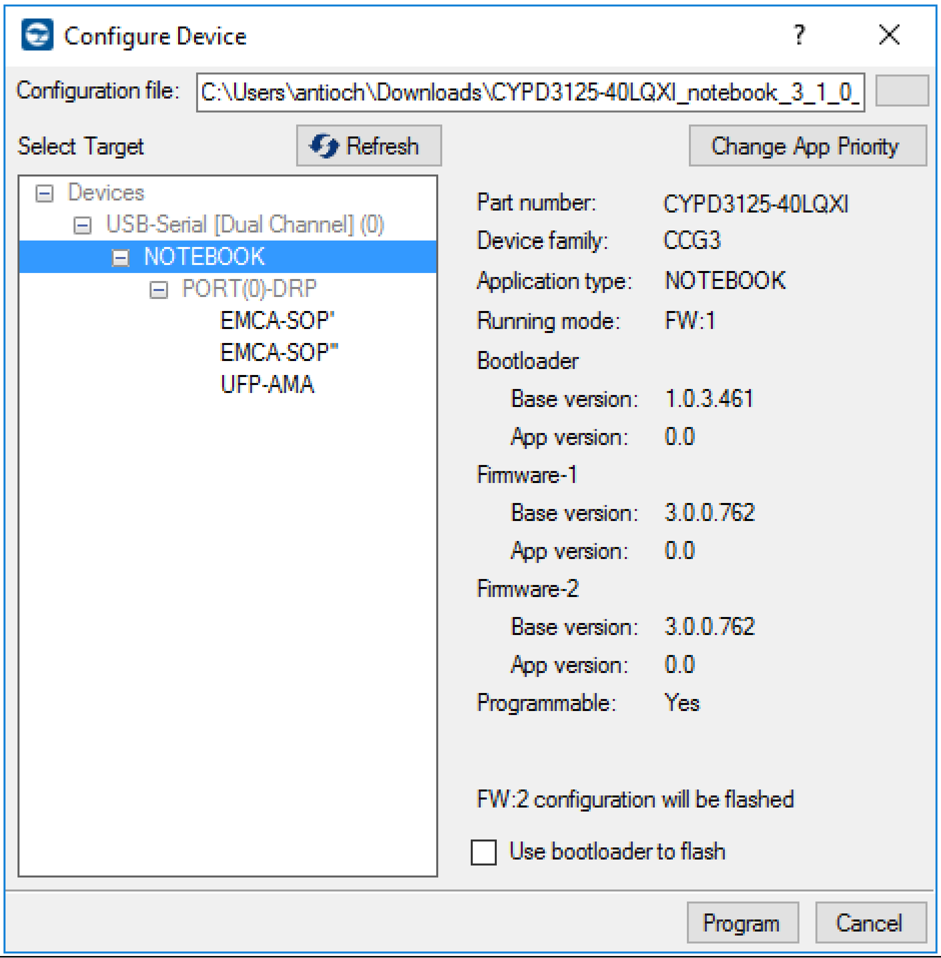
To load the saved configuration, follow the menu path **File** > **Open**.

Note that the XML file cannot be programmed to the device. A file with *.cyacd* extension is saved in the same folder along with the XML file. Use this file to program the configuration to the device.

This version of the EZ-PD™ configuration utility also generates a C source file containing the configuration data in the form of an array. You can import this source file to a CCGx SDK firmware project workspace, so that a firmware binary with the desired default values can be compiled.

2.1.3 Configuring device

To program the configuration to the device flash, follow the menu path **Tools** > **Configure Device**. In the Configure Device dialog, select the target device to be programmed and initiate programming.



Configuration file: C:\Users\antioch\Downloads\CYPD3125-40LQXI_notebook_3_1_0_

Part number: CYPD3125-40LQXI

Device family: CCG3

Application type: NOTEBOOK

Running mode: FW:1

Bootloader

Base version: 1.0.3.461

App version: 0.0

Firmware-1

Base version: 3.0.0.762

App version: 0.0

Firmware-2

Base version: 3.0.0.762

App version: 0.0

Programmable: Yes

FW:2 configuration will be flashed

□ Use bootloader to flash

Figure 4 Device configuration dialog

3 Hardware setup

3.1 Programming methods

Different USB-PD applications have different requirements in terms of configuration parameters as well as the means of updating the configuration.

3.1.1 Programming over I2C

Devices such as Type-C Notebooks or Monitor adapters will have an Embedded Controller (EC) that can talk to the CCGx controller through a control interface such as I2C. In this case, the CCGx controller will be configured with a program which can receive flash read/write commands through the I2C interface.

On the USB-PD Host and Client boards, EC is emulated by a USB-Serial bridge device. The USB-Serial bridge device receives a set of USB-based vendor commands, which are translated into I2C transfers performed on the CCGx controller. **Figure 5** shows a block diagram of the setup where the CCGx device can be programmed using USB – I2C bridge.

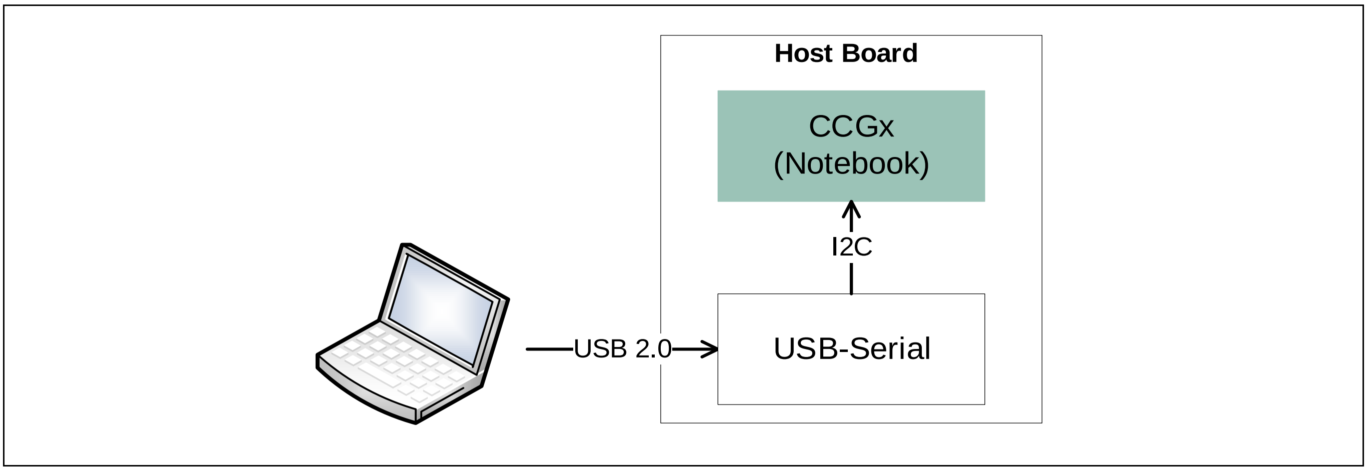


Figure 5 USB to I2C based programming setup

The **CY4501 CCG1 Development Kit** is an example of a USB-PD Host (Downstream Facing Port (DFP)) implementation using the CCG1 controller. This board makes use of a USB-Serial controller to allow the CCG1 device to be controlled through the USB port.

The **CY4531 EZ-PD™ CCG3 Development Kit** and **CY4541 EZ-PD™ CCG4 Development Kit** are USB-PD Host implementations using the CCG3 and CCG4 controllers, respectively.

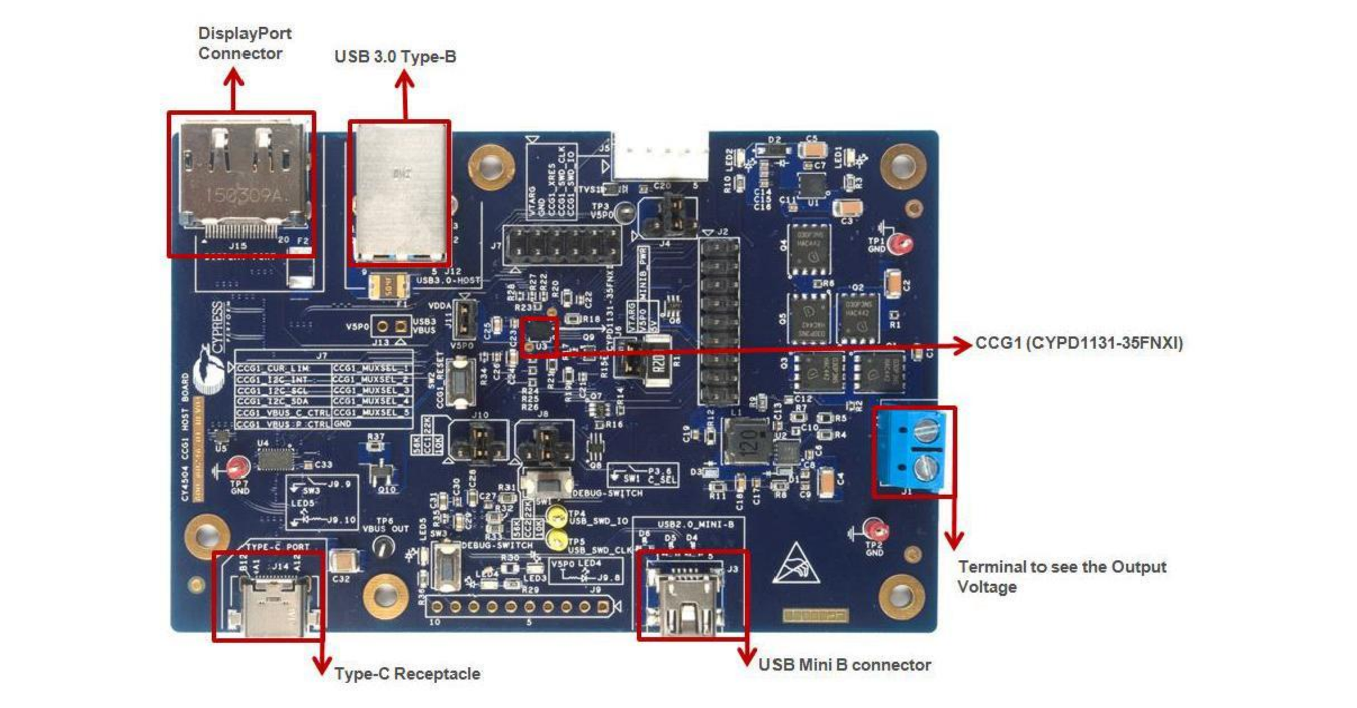


Figure 6 CY4504 - CCG1 based USB-PD host board

3.1.1.1 Programming Type-C to display dongle controllers

Devices such as display dongles with no standard USB functionality will have a USB Billboard device connected to the CCGx controller. This Billboard device will be connected to the CCGx controller through I2C, and can be used to program the CCGx controller. The **CCG1 USB Type-C to DisplayPort Cable Solution** shown in **Figure 7** uses this scheme to program the CCG1 over the I2C interface.

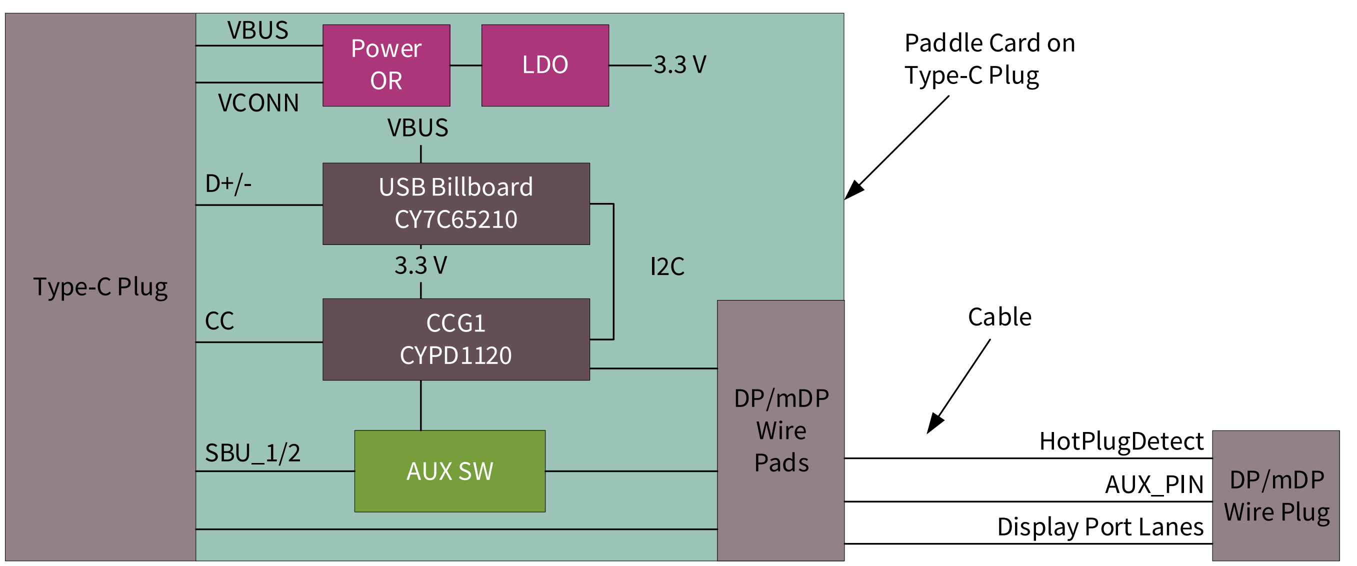


Figure 7 USB Type-C to DisplayPort dongle

Figure 7 shows the programming setup for dongle programming. The CCGx host board is used to make the USB connection between a Type-A USB port on the host computer and the USB Type-C port on the display dongle to be programmed. The actual device programming is done through the CY7C65211 based Billboard controller on the display dongle.

Figure 8 shows the programming setup used for programming the Type-C to Display dongles based on CCG1 and CCG2 devices.

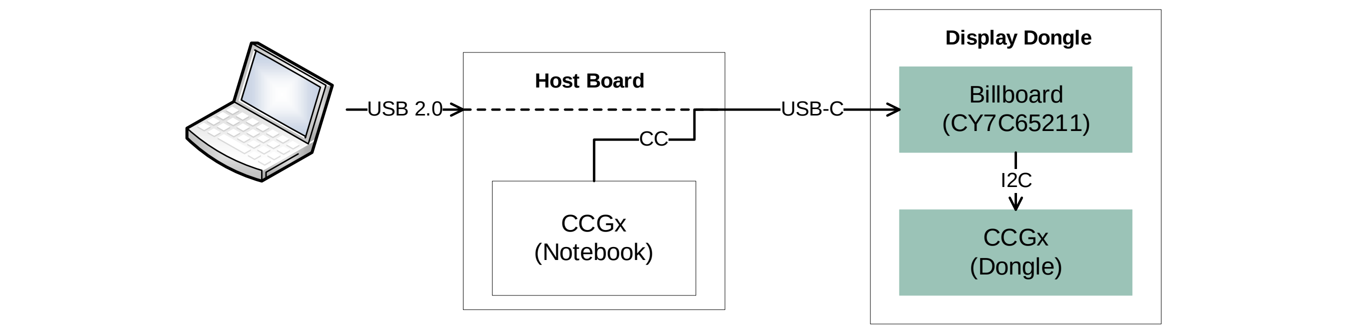


Figure 8 Programming of Type-C to display dongles

3.1.2 Programming over CC channel

Devices such as Type-C Cable (EMCA) controllers and USB Type-C power adapters do not have an EC which can be used to read/update the configuration, and the USB-PD CC channel is the only data interface for the CCGx controller. In such cases, the configuration read, configuration update, and firmware update are performed using Vendor Defined Messages (VDMs) that are sent through the CC channel in the USB Type-C interface. The target CCGx controller is configured to receive flash read/write commands in the form of USB-PD unstructured VDM messages.

As in the earlier case, the USB-Serial Bridge is used to receive USB vendor commands and translate them into a set of I2C based commands. These I2C commands are addressed to a CCGx based flashing controller, which translates them into the required USB-PD messages.

Standard DFP implementations can only initiate USB-PD commands when a USB-PD power contract is in place. Since the DFP cannot setup a power contract while a power consumer (Upstream Facing Port (UFP)) is not present; programming the CCGx based EMCA cable requires both a host (DFP) and client (UFP) to be connected.

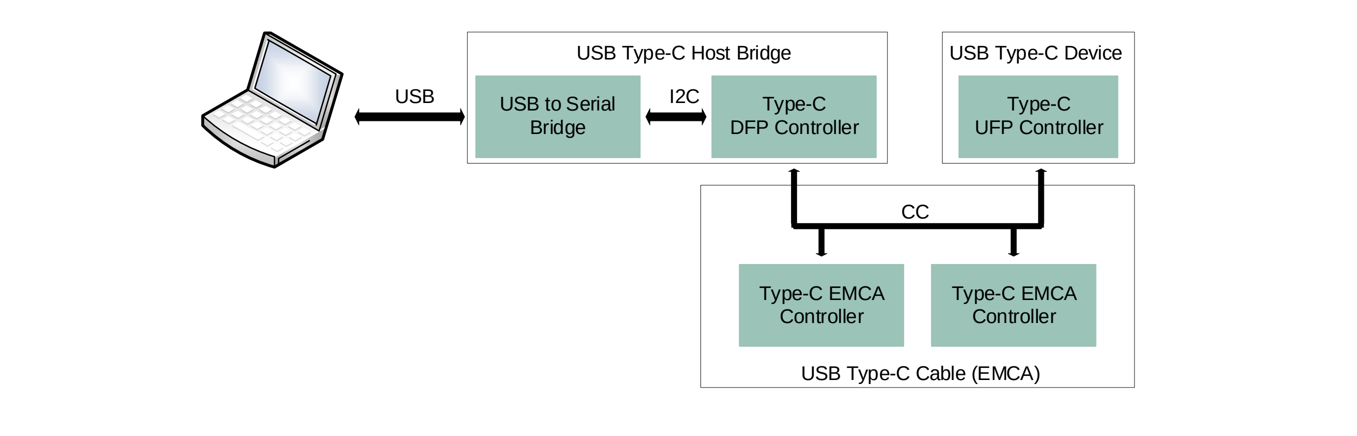


Figure 9 Setup for Programming through the USB-PD CC channel

The CY4501 CCG1 development kit provides CCG1-based host and client boards that can be used in this setup.

Power adapter designs are programmed by connecting them to the CCGx host board using a Type-C cable or using the Type-C plug on the power adapter itself.

Note: *If two Dual Role Power (DRP) controllers are connected by a Type-C cable, and both DRP controllers are connected to PC through serial bridge, then device tree will have duplicate devices and will lead to repeated role swaps, slowing the device configuration. So, it is recommended to connect only one of the DRP controllers to the PC running the configuration utility in such situations.*

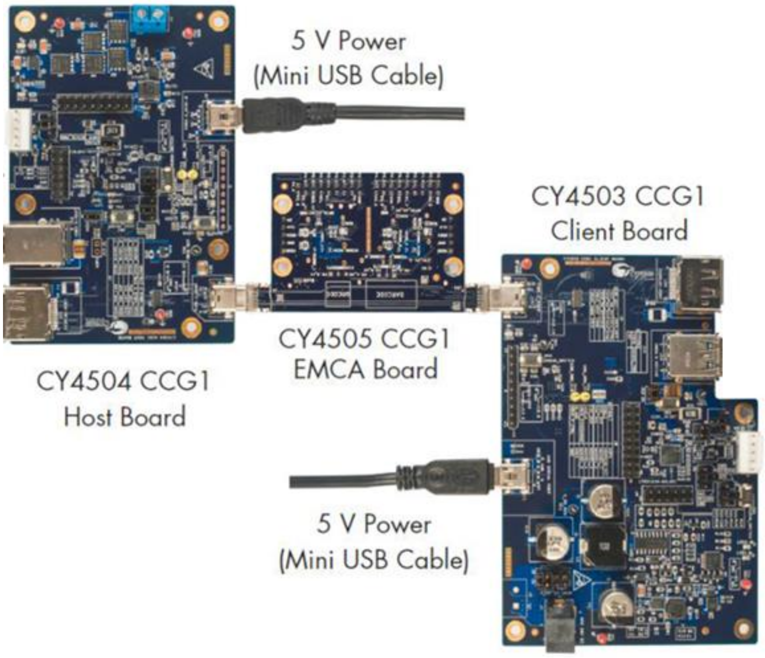


Figure 10 Programming over CC using CY4501 boards

Note: *The CC firmware update feature is not intended for field firmware update. See* ***KBA230192****.*

3.2 Programming setup

3.2.1 Programming EMCA (cables)

The setup shown in **Figure 10** is used for EMCA programming. The CY4505 CCG1 EMCA board shown in **Figure 10** can be replaced by the target EMCA that needs to be programmed.

The CY4504 CCG1 Host board should be programmed with a Notebook DRP firmware image. The *Firmware\Host* folder in the EZ-PD™ configuration utility installation contains a copy of the host (notebook) firmware that can be used.

The CY4503 CCG1 Client board should be programmed with a Monitor (Client DRP) firmware image. The *Firmware\Client* folder in the EZ-PD™ configuration utility installation contains a copy of the client firmware that can be used.

See **CY4501 Kit User Guide** for details on how to configure these boards and program them using the MiniProg programmer.

Once the Host and Client boards have been configured and programmed, both can be connected to USB hosts to power them up. The host board needs to be connected to the computer running the EZ-PD™ configuration utility, while the client board only needs to be powered through some source.

Later sections in this document provide instructions on how to read configuration and program EMCA devices.

3.2.2 Batch programming of EMCA cables

There may be cases where you want to program a large number of EMCA cables with a standard set of configuration parameters. It would be cumbersome to keep connecting the cables to the host and client boards.

To facilitate easier batch programming of EMCA cables, a custom host firmware implementation is provided with the utility. The CY4504 Host board or CY4503 Client board can be programmed with the custom flashing firmware found in the *Firmware\FlashUtil\* folder. Separate files are provided for use on the host and client boards.

Once this flashing firmware has been loaded on the host board, there is no need for the client board to be connected for EMCA programming. The required setup is shown in **Figure 11**.

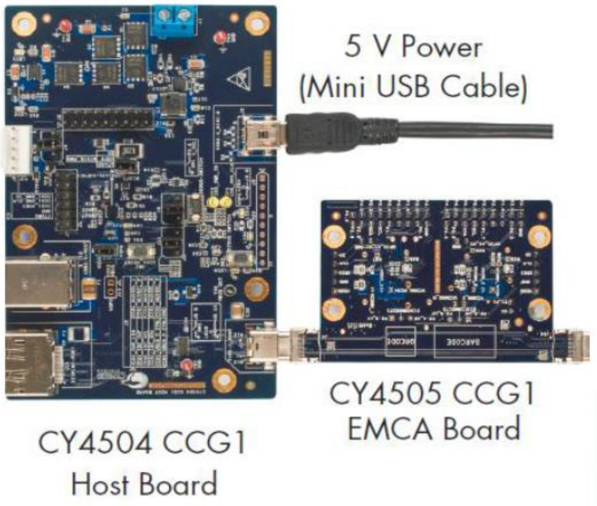


Figure 11 Setup for batch programming of EMCA

3.2.3 Programming Type-C dongles and Billboard devices

Programming a Type-C to Display (DisplayPort or HDMI) dongle requires access to the USB and USB-PD CC channels on the CCGx controller. The CC channel will be connected to the CCGx based host board as in the earlier case. The USB data pins are connected from the Billboard Controller to the PC host (**Figure 11**).

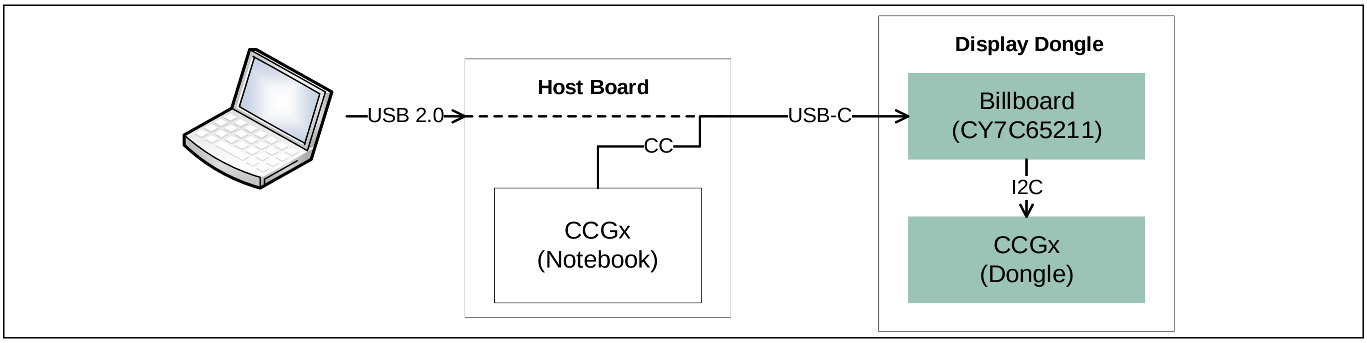


Figure 12 Programming Setup for Type-C Dongle with external Billboard device

The CCGx host is only used to power the CCGx target device through the VConn supply. The actual programming is done through USB Vendor Commands addressed to the Billboard Device.

In the case of the CY4504/CY4541 CCGx Host board, the USB data lines are also connected through the host board itself to the Type-C connector.

In case of CCG3 dongle devices, the internal USB-FS block present on CCG3 chip is used for providing Billboard interface. As there is no external Billboard device, USB-C pins are directly connected to CCG3 dongle as shown in **Figure 13**.

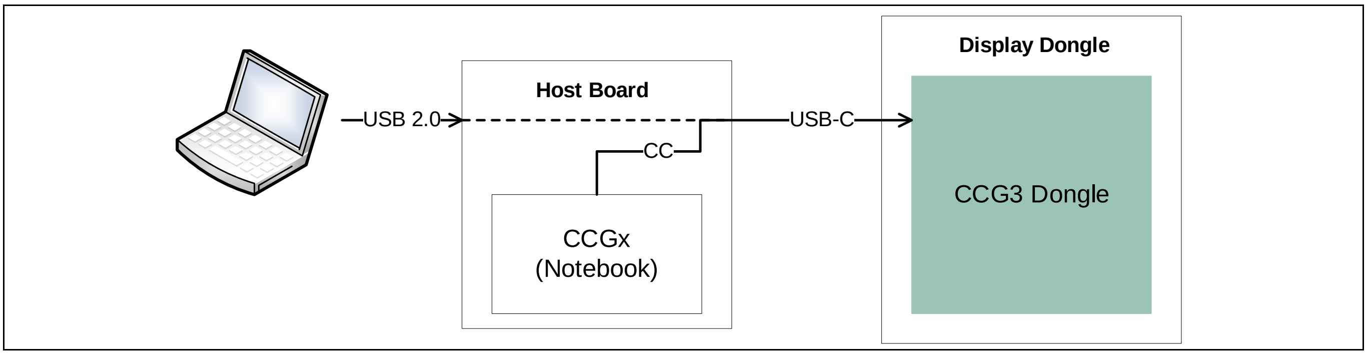


Figure 13 Programming setup for Type-C dongle with Internal On-chip Billboard interface

3.2.4 Programming Notebook Type-C port controllers

The CCGx Notebook port controller can be programmed directly by connecting it to the host computer using the Mini-USB port on the host board.

See the user guides for the respective kits for more detailed programming instructions and connections.

3.2.5 Programming Type-C power adapters

Type-C Power Adapter solutions built using the CCGx controllers need to be programmed through the CC channel, as they do not provide an I2C interface for device programming.

The setup for programming power adapter designs is similar to that used for EMCA programming. The CCGx client board will be replaced by the power adapter to be programmed as shown in **Figure 14**.

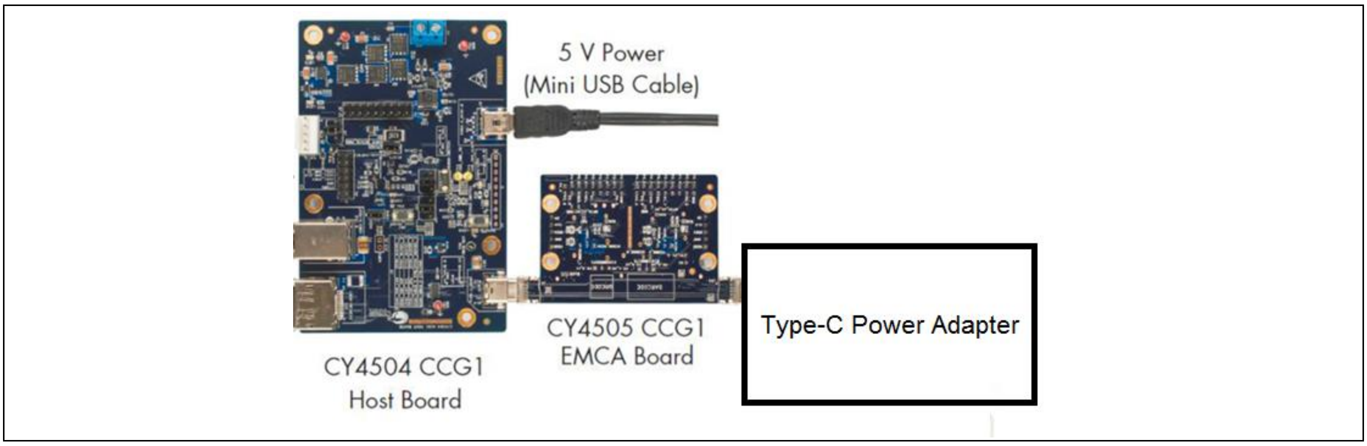


Figure 14 Setup for programming Type-C power adapters

3.2.6 Programming manufacturing test kit hardware

The Manufacturing Test Kit (MTK) hardware consists of the CY4532 EZ-PD™ CCG3PA power board (ensure that the CY4532 EZ-PD™ CCG3PA EVK Main board is detached from the Power board). The MTK hardware will be programmed over I2C. The setup used for programming the MTK hardware is shown in **Figure 15**.

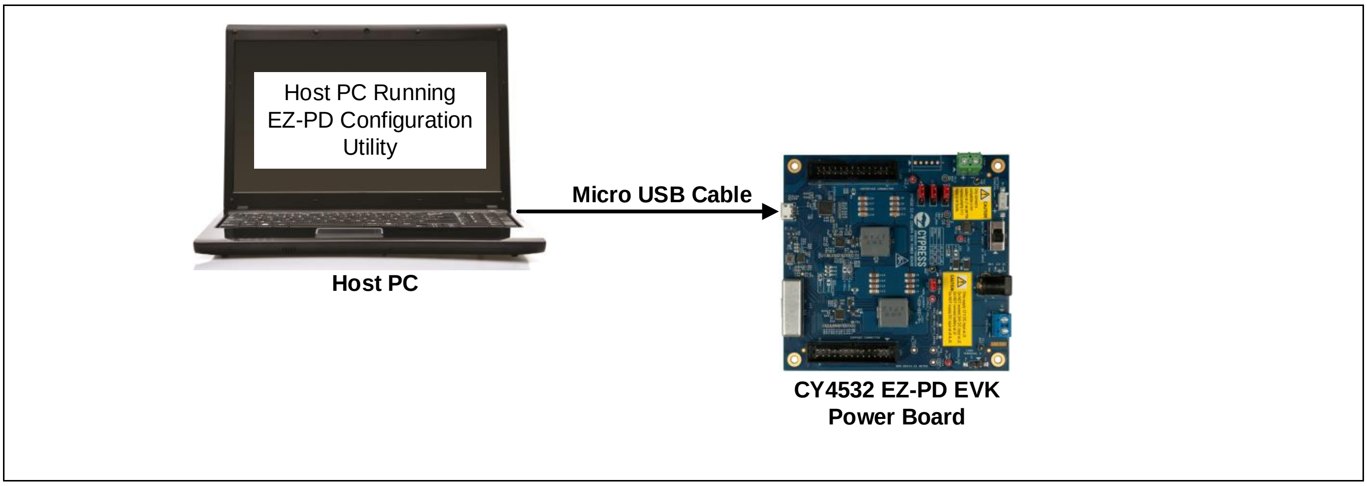
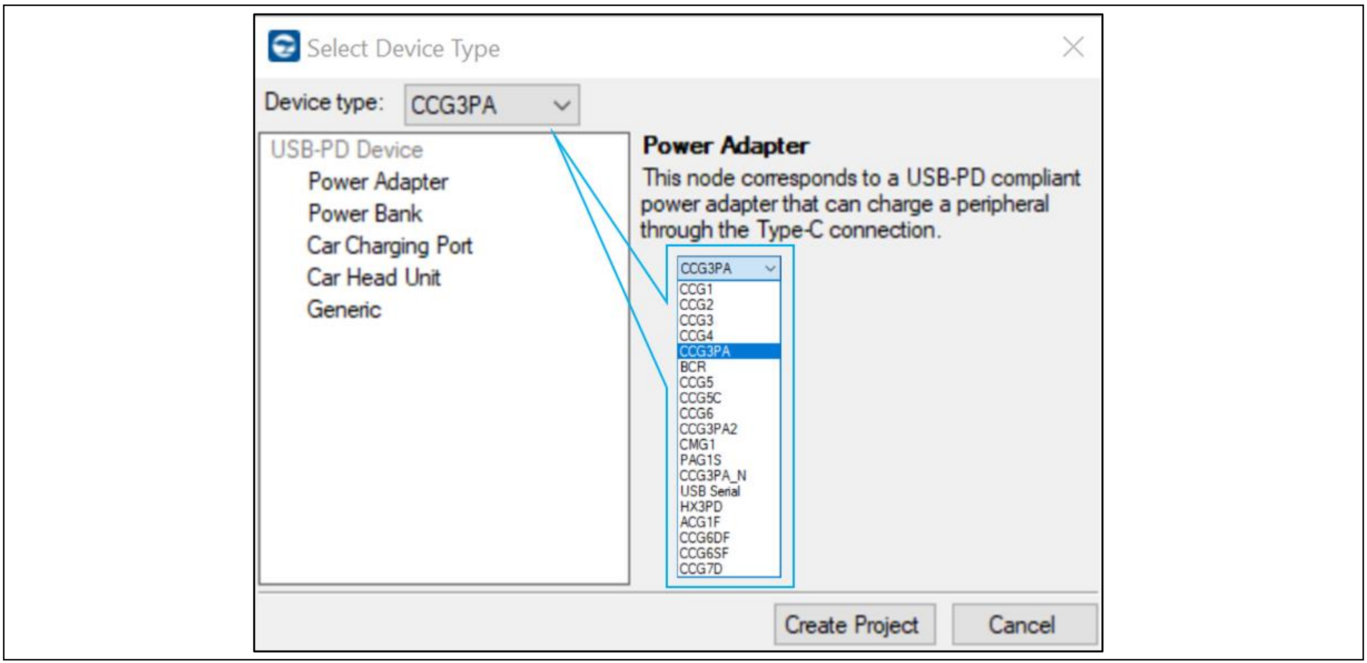


Figure 15 Programming setup for MTK hardware

4 GUI usage

To create a new device configuration, follow the menu path **File** > **New**. Then, select the CCG product family (CCG1/CCG2/CCG3/CCG4/CCG3PA/CCG5/CCG5C/CCG6/CCG3PA2/CMG1/PAG1S/PAG1P/AGC1F/CCG6DF/CCG6SF/CCG3PA_N/CCG7D or USB Serial) from the **Device type** drop-down list. The applications supported by the selected product family is displayed. Select the target application from the list.

A description of the selected application is displayed on the right pane. Click **Create Project** to start configuring the device.



Power Adapter

This node corresponds to a USB-PD compliant power adapter that can charge a peripheral through the Type-C connection.

Figure 16 New configuration dialog box

The configuration is saved as an XML file. To open a saved configuration, follow the menu path **File** > **Open**. You can modify, save, and program this configuration to the Type-C or Billboard controller.

To read the current configuration from the target device, follow the menu path **File** > **Read from Device**. All devices currently connected to the host PC are displayed in a tree view. You can select a desired target to read from. You can modify, save, and re-program the loaded configuration to the controller.

4.1 USB-PD device configuration

Device configuration parameters are classified into device parameters and port parameters. For multi-port devices there will be multiple copies of port parameter entry options. Port parameters are further classified and grouped into a hierarchical *Tree* structure. The following sets of parameters will be displayed:

- Device parameters
- Port 0
  - Discover identity
  - SVID configuration
  - PDO configuration
  - Port information
  - Billboard parameters

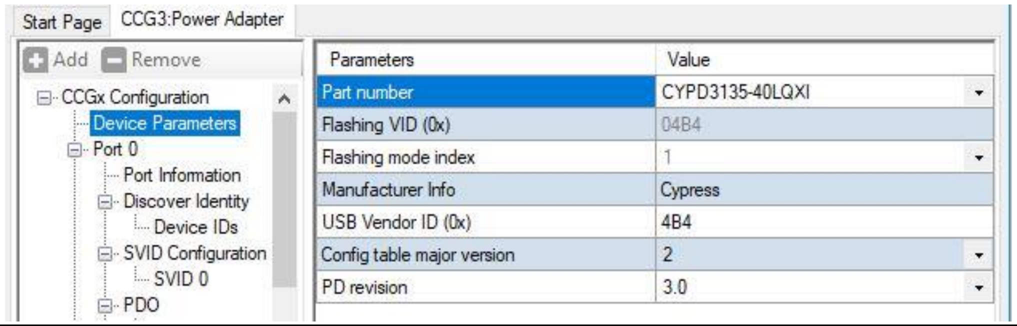
  - DP mode parameters
  - SCEDB configuration
  - SKEDB configuration
  - Power protection parameters
  - Power settings
  - Battery configuration
  - Charging configuration
  - BCR configuration
  - Type-A configuration
  - Auto configuration
  - Thunderbolt host configuration
  - Base alternate modes configuration
  - Custom alternate mode configuration
  - Custom host configuration
  - ICL/TGL configuration
- User parameters

Some of these sections only apply to specific device types and are visible only when working with such devices. Each of the above parameter sets are described in subsequent sections.

For description of the Notebook application parameters, see Section 3.2 of the **EZ-PD™ CCGx Host SDK User Guide** and for Power based applications (power bank, power adapter) parameters, see Section 3.2 **EZ-PD™ CCGx Power SDK User Guide**.

4.1.1 Device parameters

When creating a new configuration, the first step is to select the EZ-PD™ part that will be used for the design. In addition to the part selection, the **Device Parameters** node also contains EZ-PD™ controller specific parameters that determine how the Type-C controller works.



Parameters	Value
Part number	CYPD3135-40LQXI
Flashing VID (0x)	04B4
Flashing mode index	1
Manufacturer Info	Cypress
USB Vendor ID (0x)	4B4
Config table major version	2
PD revision	3.0

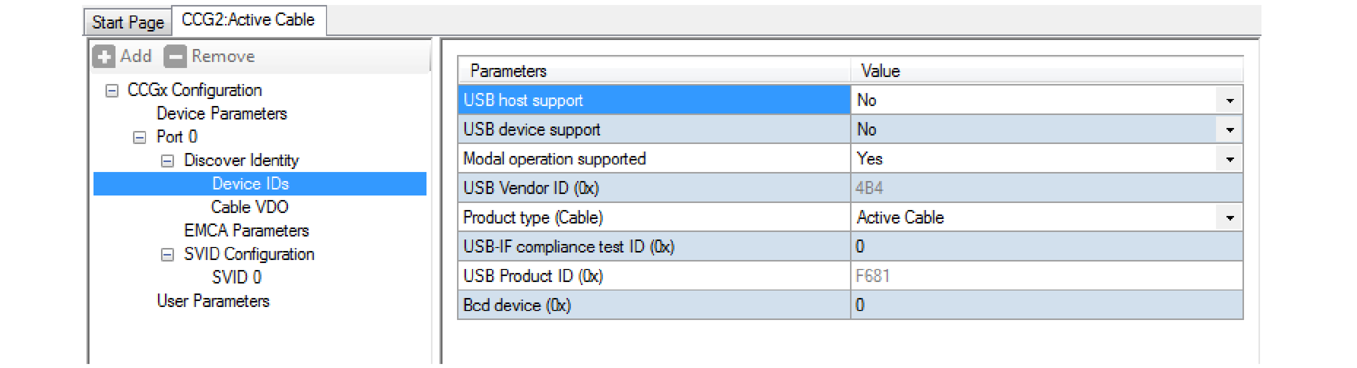
Figure 17 **CCGx device parameters configuration**

Note: *The Flashing VID and Flashing mode index parameters are only required by EMCA and Power Adapter solutions where the device programming is done through the CC interface.*

4.1.2 Discover identity

The Discover Identity nodes contain the parameters that are reported by the EZ-PD™ controller in the form of the USB-PD Discover Identity response. The parameters include the responses included in the Id Header, Cert Stat, Product, and any Product Specific VDOs.

Some of these fields like Product Type are fixed for a specific application. These values are grayed out and cannot be modified.

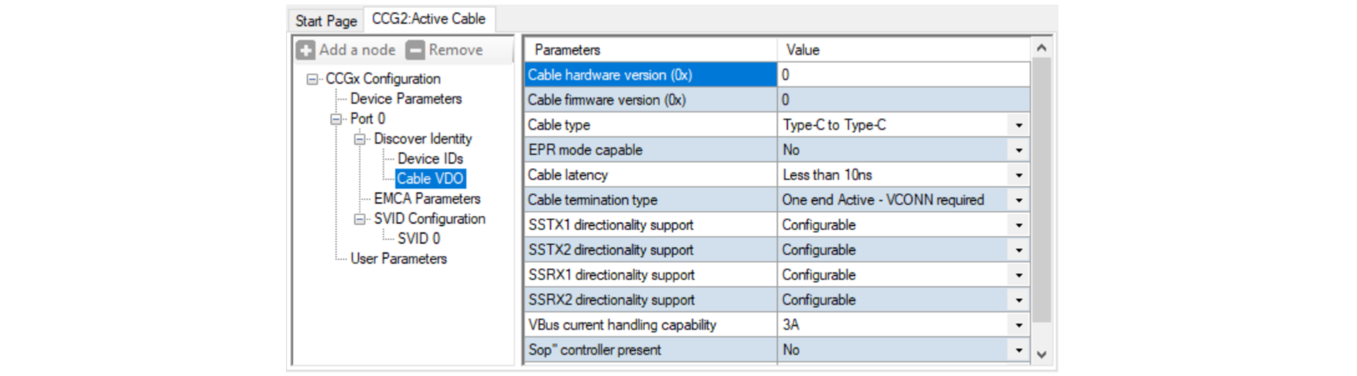


Parameters	Value
USB host support	No
USB device support	No
Modal operation supported	Yes
USB Vendor ID (0x)	4B4
Product type (Cable)	Active Cable
USB-IF compliance test ID (0x)	0
USB Product ID (0x)	F681
Bcd device (0x)	0

Figure 18 Device ID configuration

The Device IDs are VDOs that need to be configured for all USB-PD devices. These contain fields that describe the device's identity.

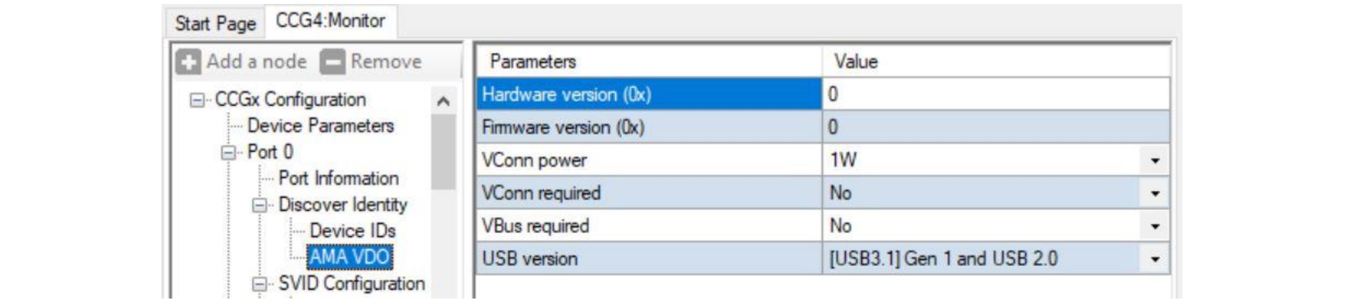
The product-specific VDOs vary with the type of USB-PD application. In case of EMCA applications, the Cable VDO needs to be configured.



Parameters	Value
Cable hardware version (0x)	0
Cable fimware version (0x)	0
Cable type	Type-C to Type-C
EPR mode capable	No
Cable latency	Less than 10ns
Cable temination type	One end Active - VCONN required
SSTX1 directionality support	Configurable
SSTX2 directionality support	Configurable
SSRX1 directionality support	Configurable
SSRX2 directionality support	Configurable
VBus cument handling capability	3A
Sop" controller present	No

Figure 19 Cable VDO configuration

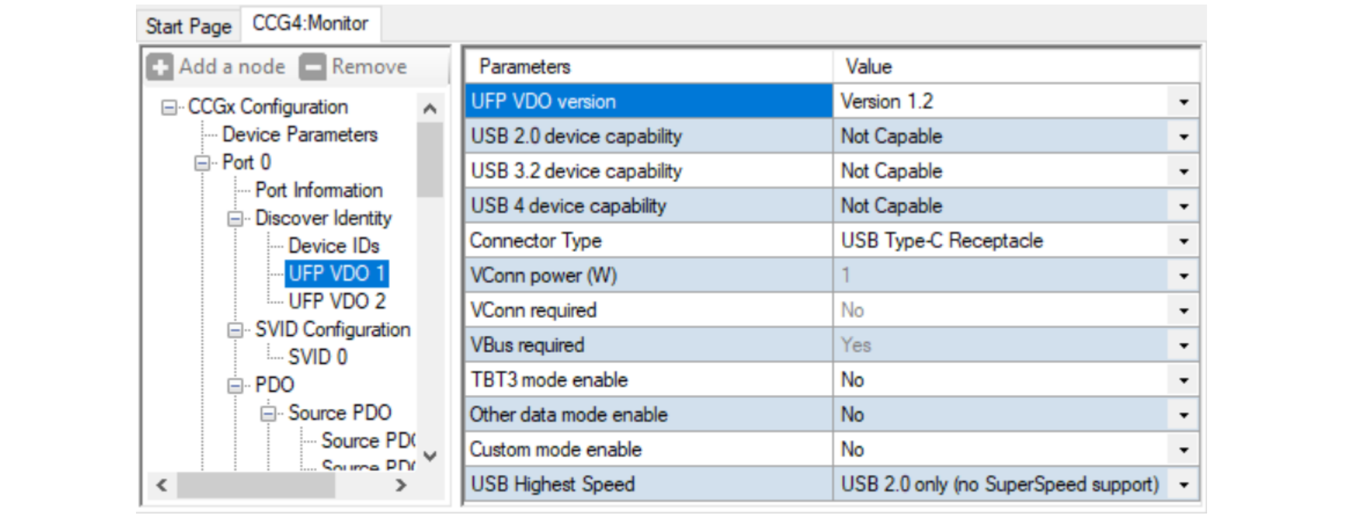
The AMA VDO needs to be configured for applications that support one or more Alternate Modes.



Parameters	Value
Hardware version (0x)	0
Fimware version (0x)	0
VConn power	1W
VConn required	No
VBus required	No
USB version	[USB3.1] Gen 1 and USB 2.0

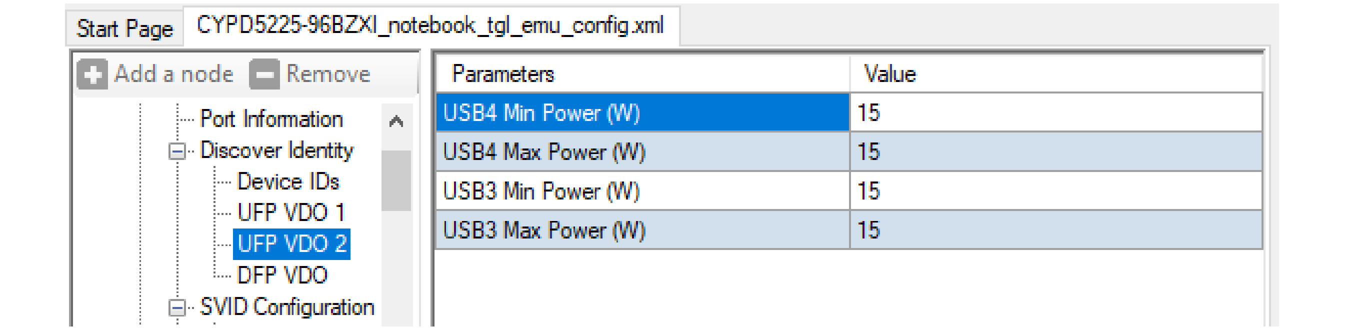
Figure 20 AMA VDO configuration

The UFP VDOs need to be configured for ports capable of operating as a UFP; including traditional USB peripherals, USB hub's upstream Port, and DRD capable host Ports.



Parameters	Value
UFP VDO version	Version 1.2
USB 2.0 device capability	Not Capable
USB 3.2 device capability	Not Capable
USB 4 device capability	Not Capable
Connector Type	USB Type-C Receptacle
VConn power (W)	1
VConn required	No
VBus required	Yes
TBT3 mode enable	No
Other data mode enable	No
Custom mode enable	No
USB Highest Speed	USB 2.0 only (no SuperSpeed support)

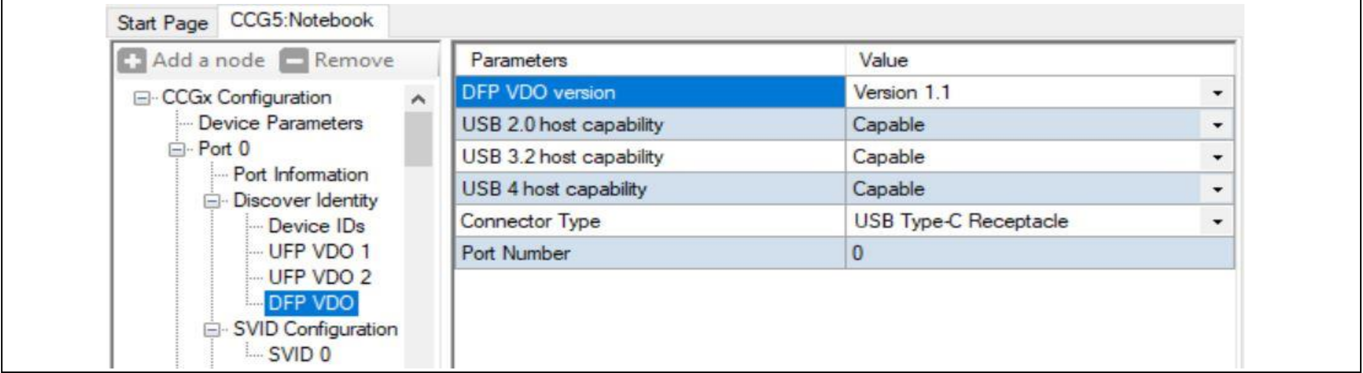
Figure 21 UFP VDO 1 configuration



Parameters	Value
USB4 Min Power (W)	15
USB4 Max Power (W)	15
USB3 Min Power (W)	15
USB3 Max Power (W)	15

Figure 22 UFP VDO 2 configuration

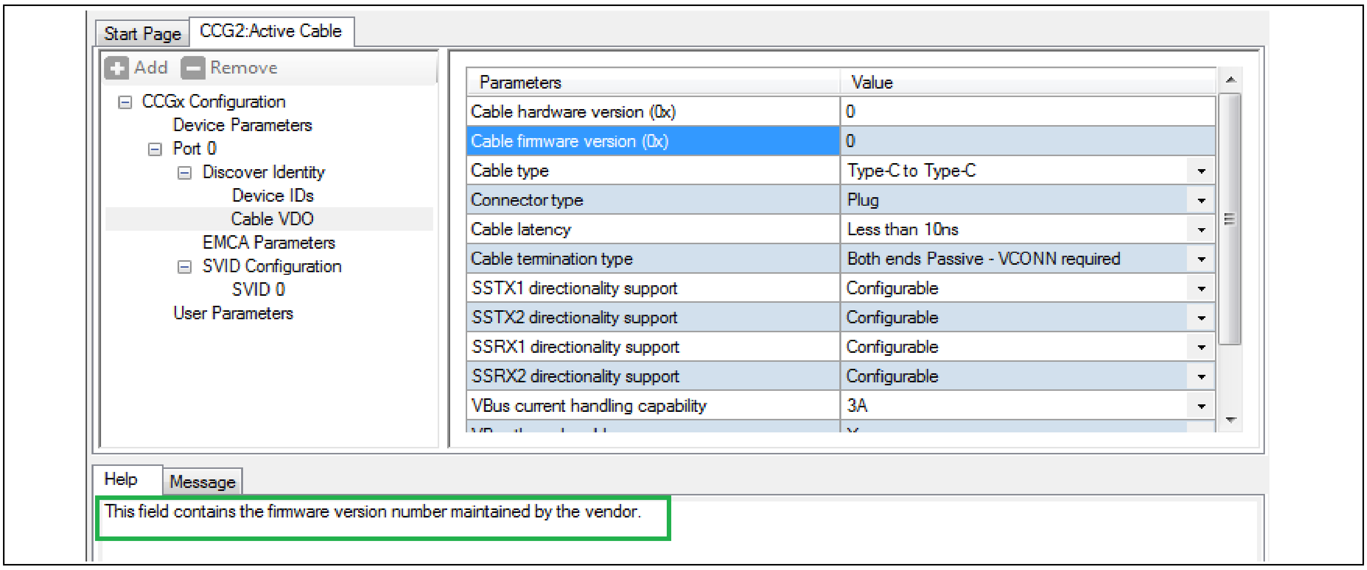
The DFP VDO needs to be configured for ports capable of operating as a DFP; including those implemented by Hosts, Hubs, and Power Bricks.



Parameters	Value
DFP VDO version	Version 1.1
USB 2.0 host capability	Capable
USB 3.2 host capability	Capable
USB 4 host capability	Capable
Connector Type	USB Type-C Receptacle
Port Number	0

Figure 23 DFP VDO configuration

To view the description of a parameter, hover the mouse over the parameter name. A tool-tip will display the description. You can also click a parameter name and view the description in the **Help** tab at the bottom of the dialog.



Parameters	Value
Cable hardware version (0x)	0
Cable firmware version (0x)	0
Cable type	Type-C to Type-C
Connector type	Plug
Cable latency	Less than 10ns
Cable termination type	Both ends Passive - VCONN required
SSTX1 directionality support	Configurable
SSTX2 directionality support	Configurable
SSRX1 directionality support	Configurable
SSRX2 directionality support	Configurable
VBus current handling capability	3A

This field contains the firmware version number maintained by the vendor.

Figure 24 Tool-tip Display for parameters

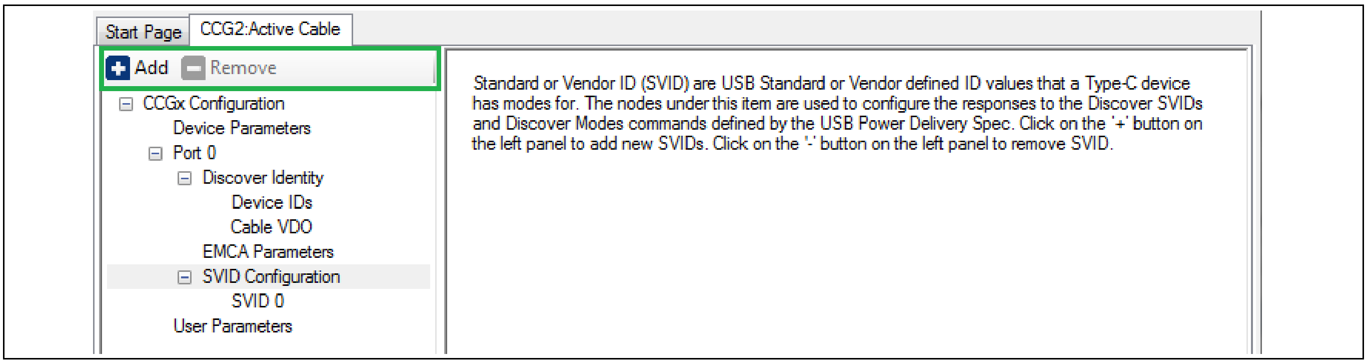
4.1.3 SVID configuration

The SVID Configuration node is used to specify the various USB-PD alternate modes supported by the Type-C controller.

For EMCA and Power Adapter devices, this node is pre-populated with SVID (0x04B4) and flashing mode (0x01). This alternate mode is used by the EZ-PD™ controllers to implement configuration update and firmware upgrades through CC VDMs. The value of the Flashing VID and Mode can be changed in the Device Parameters section. The values displayed under SVID0 will automatically be changed when flashing VID and Mode Index are changed.

Use **Add** and **Remove** on the top of the left pane to add new SVIDs and to remove existing SVIDs from the configuration. **Add** is active only when you select a SVID Configuration node. **Remove** is active only when you select one of the SVIDs. See **Figure 25**.

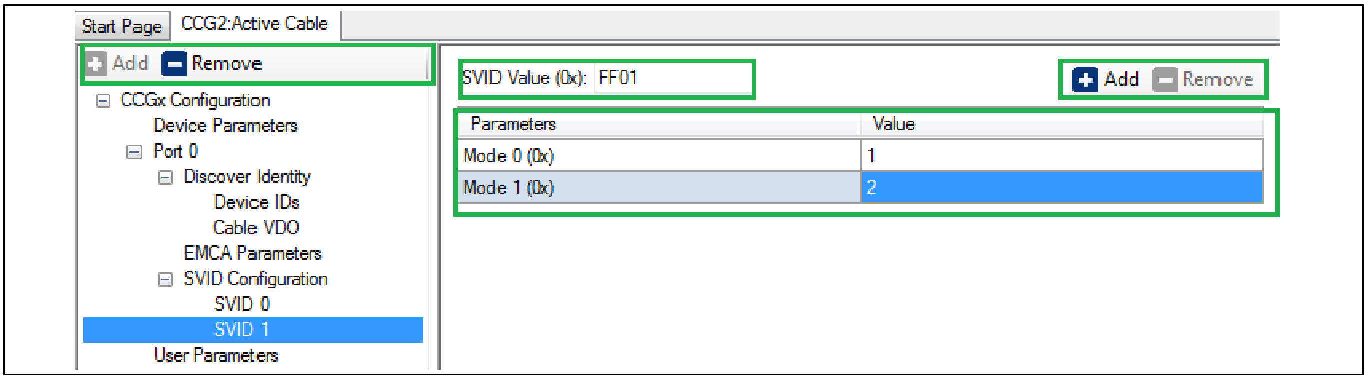
Attention: ***Devices that are programmed over the CC channel (EMCA and Power Adapters) need to support a special SVID and mode, which are used for sending the Unstructured VDMs used for the programming operation. Deleting the flashing SVID (SVID0) in these cases will cause the CCGx device to stop supporting the configuration and firmware update feature. Delete this SVID only if you want to disable the update capability in the design.***



Standard or Vendor ID (SVID) are USB Standard or Vendor defined ID values that a Type-C device has modes for. The nodes under this item are used to configure the responses to the Discover SVIDs and Discover Modes commands defined by the USB Power Delivery Spec. Click on the '+' button on the left panel to add new SVIDs. Click on the '-' button on the left panel to remove SVID.

Figure 25 Adding or removing SVIDs

After selecting an SVID note, you can edit the corresponding SVID value from the right pane. You can use **Add** and **Remove** to add and delete modes from the active SVID. **Figure 26** shows the SVID and mode values to be edited, and the options to add or remove modes to a selected SVID.



SVID Value (0x): FF01

Parameters	Value
Mode 0 (0x)	1
Mode 1 (0x)	2

Figure 26 Editing modes for an SVID

4.1.4 PDO configuration

Power Data Objects (PDOs) are data sets that describe the power capabilities of a USB-PD device. PDOs can be Source PDOs or Sink PDOs, depending on whether they describe the power sourcing or sinking capabilities of the USB-PD device.

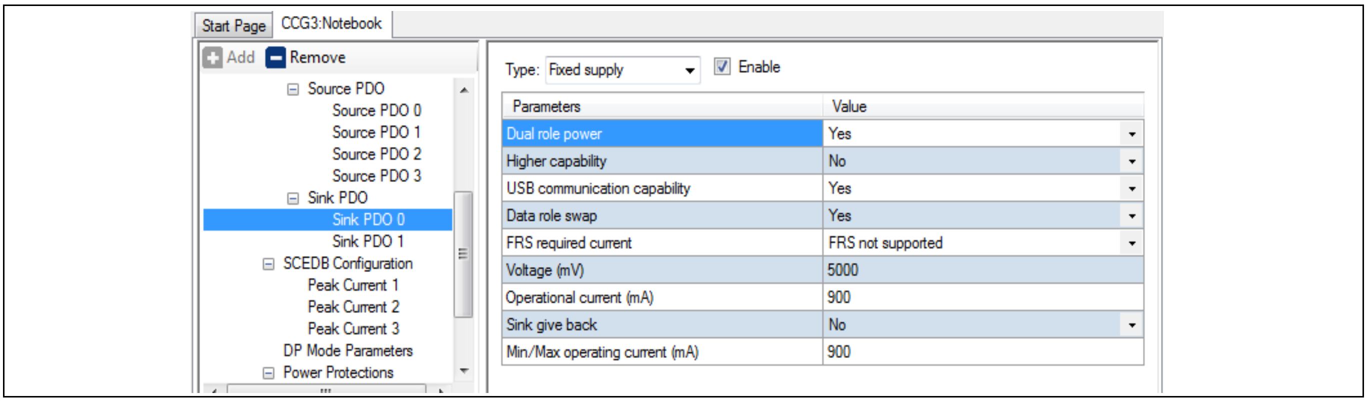
The PDO configuration section is used to define one or more PDOs describing the capabilities of the current device. This section is not applicable to EMCA devices, as EMCA does not require any PDOs.

Type-C Dongles are not expected to have any external power source and hence will support only Sink PDOs. Type-C Power Adapters will only function as power sources and will support only Source PDOs. Notebook Port Controllers will support both Source and Sink PDOs.

Figure 27 and **Figure 28** show the configuration screens used to set the properties of PDOs. All fields listed under Source and Sink PDOs are defined in the USB-PD specification. For more details, see the specification.

First, select the supply type, and then configure the remaining parameters.

An **Enable** checkbox is associated with each PDO, which specifies whether the PDO is enabled at system start-up. Solutions that support the Host Processor Interface (HPI) allow to dynamically enable/disable of PDOs at runtime. In other solutions, all available PDOs should be enabled.



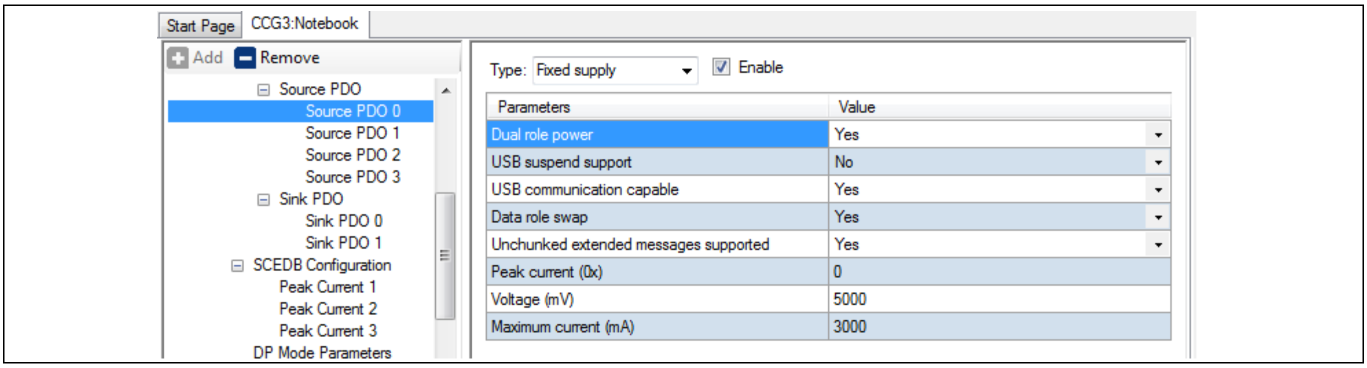
Type: Fixed supply ☑ Enable

Parameters	Value
Dual role power	Yes
Higher capability	No
USB communication capability	Yes
Data role swap	Yes
FRS required current	FRS not supported
Voltage (mV)	5000
Operational current (mA)	900
Sink give back	No
Min/Max operating current (mA)	900

Figure 27 Configuring sink PDOs

Sink PDO configuration includes two additional parameters:

- The **Sink gives back** parameter specifies whether the feature is enabled for this PDO.
- The **Min/Max operating current** parameter specifies the minimum current required for device operation if give back is enabled. If give back is disabled, this field specifies the maximum operating current that may be required.



Start Page | CCG3:Notebook

Add | Remove

- Source PDO
  - Source PDO 0
  - Source PDO 1
  - Source PDO 2
  - Source PDO 3
- Sink PDO
  - Sink PDO 0
  - Sink PDO 1
- SCEDB Configuration
  - Peak Current 1
  - Peak Current 2
  - Peak Current 3
- DP Mode Parameters

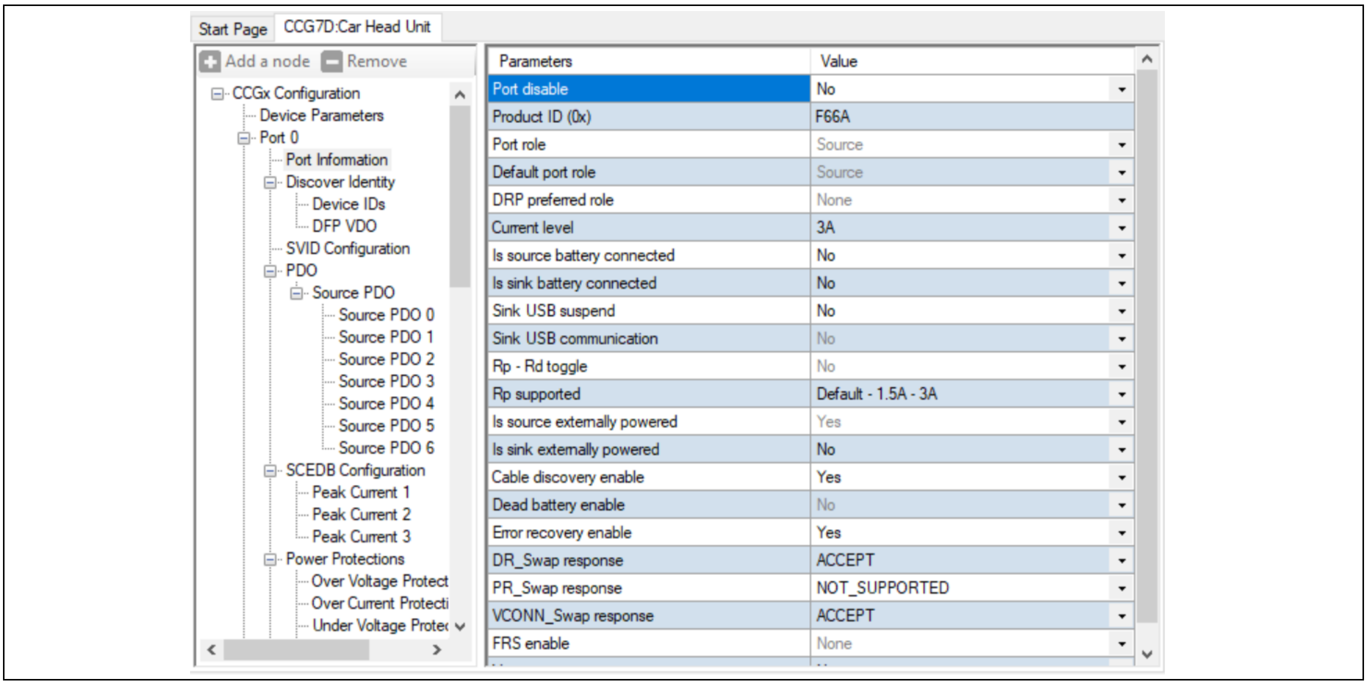
Type: Fixed supply ☑ Enable

Parameters	Value
Dual role power	Yes
USB suspend support	No
USB communication capable	Yes
Data role swap	Yes
Unchunked extended messages supported	Yes
Peak current (0x)	0
Voltage (mV)	5000
Maximum current (mA)	3000

Figure 28 Configuring source PDOs

4.1.5 Port information

The PD Port Information node contains a set of properties that control how the CCGx controller manages the USB-PD Port. Most of these properties are mapped to the properties defined by the USB-PD Specification, and the rest of the device-specific parameters. View the tooltips for the description of each parameter.



Start Page | CCG7D:Car Head Unit

Add a node | Remove

- CCGx Configuration
  - Device Parameters
  - Port 0
    - Port Information
    - Discover Identity
      - Device IDs
      - DFP VDO
    - SVID Configuration
    - PDO
      - Source PDO
        - Source PDO 0
        - Source PDO 1
        - Source PDO 2
        - Source PDO 3
        - Source PDO 4
        - Source PDO 5
        - Source PDO 6
    - SCEDB Configuration
      - Peak Current 1
      - Peak Current 2
      - Peak Current 3
    - Power Protections
      - Over Voltage Protect
      - Over Current Protecti
      - Under Voltage Protec

Parameters	Value
Port disable	No
Product ID (0x)	F66A
Port role	Source
Default port role	Source
DRP preferred role	None
Current level	3A
Is source battery connected	No
Is sink battery connected	No
Sink USB suspend	No
Sink USB communication	No
Rp - Rd toggle	No
Rp supported	Default - 1.5A - 3A
Is source externally powered	Yes
Is sink externally powered	No
Cable discovery enable	Yes
Dead battery enable	No
Error recovery enable	Yes
DR_Swap response	ACCEPT
PR_Swap response	NOT_SUPPORTED
VCONN_Swap response	ACCEPT
FRS enable	None

Figure 29 Configuring PD Port information

4.1.6 Billboard parameters

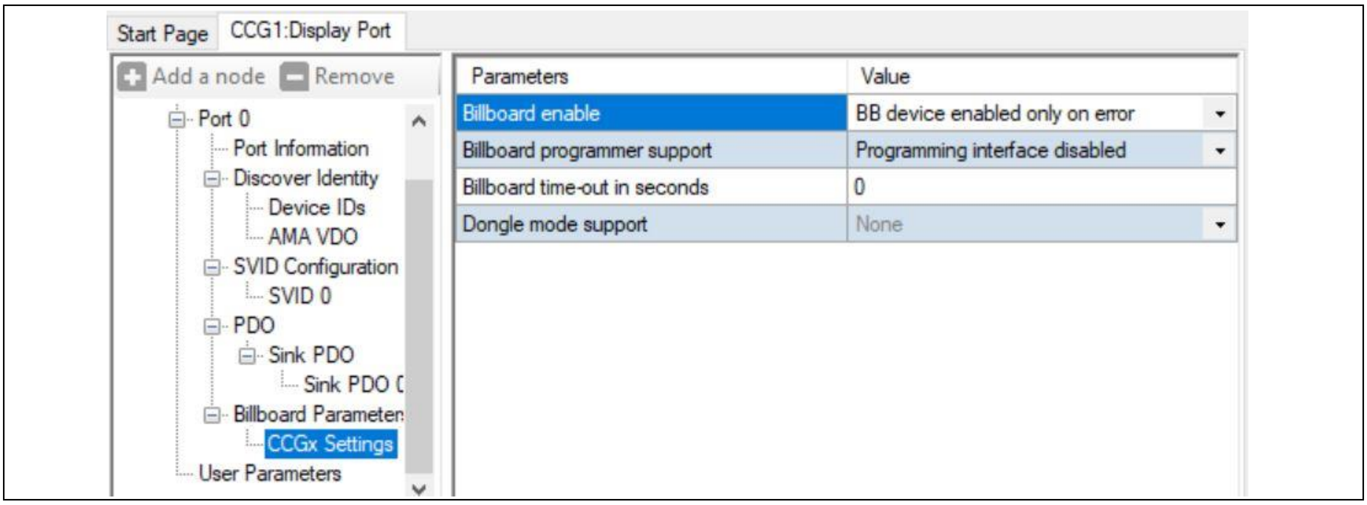
USB-PD devices such as display dongles and monitors that do not have a standard USB function need to present a USB Billboard interface to report the device capabilities and status. The Billboard interface can make use of the on-chip (internal) USB-FS block for CCG3 designs or an external USB Billboard controller for other designs.

The billboard parameters section is only applicable to USB-PD devices that have either an internal USB block or an external Billboard controller associated with them.

4.1.6.1 Designs with External Billboard controller

CCG1/CCG2 devices do not have an on-chip USB block, hence billboard functionality are provided through an external Billboard controller.

These Billboard properties for such designs control the behavior of the Billboard device and specify whether it is possible to perform configuration updates through the Billboard device. Parameters such as the USB IDs used by the Billboard Controller, the string descriptors, the supported alternate mode descriptions, and so on will be configured separately on the Billboard controller itself (see **Billboard device configuration**).



Parameters	Value
Billboard enable	BB device enabled only on error
Billboard programmer support	Programming interface disabled
Billboard time-out in seconds	0
Dongle mode support	None

Figure 30 CCG1/CCG2 Billboard control parameter configuration

- The **Billboard enable** field specifies whether the Billboard Device USB enumeration is always enabled or enabled only in case of any error on the CCGx device side.
- The **Billboard programmer support** field specifies whether the vendor mode used for CCGx flash read and update should be enabled on the Billboard device.
- The **Billboard time-out in seconds** field specifies the duration for which the Billboard device will remain enabled before being put into sleep to save power. This parameter only applies when the CCGx controller is working without any errors. If there is an error on the CCGx side, the Billboard device remains enabled continuously.

4.1.6.2 Designs with On-Chip Billboard interface

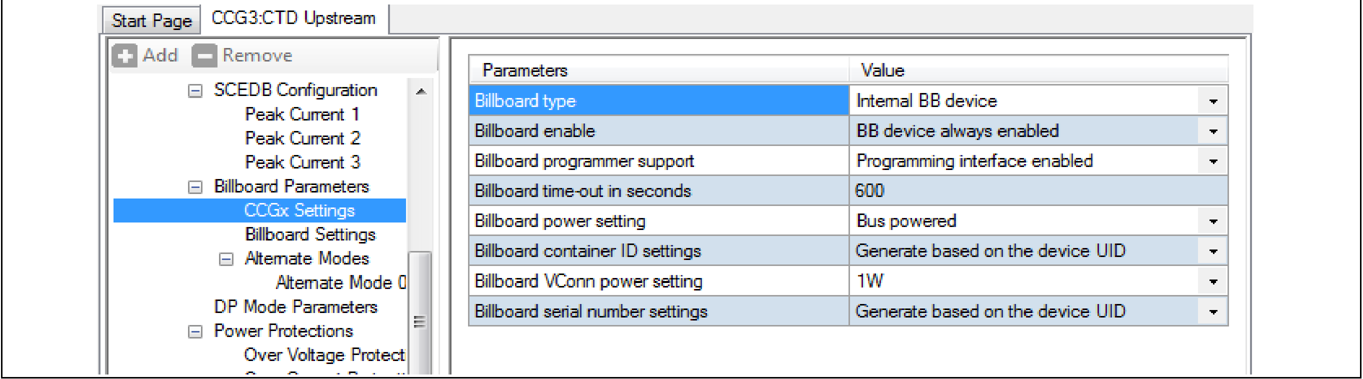
CCG3 devices have an internal USB-FS block that can be used for Billboard implementation. Since the Billboard device is implemented on CCG3 itself, all Billboard configuration is embedded in the CCG3 device's configuration table.

The following parameters need to be configured for CCG3 designs with Billboard.

CCGx settings

This node specifies the parameters that control the behavior of the Billboard interface. These parameters include:

- **Billboard type**: Specifies the Billboard type – Internal, External, or No Billboard
- **Billboard enable**: Indicates whether the Billboard functionality is always enabled or enabled only when an error occurs on CCGX device
- **Billboard programmer support**: Indicates whether flash update through Billboard device is enabled
- **Billboard timeout in seconds**: Indicates delay in seconds before the Billboard USB device is turned OFF to save power
- **Billboard power setting**: Indicates whether the Billboard is Bus powered or Self powered
- **Billboard VConn power setting**: Specifies the power that the design will draw from the VConn supply
- **Billboard container ID settings**
- **Billboard serial number settings**



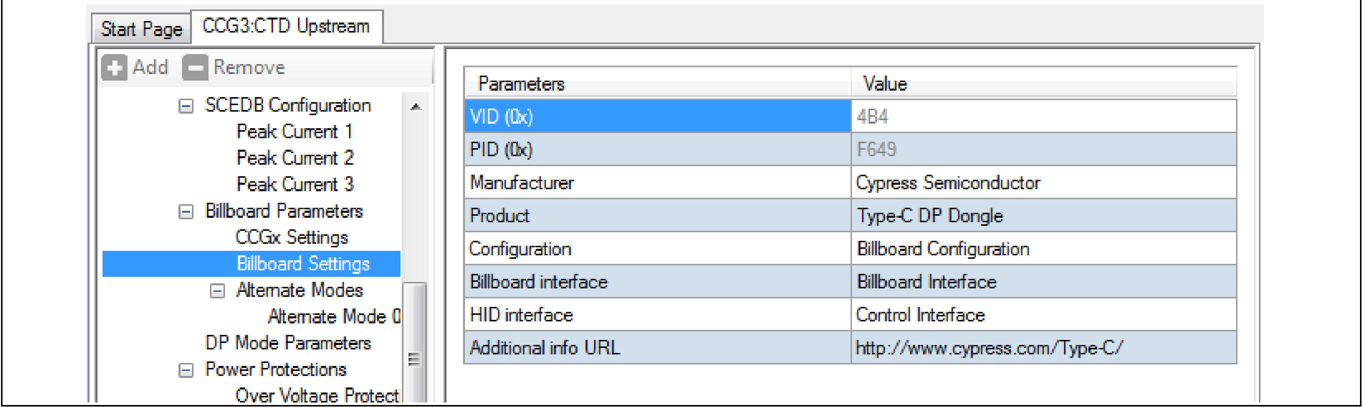
Parameters	Value
Billboard type	Internal BB device
Billboard enable	BB device always enabled
Billboard programmer support	Programming interface enabled
Billboard time-out in seconds	600
Billboard power setting	Bus powered
Billboard container ID settings	Generate based on the device UID
Billboard VConn power setting	1W
Billboard serial number settings	Generate based on the device UID

Figure 31 CCG3 Billboard CCGx parameters configurations

Billboard settings

This node specifies parameters associated with Billboard USB device. These parameters include:

- VID and PID used to identify the device.
- Manufacturer, Product, Serial Number, Configuration, and Interface strings used to identify the device.
- URL that can provide more information about the device.



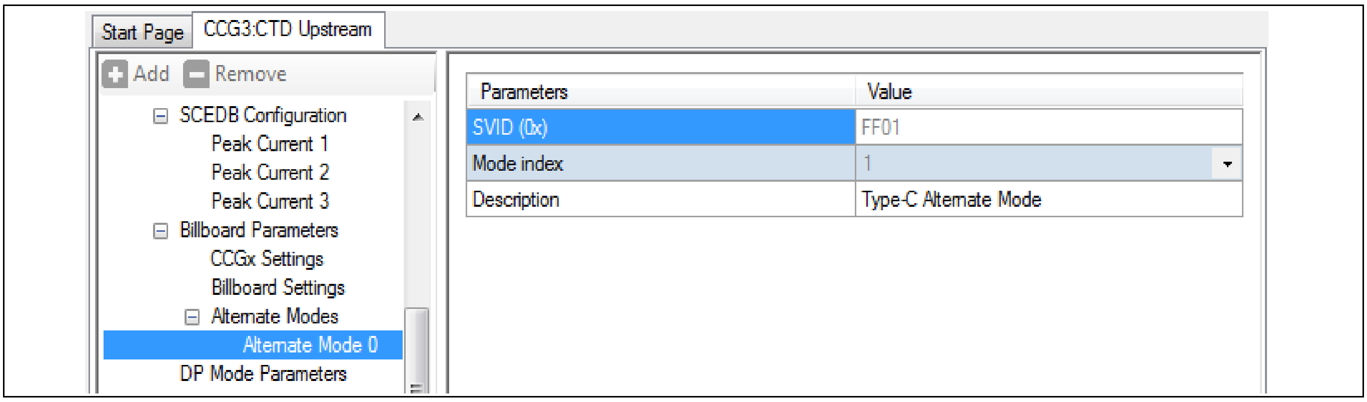
Parameters	Value
VID (0x)	4B4
PID (0x)	F649
Manufacturer	Cypress Semiconductor
Product	Type-C DP Dongle
Configuration	Billboard Configuration
Billboard interface	Billboard Interface
HID interface	Control Interface
Additional info URL	http://www.cypress.com/Type-C/

Figure 32 CCG3 Billboard parameter configuration

Alternate modes

Alternate modes configuration parameters specify the alternate modes supported by USB-PD controller. These parameters include Standard or Vendor ID (SVID), mode index, and description corresponding to each supported alternate mode.

All alternate modes defined under the SVID section are listed under this node and you can edit its description in this section. To add or remove an alternate mode, you need to add or remove modes under the SVID section.

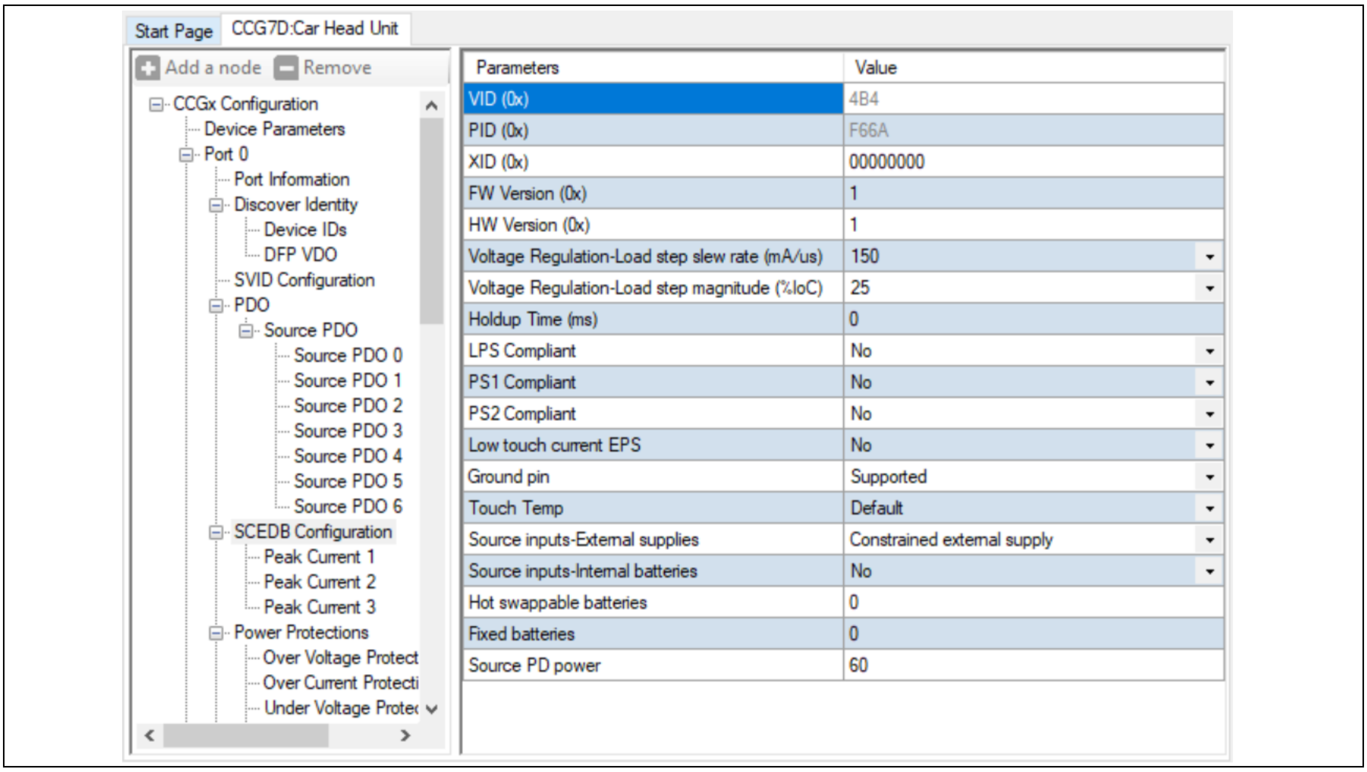


Parameters	Value
SVID (0x)	FF01
Mode index	1
Description	Type-C Alternate Mode

Figure 33 CCG3 Billboard alternate modes parameter configuration

4.1.7 SCEDB configuration

Source Capabilities Extended Data Block (SCEDB) contains additional information about a Port's Source Capabilities. **Figure 34** shows the SCEDB configurations.

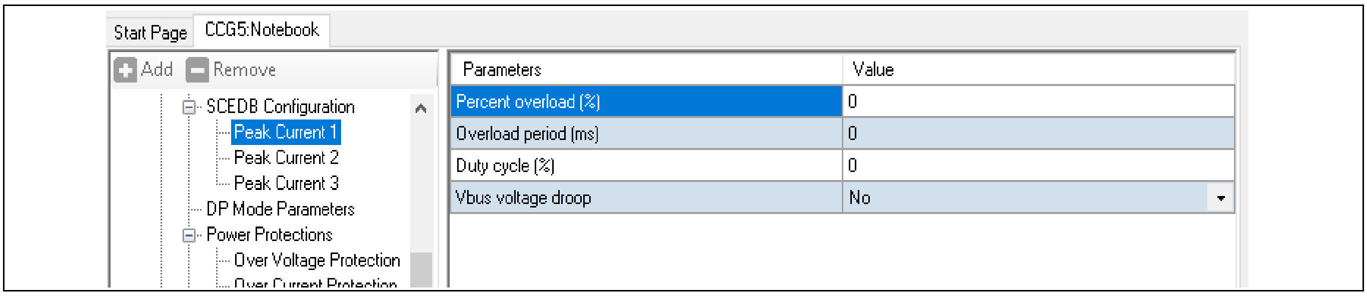


Parameters	Value
VID (0x)	4B4
PID (0x)	F66A
XID (0x)	00000000
FW Version (0x)	1
HW Version (0x)	1
Voltage Regulation-Load step slew rate (mA/us)	150
Voltage Regulation-Load step magnitude (%IoC)	25
Holdup Time (ms)	0
LPS Compliant	No
PS1 Compliant	No
PS2 Compliant	No
Low touch current EPS	No
Ground pin	Supported
Touch Temp	Default
Source inputs-External supplies	Constrained external supply
Source inputs-Internal batteries	No
Hot swappable batteries	0
Fixed batteries	0
Source PD power	60

Figure 34 SCEDB configuration

4.1.7.1 Peak current configuration

The **Peak Current** field contains the combination of Peak Currents that the Source supports. Peak Current provides a means for the Source to report its ability to provide current in excess of the negotiated amount for short periods. The Peak Current descriptor defines up to three combinations of percent overload, duration, and duty cycle defined as PeakCurrent1, PeakCurrent2, and PeakCurrent3 that the Source supports. In some cases, a Source might not offer Peak Current capability.

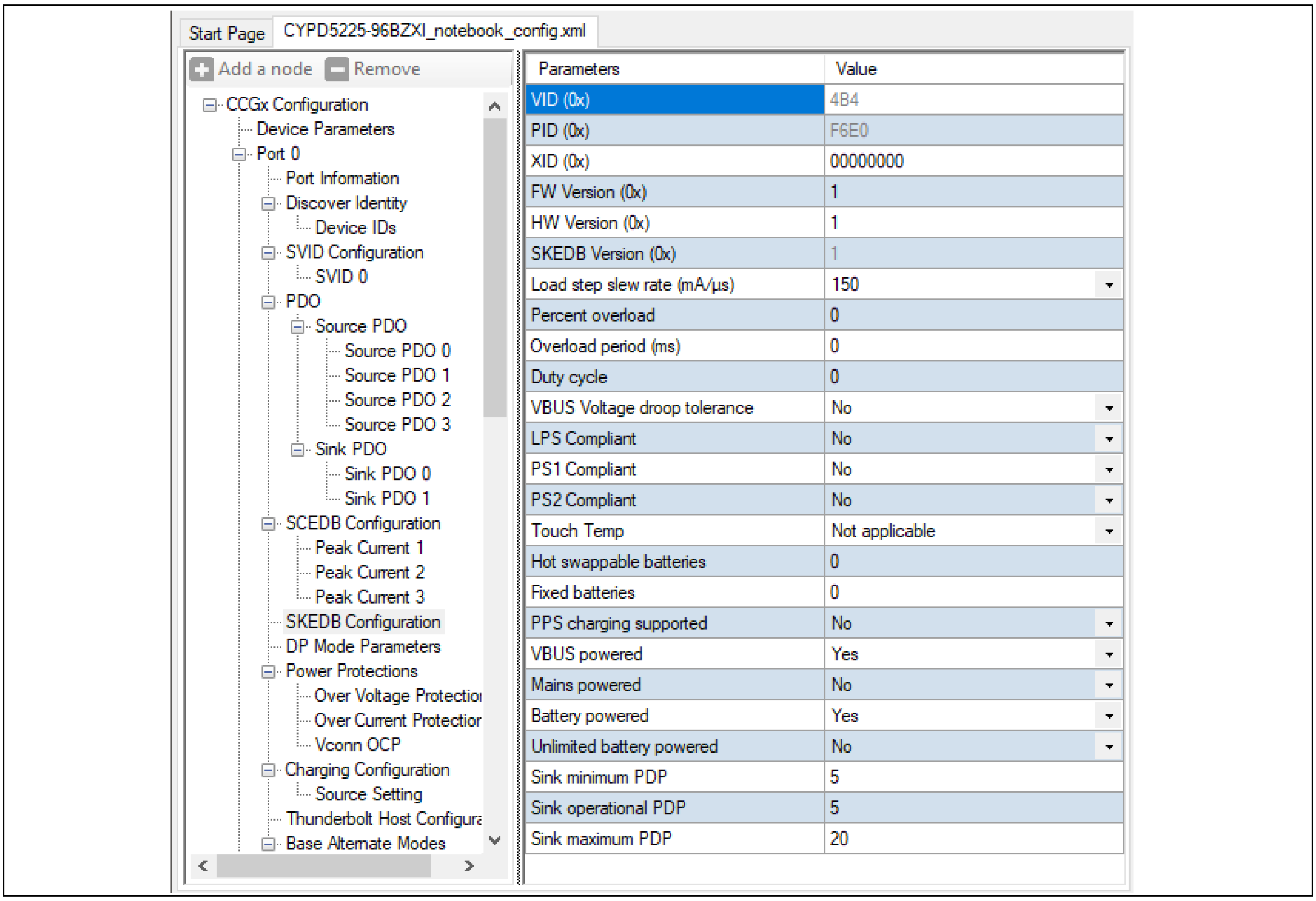


Parameters	Value
Percent overload (%)	0
Overload period (ms)	0
Duty cycle (%)	0
Vbus voltage droop	No

Figure 35 SCEDB peak current configuration

4.1.8 SKEDB configuration

Sink Capabilities Extended Data Block (SKEDB) contains additional information about a Port’s Sink Capabilities. **Figure 36** shows the SKEDB configurations.



Parameters	Value
VID (0x)	4B4
PID (0x)	F6E0
XID (0x)	00000000
FW Version (0x)	1
HW Version (0x)	1
SKEDB Version (0x)	1
Load step slew rate (mA/µs)	150
Percent overload	0
Overload period (ms)	0
Duty cycle	0
VBUS Voltage droop tolerance	No
LPS Compliant	No
PS1 Compliant	No
PS2 Compliant	No
Touch Temp	Not applicable
Hot swappable batteries	0
Fixed batteries	0
PPS charging supported	No
VBUS powered	Yes
Mains powered	No
Battery powered	Yes
Unlimited battery powered	No
Sink minimum PDP	5
Sink operational PDP	5
Sink maximum PDP	20

Figure 36 SKEDB configuration

4.1.9 Protection parameters

USB-PD designs such as Notebook port controllers, power adapters, and dock/monitors need to implement protection circuitry that ensures that the device is not harmed due to over voltage, over current, over temperature, under voltage, or short circuits.

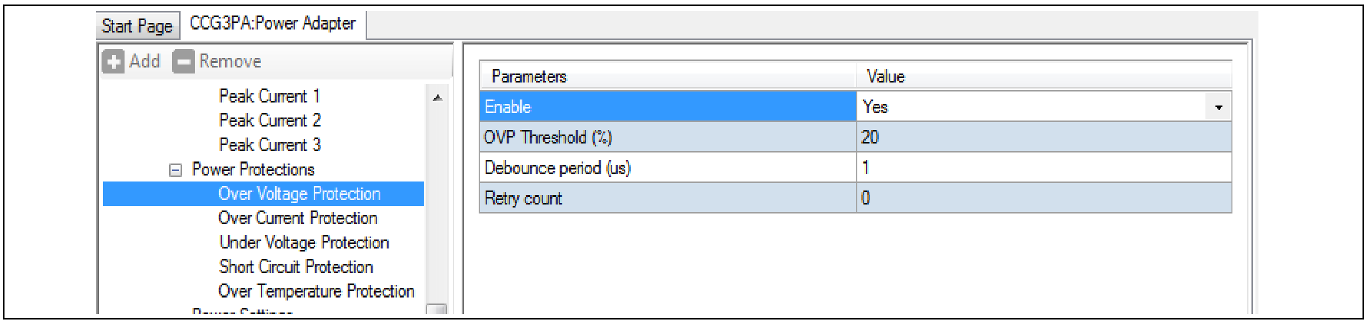
CCGx based designs can implement VBUS Over Voltage (OVP), VBUS Over Current (OCP), VBUS Under Voltage (UVP), VCONN Over Current (VCONN-OCP), Over Temperature (OTP), and Short Circuit (SCP) protection schemes.

The EZ-PD™ configuration utility allows to enable/disable these protection schemes, and configure the parameters associated with the schemes to match the target system behavior. The Protection parameters node in the configuration includes sub nodes for all protection mechanisms implemented by the corresponding firmware solution. Each protection scheme node contains the corresponding configuration parameters.

Note that these protection schemes are not implemented in all firmware solutions.

4.1.9.1 VBUS over voltage protection

This node contains the parameters that configure VBUS OVP.

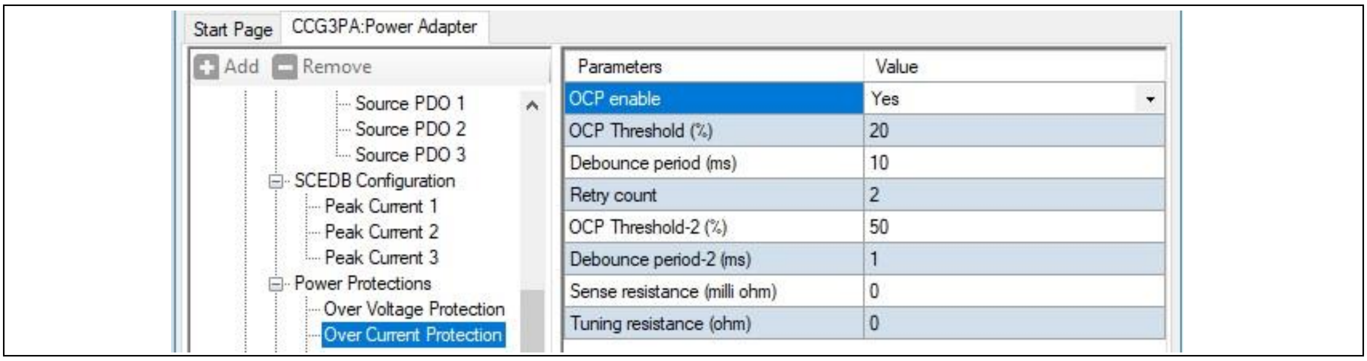


Parameters	Value
Enable	Yes
OVP Threshold (%)	20
Debounce period (us)	1
Retry count	0

Figure 37 VBUS OVP parameter configuration

4.1.9.2 VBUS over current protection

This node contains the parameters that configure VBUS OCP. The configuration parameters **Off time** and **Sample period** are displayed only when value of the **Config table major version** parameter is set to **1** in the Device Parameter configuration node.

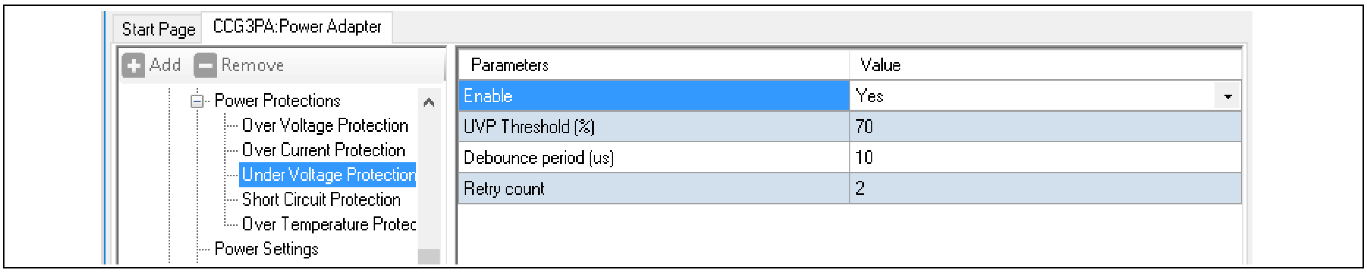


Parameters	Value
OCP enable	Yes
OCP Threshold (%)	20
Debounce period (ms)	10
Retry count	2
OCP Threshold-2 (%)	50
Debounce period-2 (ms)	1
Sense resistance (milli ohm)	0
Tuning resistance (ohm)	0

Figure 38 VBUS OCP parameter configuration

4.1.9.3 VBUS under voltage protection

This node contains the parameters that configure VBUS UVP.

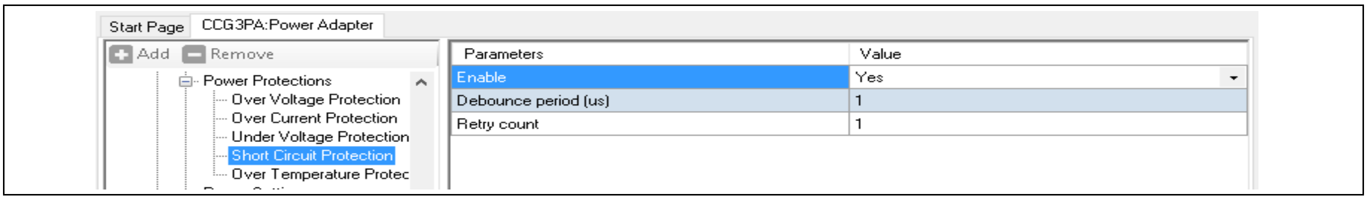


Parameters	Value
Enable	Yes
UVP Threshold (%)	70
Debounce period (us)	10
Retry count	2

Figure 39 VBUS UVP parameter configuration

4.1.9.4 Short circuit protection

This node contains the parameters that configure SCP.



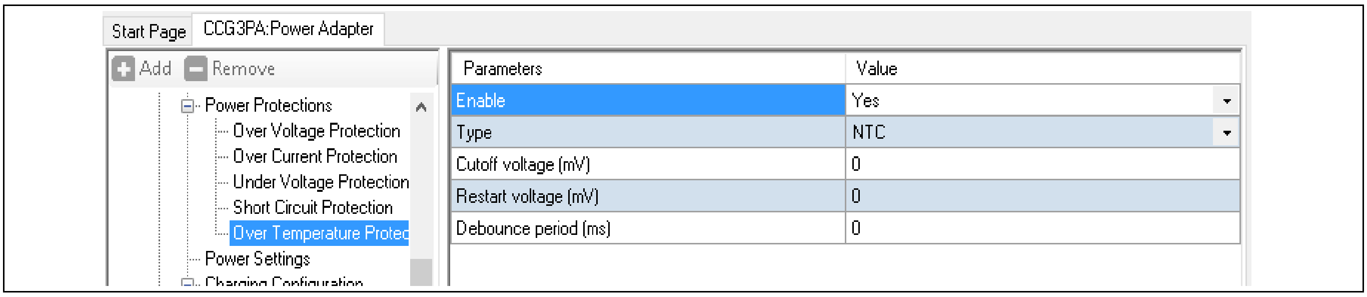
Parameters	Value
Enable	Yes
Debounce period (us)	1
Retry count	1

Figure 40 Short circuit protection parameter configuration

Note: Enabling SCP may cause false triggering of SCP on attach/detach when using Type-C to micro-B adapter.

4.1.9.5 Over temperature protection

This node contains the parameters that configure OTP.

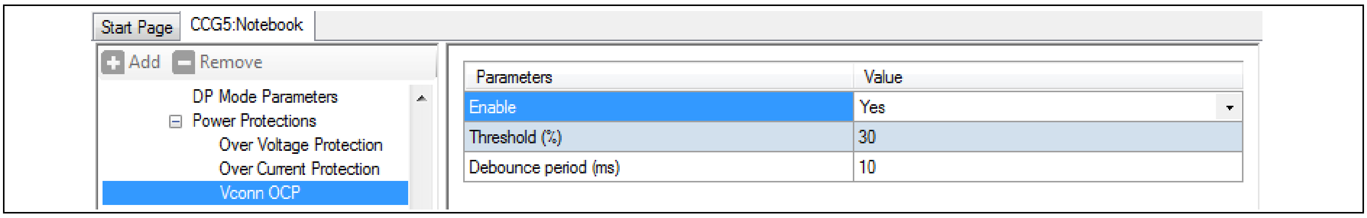


Parameters	Value
Enable	Yes
Type	NTC
Cutoff voltage (mV)	0
Restart voltage (mV)	0
Debounce period (ms)	0

Figure 41 OTP parameter configuration

4.1.9.6 VCONN over current protection

This node contains the parameters that configure the VCONN over current protection.

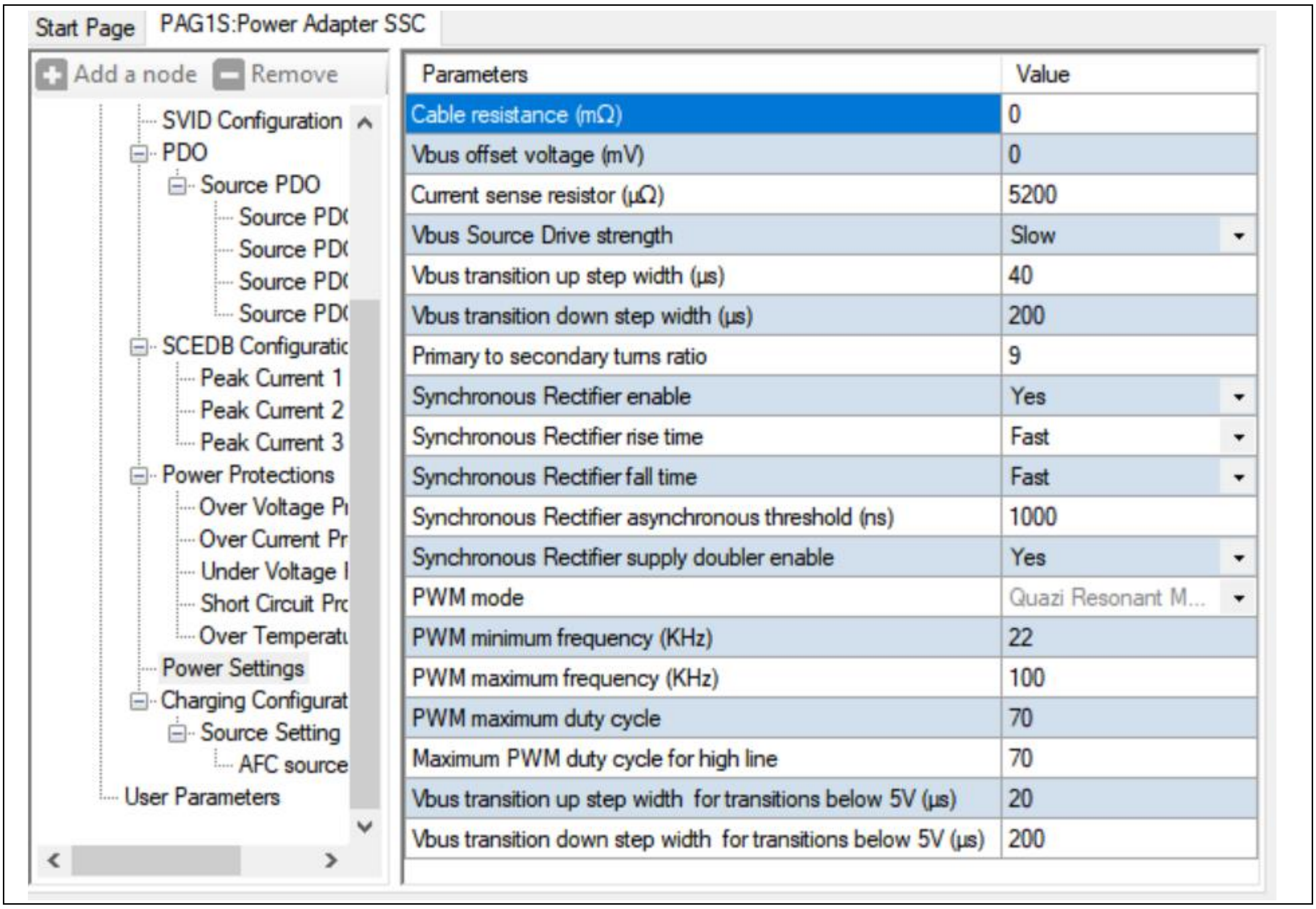


Parameters	Value
Enable	Yes
Threshold (%)	30
Debounce period (ms)	10

Figure 42 VOCNN over current protection parameter configuration

4.1.10 Power settings

Power settings node provides VBUS control configurations.



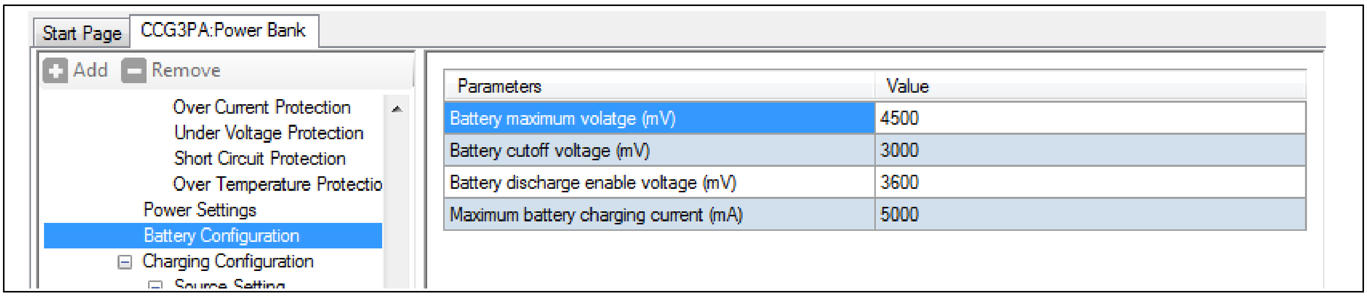
Parameters	Value
Cable resistance (mΩ)	0
Vbus offset voltage (mV)	0
Current sense resistor (µΩ)	5200
Vbus Source Drive strength	Slow
Vbus transition up step width (µs)	40
Vbus transition down step width (µs)	200
Primary to secondary turns ratio	9
Synchronous Rectifier enable	Yes
Synchronous Rectifier rise time	Fast
Synchronous Rectifier fall time	Fast
Synchronous Rectifier asynchronous threshold (ns)	1000
Synchronous Rectifier supply doubler enable	Yes
PWM mode	Quazi Resonant M...
PWM minimum frequency (KHz)	22
PWM maximum frequency (KHz)	100
PWM maximum duty cycle	70
Maximum PWM duty cycle for high line	70
Vbus transition up step width for transitions below 5V (µs)	20
Vbus transition down step width for transitions below 5V (µs)	200

Figure 43 Power settings parameter configuration

4.1.11 Battery configuration

Battery configuration provides following battery and charging parameters configurations:

- **Battery maximum voltage (mV)**: Maximum battery voltage beyond which it is unsafe to charge.
- **Battery cutoff voltage (mV)**: Battery voltage at which the device is expected to stop discharging and be on low voltage consumption
- **Battery discharge enable voltage (mV)**: Battery voltage at which discharge can be re-enabled.
- **Maximum battery charging current (mA)**: Maximum charging current allowed for the battery.



Parameters	Value
Battery maximum volatge (mV)	4500
Battery cutoff voltage (mV)	3000
Battery discharge enable voltage (mV)	3600
Maximum battery charging current (mA)	5000

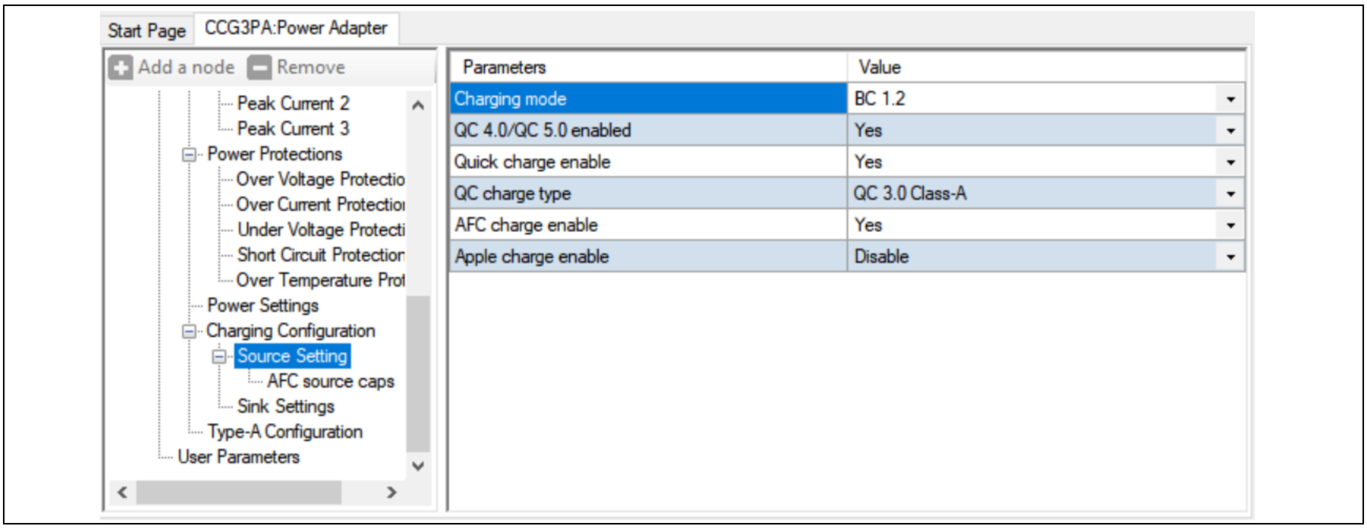
Figure 44 Battery configuration

4.1.12 Charging configuration

Charging configuration contains configurations of legacy charging support as source and sink.

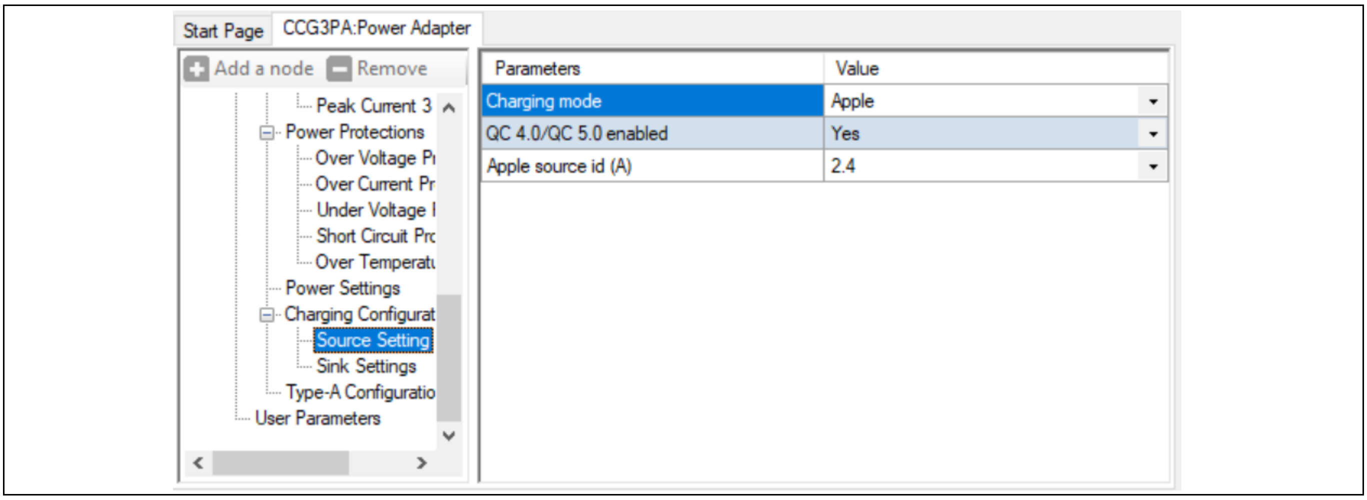
4.1.12.1 Source setting

Figure 45 and **Figure 46** show the legacy charging source setting configuration parameters for BC 1.2 and Apple charging modes, respectively.



Parameters	Value
Charging mode	BC 1.2
QC 4.0/QC 5.0 enabled	Yes
Quick charge enable	Yes
QC charge type	QC 3.0 Class-A
AFC charge enable	Yes
Apple charge enable	Disable

Figure 45 Legacy charging source setting parameter configuration for BC1.2 charging mode

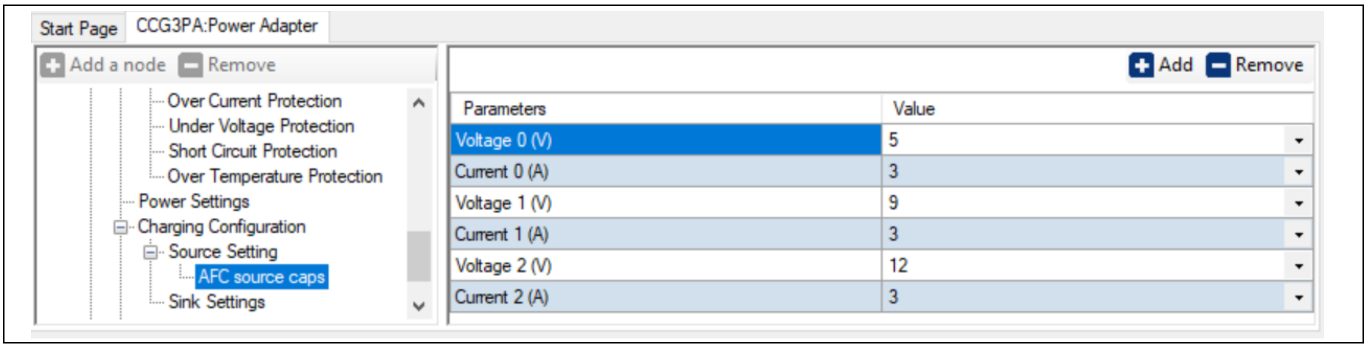


Parameters	Value
Charging mode	Apple
QC 4.0/QC 5.0 enabled	Yes
Apple source id (A)	2.4

Figure 46 Legacy charging source setting parameter configuration for Apple charging mode

AFC source caps

This node contains a list of Adaptive Fast Charging Source capability objects. This node is visible only when AFC charge has been enabled in the source setting configuration node. You can add or remove AFC Source capability objects using the **Add** and **Remove** buttons provided on the upper right corner.

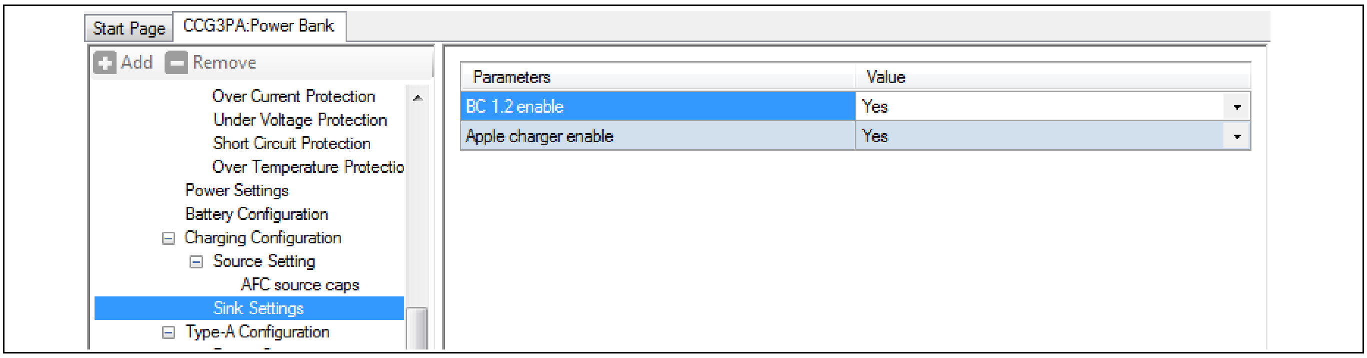


Parameters	Value
Voltage 0 (V)	5
Current 0 (A)	3
Voltage 1 (V)	9
Current 1 (A)	3
Voltage 2 (V)	12
Current 2 (A)	3

Figure 47 AFC source capability objects

Sink settings

This node is valid only for CCG3PA Power Bank application, and contains legacy charging sink setting configuration parameters.

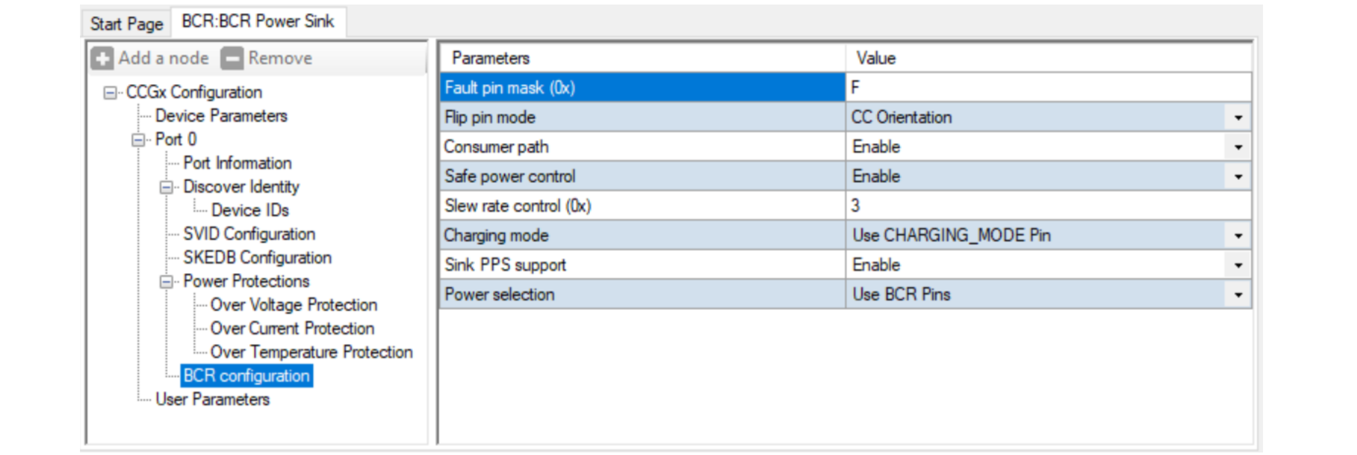


Parameters	Value
BC 1.2 enable	Yes
Apple charger enable	Yes

Figure 48 Legacy charging source setting parameter configuration

4.1.13 BCR configuration

This node is applicable only for CCG3PA BCR Power Sink application, and contains BCR configuration parameters.

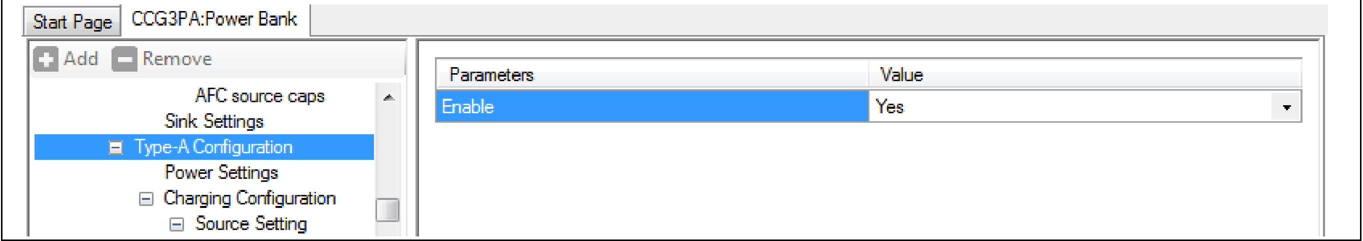


Parameters	Value
Fault pin mask (0x)	F
Flip pin mode	CC Orientation
Consumer path	Enable
Safe power control	Enable
Slew rate control (0x)	3
Charging mode	Use CHARGING_MODE Pin
Sink PPS support	Enable
Power selection	Use BCR Pins

Figure 49 BCR configuration parameters

4.1.14 Type-A configuration

In addition to the Type-C port, the device can optionally support a Type-A port. This node provides the configuration information for the Type-A port. This is applicable only for CCG3PA applications.



Parameters	Value
Enable	Yes

Figure 50 Enable Type-A configuration

4.1.14.1 Power setting

This node provides VBUS control additional configurations for Type-A port. These configuration parameters are the same as the Type-C power setting parameters

4.1.14.2 Charging configuration

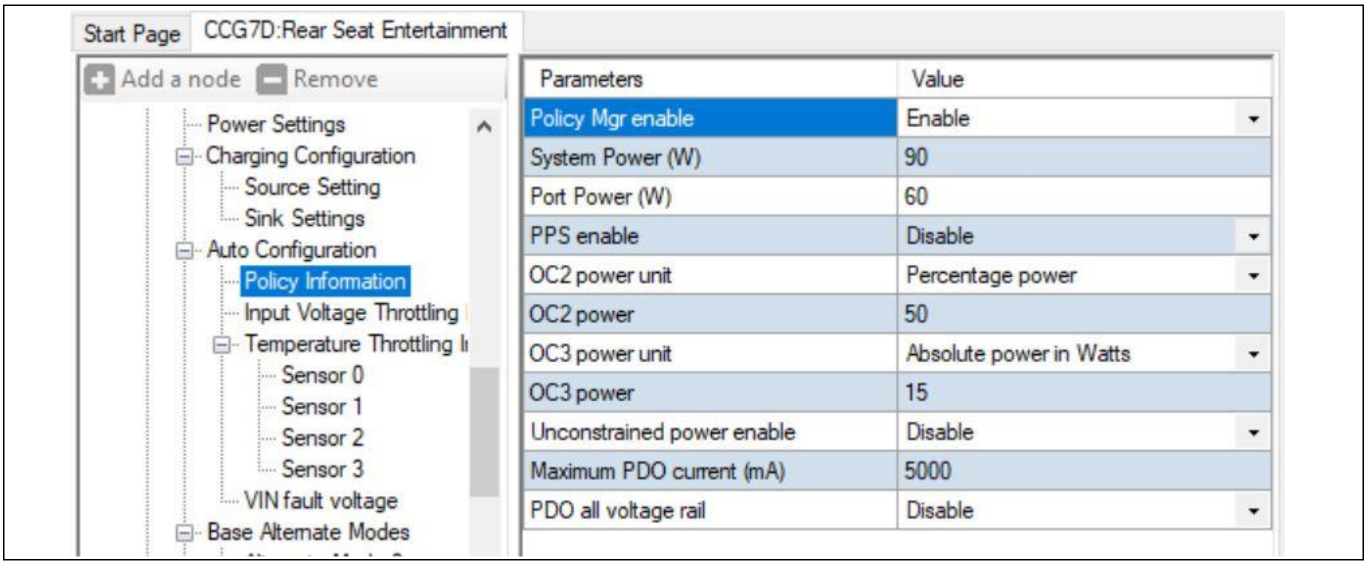
This node contains configurations of legacy charging support as Source for Type-A port. These configuration parameters are the same as the Type-C power charging configuration Source setting parameters.

4.1.15 Auto configuration

This node provides configurations for Automotive Power applications.

4.1.15.1 Policy information

This node contains policy manager information.

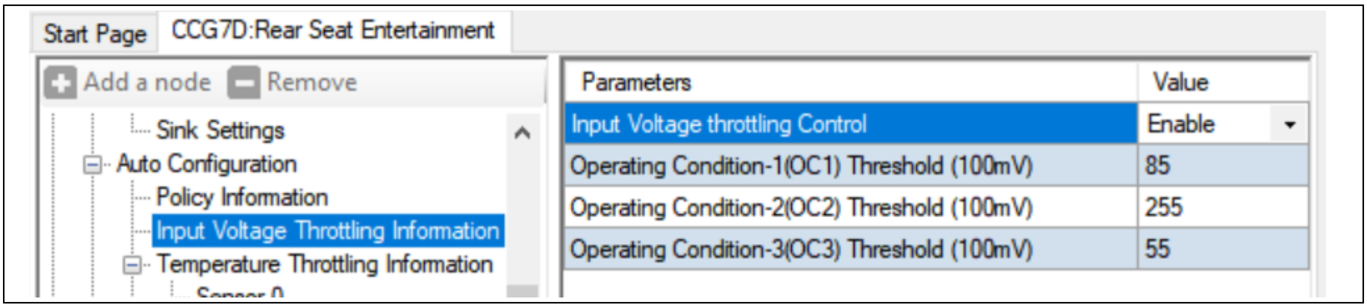


Parameters	Value
Policy Mgr enable	Enable
System Power (W)	90
Port Power (W)	60
PPS enable	Disable
OC2 power unit	Percentage power
OC2 power	50
OC3 power unit	Absolute power in Watts
OC3 power	15
Unconstrained power enable	Disable
Maximum PDO current (mA)	5000
PDO all voltage rail	Disable

Figure 51 Policy information

4.1.15.2 Input voltage throttling information

This node contains input voltage throttling information.



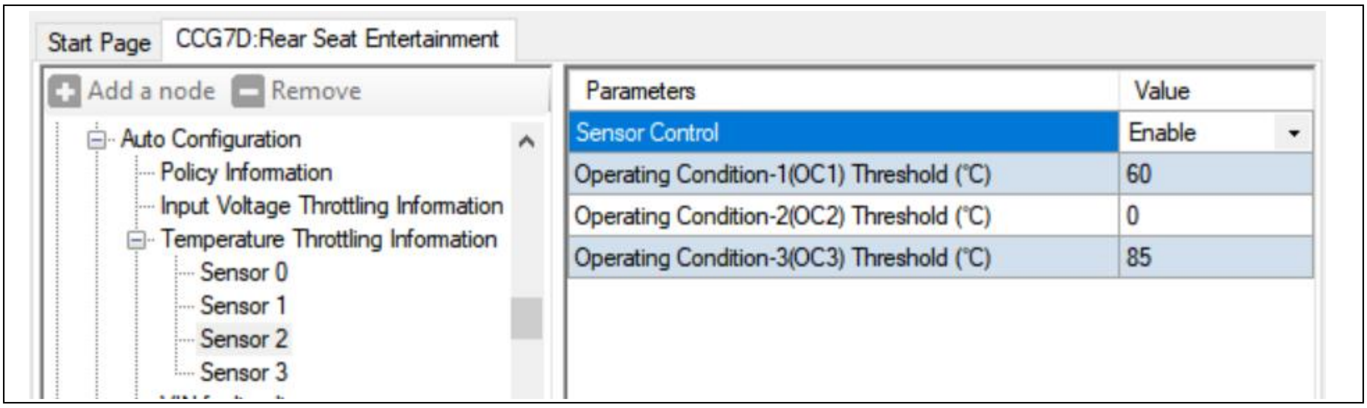
Start Page
CCG7D:Rear Seat Entertainment
Add a node
Remove
Sink Settings
Auto Configuration
Policy Information
Input Voltage Throttling Information
Temperature Throttling Information
Sensor 0

Parameters	Value
Input Voltage throttling Control	Enable
Operating Condition-1(OC1) Threshold (100mV)	85
Operating Condition-2(OC2) Threshold (100mV)	255
Operating Condition-3(OC3) Threshold (100mV)	55

Figure 52 Input voltage throttling information

4.1.15.3 Temperature throttling information

This node contains temperature throttling information. It contains sub-nodes for sensor information.

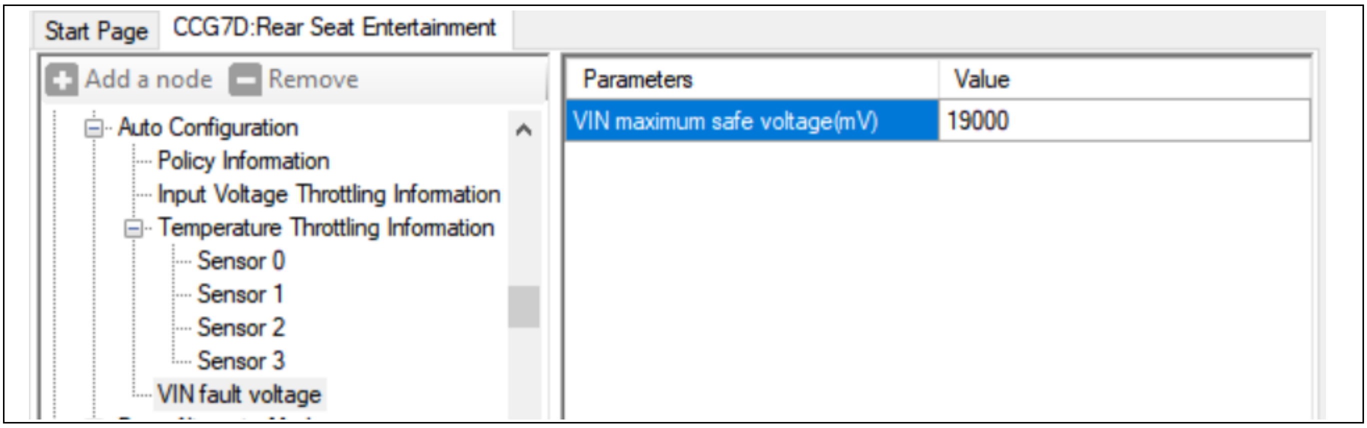

Start Page
CCG7D:Rear Seat Entertainment
Add a node
Remove
Auto Configuration
Policy Information
Input Voltage Throttling Information
Temperature Throttling Information
Sensor 0
Sensor 1
Sensor 2
Sensor 3

Parameters	Value
Sensor Control	Enable
Operating Condition-1(OC1) Threshold (°C)	60
Operating Condition-2(OC2) Threshold (°C)	0
Operating Condition-3(OC3) Threshold (°C)	85

Figure 53 Temperature throttling sensor information

4.1.15.4 VIN fault voltage

This node contains VIN fault voltage information.

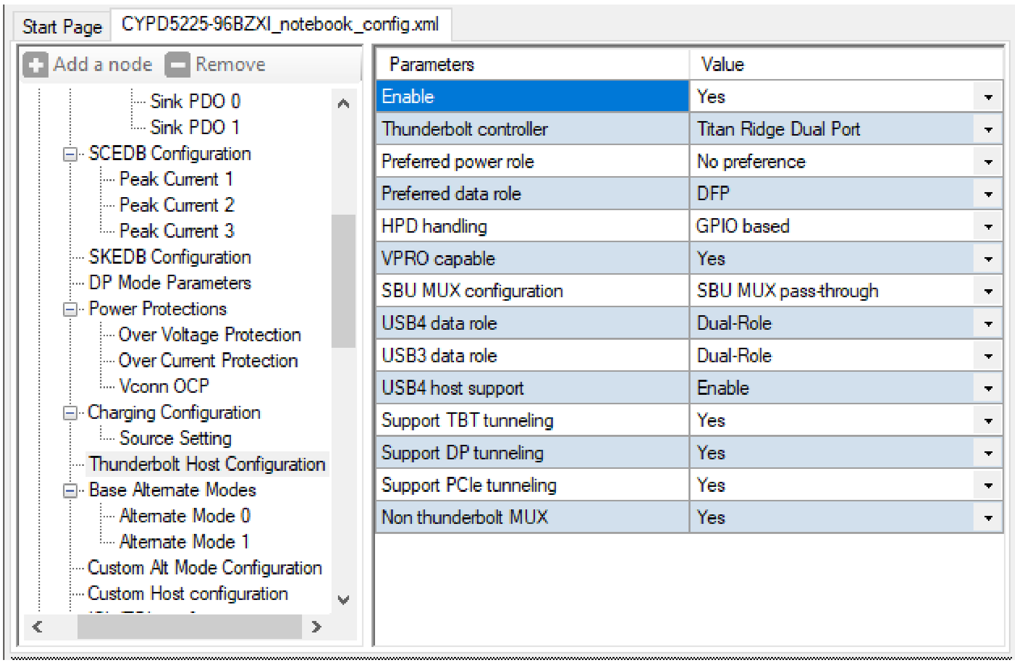


Parameters	Value
VIN maximum safe voltage(mV)	19000

Figure 54 VIN fault voltage information

4.1.16 Thunderbolt host configurations

This node contains Thunderbolt host configuration parameters. The Thunderbolt host configurations are only applicable for CCG5, CCG5C, CCG6, CCG6DF, and CCG6SF device applications.



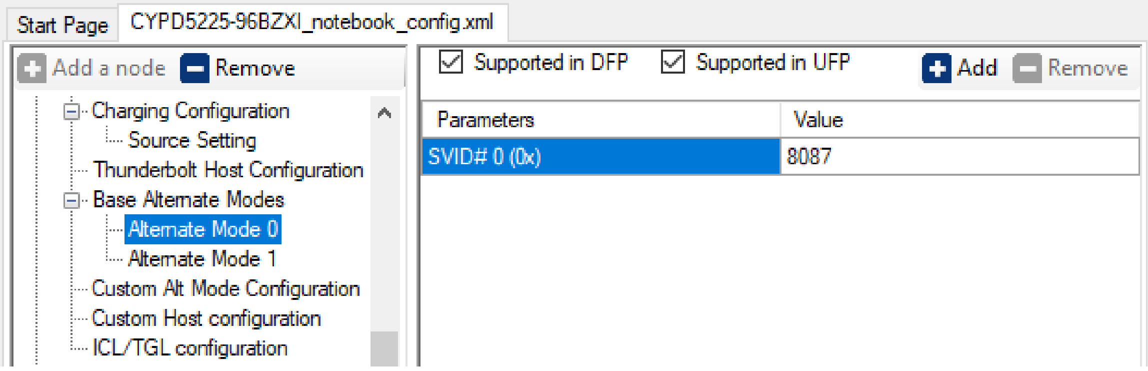
Parameters	Value
Enable	Yes
Thunderbolt controller	Titan Ridge Dual Port
Preferred power role	No preference
Preferred data role	DFP
HPD handling	GPIO based
VPRO capable	Yes
SBU MUX configuration	SBU MUX pass-through
USB4 data role	Dual-Role
USB3 data role	Dual-Role
USB4 host support	Enable
Support TBT tunneling	Yes
Support DP tunneling	Yes
Support PCIe tunneling	Yes
Non thunderbolt MUX	Yes

Figure 55 Thunderbolt host configurations

4.1.17 Base alternate modes configuration

This node contains information about alternate modes and simultaneous entry with other alternate modes. These alternate mode definitions are added/deleted using the **Add a node** and **Remove** buttons on the left pane. Once an alternate mode has been added to the project, the following fields need to be configured:

- **SVID# 0**: SVID corresponding to this alternate mode. This is mandatory field.
- Optionally, add SVIDs which could be processed/entered simultaneously with the main SVID (SVID# 0) using **Add** button on the right panel.
- Check **Supported in DFP** checkbox if the alternate mode is supported in DFP data role.
- Check **Supported in UFP** checkbox if the alternate mode is supported in UFP data role.



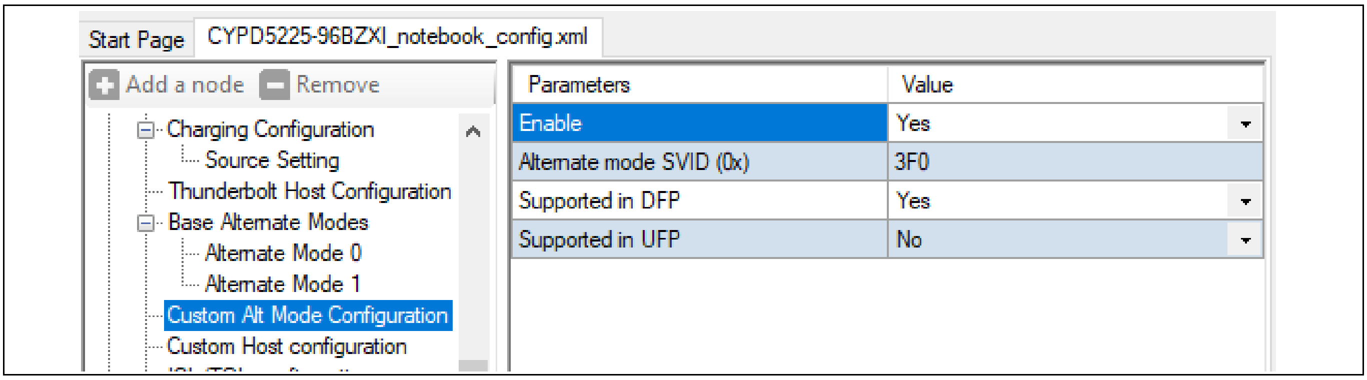
☑ Supported in DFP ☑ Supported in UFP

Parameters	Value
SVID# 0 (0x)	8087

Figure 56 Base alternate modes configuration

4.1.18 Custom alternate mode configuration

This node contains custom alternate mode configuration parameters. Custom alternate mode could be run simultaneously with any alternate mode from the base alternate mode configurations.

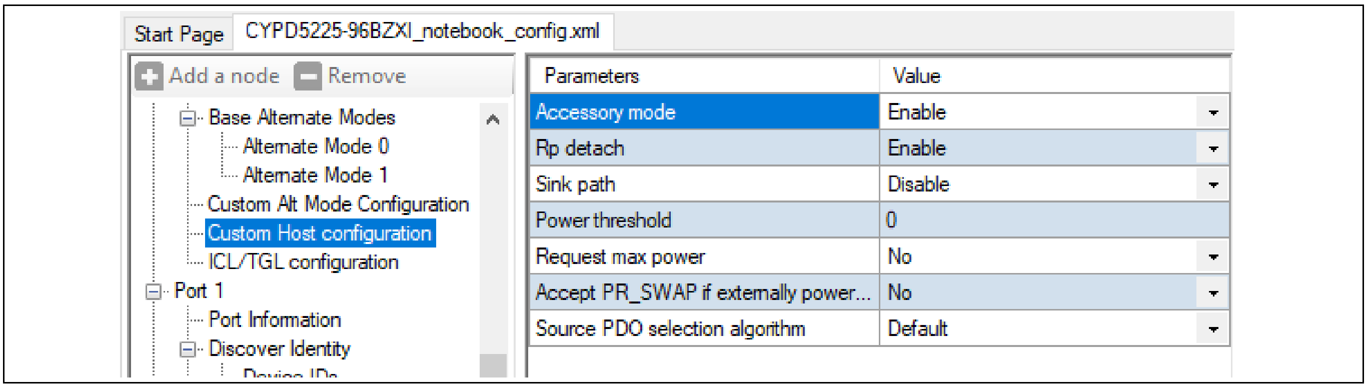

Start Page
CYPD5225-96BZXI_notebook_config.xml
Add a node
Remove
Charging Configuration
Source Setting
Thunderbolt Host Configuration
Base Alternate Modes
Alternate Mode 0
Alternate Mode 1
Custom Alt Mode Configuration
Custom Host configuration

Parameters	Value
Enable	Yes
Alternate mode SVID (0x)	3F0
Supported in DFP	Yes
Supported in UFP	No

Figure 57 Custom alternate mode configuration

4.1.19 Custom host configuration

This node contains custom host configuration parameters.

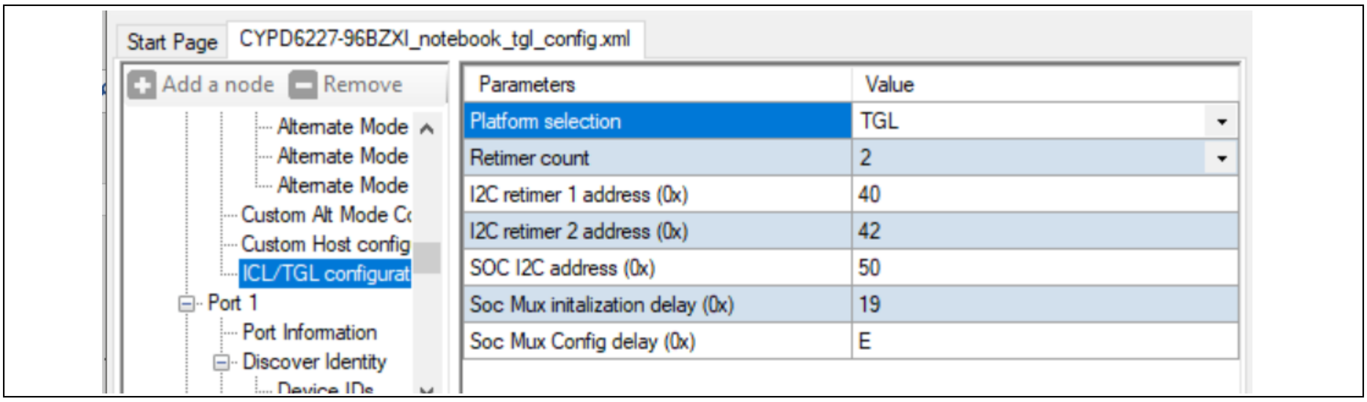


Parameters	Value
Accessory mode	Enable
Rp detach	Enable
Sink path	Disable
Power threshold	0
Request max power	No
Accept PR_SWAP if externally power...	No
Source PDO selection algorithm	Default

Figure 58 Custom host configuration

4.1.20 ICL/TGL configuration

This node contains Ice Lake/Tiger Lake (ICL/TGL) configuration settings. ICL/TGL Platform selection configuration parameter value should be the same for both ports, however it can be edited only in Port 0.

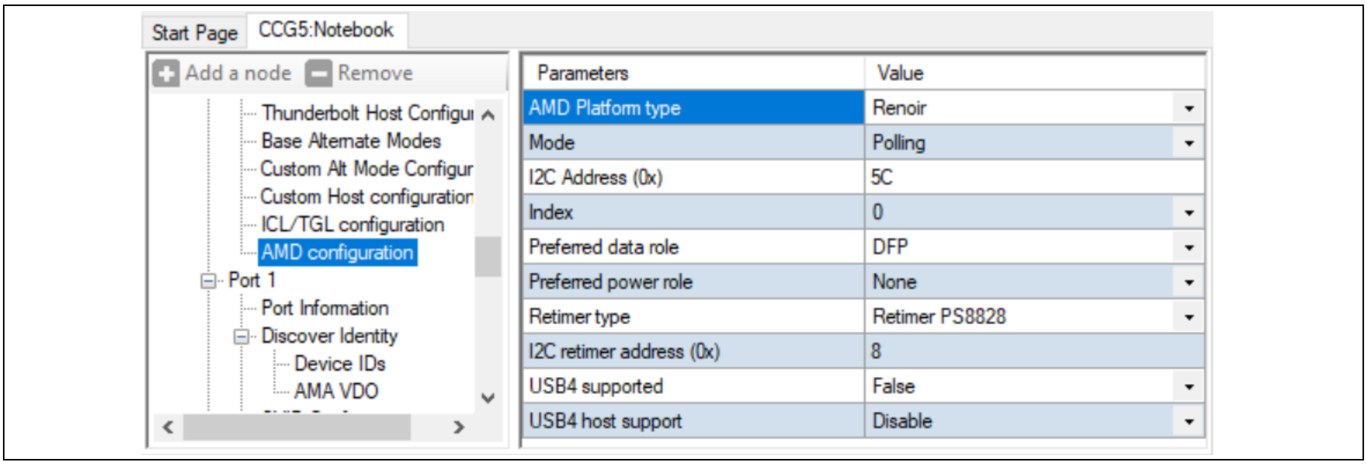


Parameters	Value
Platform selection	TGL
Retimer count	2
I2C retimer 1 address (0x)	40
I2C retimer 2 address (0x)	42
SOC I2C address (0x)	50
Soc Mux initalization delay (0x)	19
Soc Mux Config delay (0x)	E

Figure 59 ICL/TGL configuration

4.1.21 AMD configuration

This node contains AMD configuration parameters.



Parameters	Value
AMD Platform type	Renoir
Mode	Polling
I2C Address (0x)	5C
Index	0
Preferred data role	DFP
Preferred power role	None
Retimer type	Retimer PS8828
I2C retimer address (0x)	8
USB4 supported	False
USB4 host support	Disable

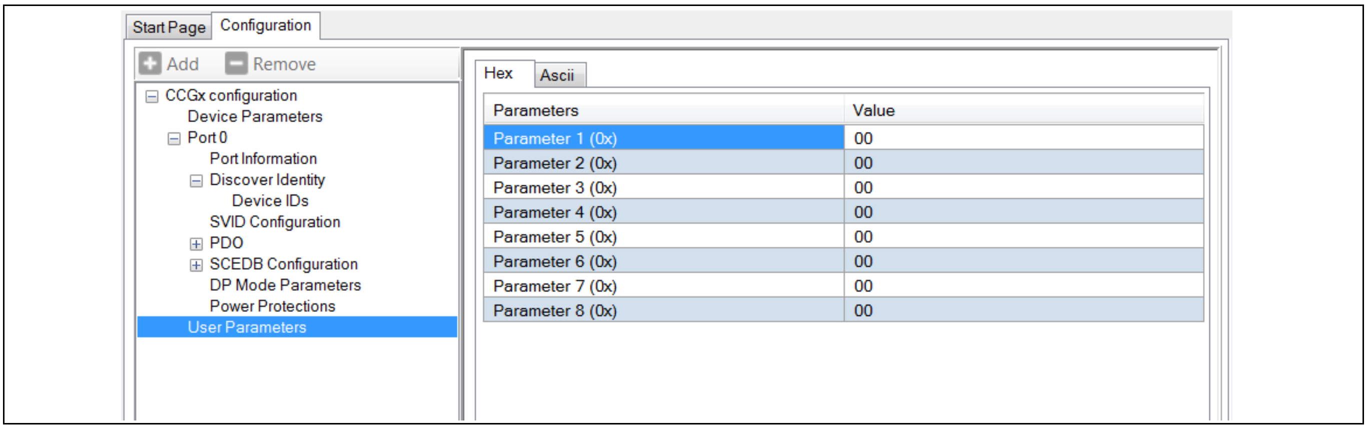
Figure 60 AMD configuration

4.1.22 User parameters

The CCGx device configuration table also provides a 32-byte region where user-defined parameters can be saved. This data is not used by the CCGx firmware provided by Cypress or the configuration utility itself. These values can be programmed through the configuration utility, and can be read out by a controller talking to the CCGx device.

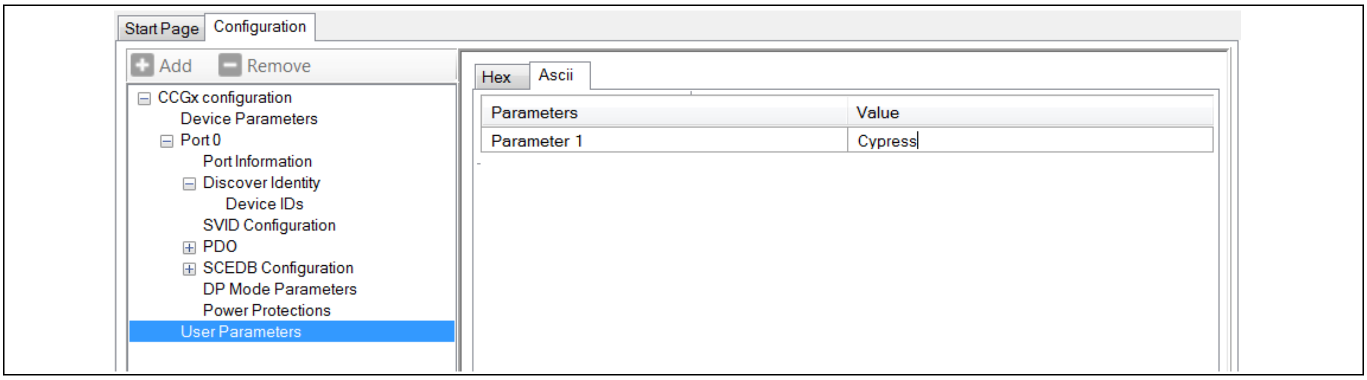
The 32-byte wide user parameters can be input using either of the following:

- A set of eight 4-byte hexadecimal values. This method allows any parameter value to be entered.
- An ASCII string of 32-bytes or lesser. Only printable characters can be input in this case.



Parameters	Value
Parameter 1 (0x)	00
Parameter 2 (0x)	00
Parameter 3 (0x)	00
Parameter 4 (0x)	00
Parameter 5 (0x)	00
Parameter 6 (0x)	00
Parameter 7 (0x)	00
Parameter 8 (0x)	00

Figure 61 User parameters in CCGx configuration – hexadecimal input



Parameters	Value
Parameter 1	Cypress

Figure 62 User parameters in CCGx configuration - ASCII input

Note: Any non-printable characters in the hexadecimal input will be lost when switching to the ASCII string view. Use only one form of input for each CCGx configuration project.

4.2 Billboard device configuration

The Billboard Controller is a standard USB device and not a USB-PD controller. Therefore, the configuration parameters applicable to Billboard Devices are different from those for the USB-PD Devices. The following need to be configured for Billboard controllers:

- Device parameters
- Flashing parameters
- Billboard parameters
- Alternate modes

4.2.1 Device parameters

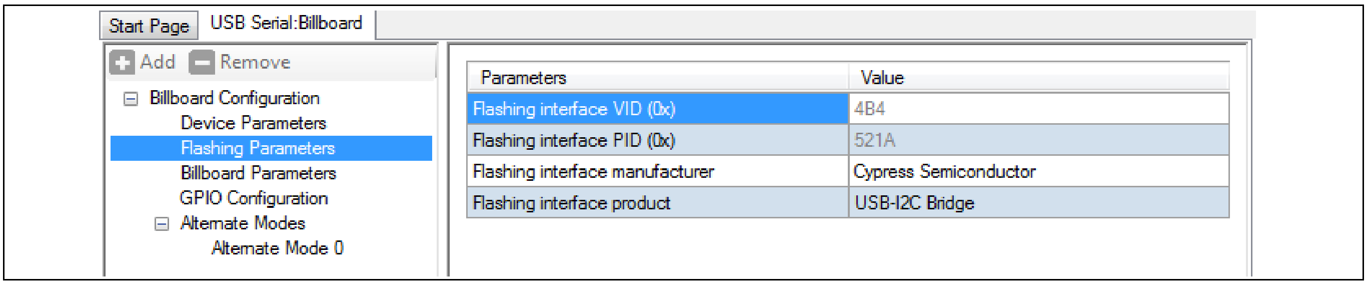
The device parameters section for Billboard Devices is similar to that for USB-PD devices, and is used to select the part number used for Billboard implementation.

4.2.2 Flashing parameters

The Billboard device implementation can have following modes of operation:

- Billboard mode: In this mode, the Billboard device provides the Binary Object Store (BOS) descriptors that allows the USB host to identify the capabilities of the USB-PD controller associated with the Billboard device. In addition to the Billboard interface, this mode also supports a Human Interface Device (HID) function, which can be used to control the device.
- Flashing mode: The Billboard and HID interfaces do not allow you to update the configuration or firmware on the USB-PD controller or the Billboard interface itself. Such functionality is provided through a special flashing or programming mode. The switch from the billboard to the flashing mode is achieved by sending a specific HID report command.

The Flashing Parameters Node is used to specify the properties of the flashing interface of the Billboard device.



Parameters	Value
Flashing interface VID (0x)	4B4
Flashing interface PID (0x)	521A
Flashing interface manufacturer	Cypress Semiconductor
Flashing interface product	USB-I2C Bridge

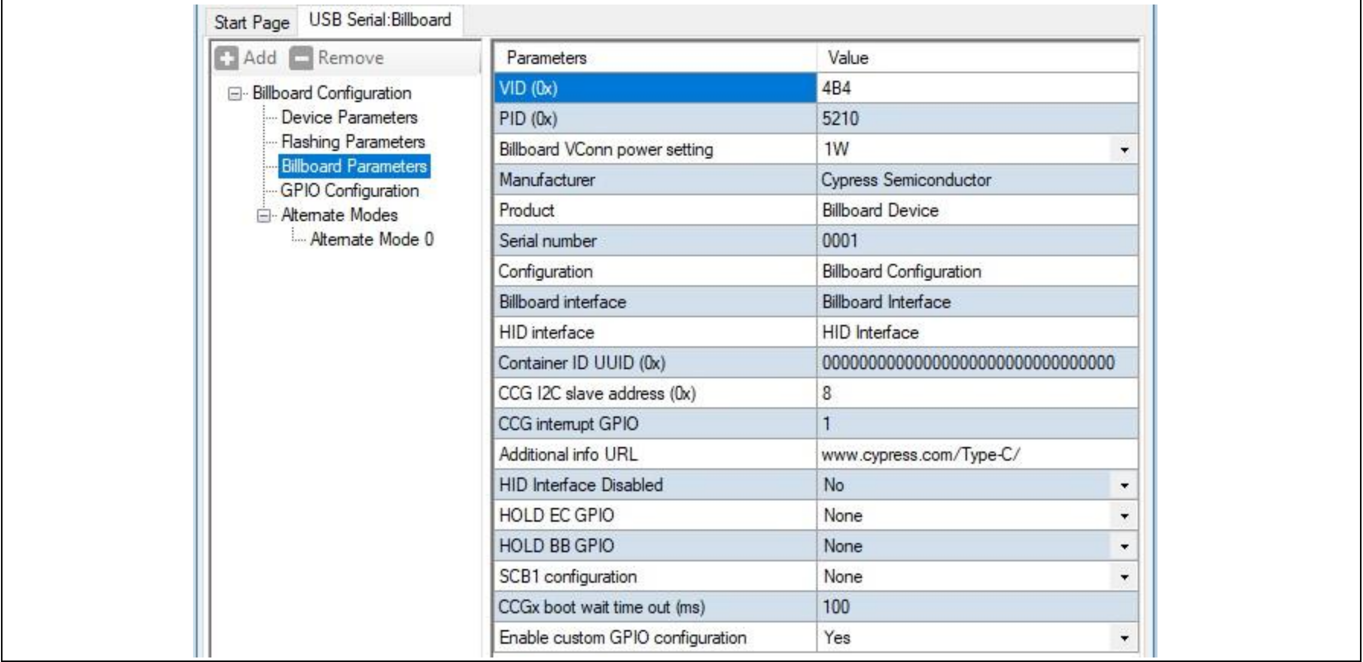
Figure 63 Flashing parameter configuration

The **Flashing interface VID** and **Flashing interface PID** values cannot be changed as these are used to bind the Billboard device to the Cypress USB driver used for programming. The **Flashing interface manufacturer** and **Flashing interface product** strings can be changed as required.

4.2.3 Billboard parameters

The Billboard Parameters node specifies parameters associated with the Billboard mode operation. These parameters include:

- **VID** and **PID** used to identify the device
- **Manufacturer**, **Product**, **Serial number**, **Configuration**, and **Billboard interface** strings used to identify the device.
- **Container ID UUID** value to be reported in the BOS descriptor.
- **Billboard VConn power** setting specifies amount of power required by the USB-PD device from the VConn supply.
- **CCG I2C slave address** is used to configure the I2C interface between the Billboard controller and the USB-PD controller.
- **Additional info URL** provides more information about the device.
- **HOLD EC GPIO** and **HOLD BB GPIO** are used to avoid simultaneous access of CCGx HPI by billboard controller and EC.
- **SCB1 configuration** is used to configure SCB1 block.



Parameters	Value
VID (0x)	4B4
PID (0x)	5210
Billboard VConn power setting	1W
Manufacturer	Cypress Semiconductor
Product	Billboard Device
Serial number	0001
Configuration	Billboard Configuration
Billboard interface	Billboard Interface
HID interface	HID Interface
Container ID UUID (0x)	00000000000000000000000000000000
CCG I2C slave address (0x)	8
CCG interrupt GPIO	1
Additional info URL	www.cypress.com/Type-C/
HID Interface Disabled	No
HOLD EC GPIO	None
HOLD BB GPIO	None
SCB1 configuration	None
CCGx boot wait time out (ms)	100
Enable custom GPIO configuration	Yes

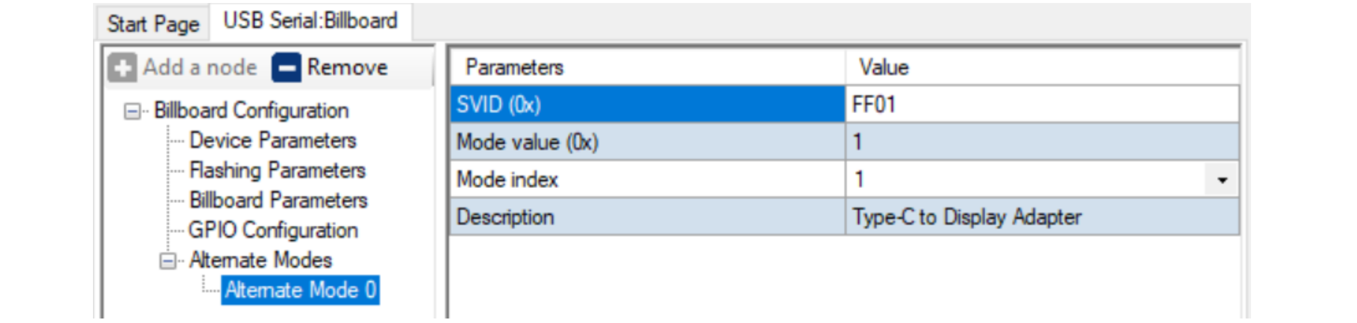
Figure 64 Billboard parameter configuration

4.2.4 Alternate modes

The main functionality of the Billboard device is to report the various alternate modes supported by the corresponding USB-PD controller.

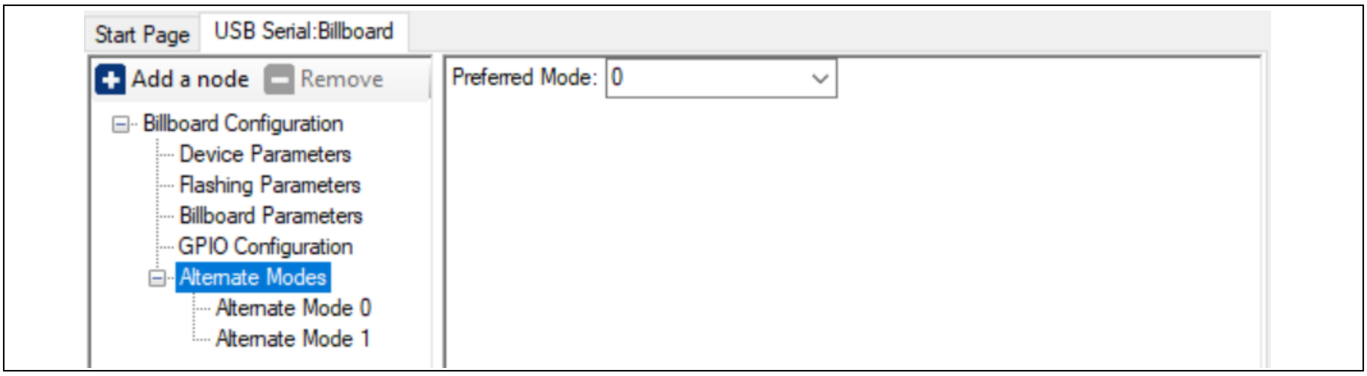
The number of alternate modes that can be reported through the Cypress Billboard implementation varies from 1 to 8. You can add and delete these alternate mode definitions using the **Add** and **Delete** buttons on the left pane. Once an alternate mode has been added to the project, the following fields need to be configured:

- **SVID:** The Standard or Vendor ID (SVID) corresponding to this mode.
- **Mode index**: The index corresponding to this mode.
- **Description**: A descriptive string associated with this mode.
- **Mode value**: Vendor defined mode value.
- **Preferred Mode**: Preferred alternate mode. Select the mode from drop-down list as shown in **Figure 66** (only one can be preferred mode).



Parameters	Value
SVID (0x)	FF01
Mode value (0x)	1
Mode index	1
Description	Type-C to Display Adapter

Figure 65 Alternate mode configuration



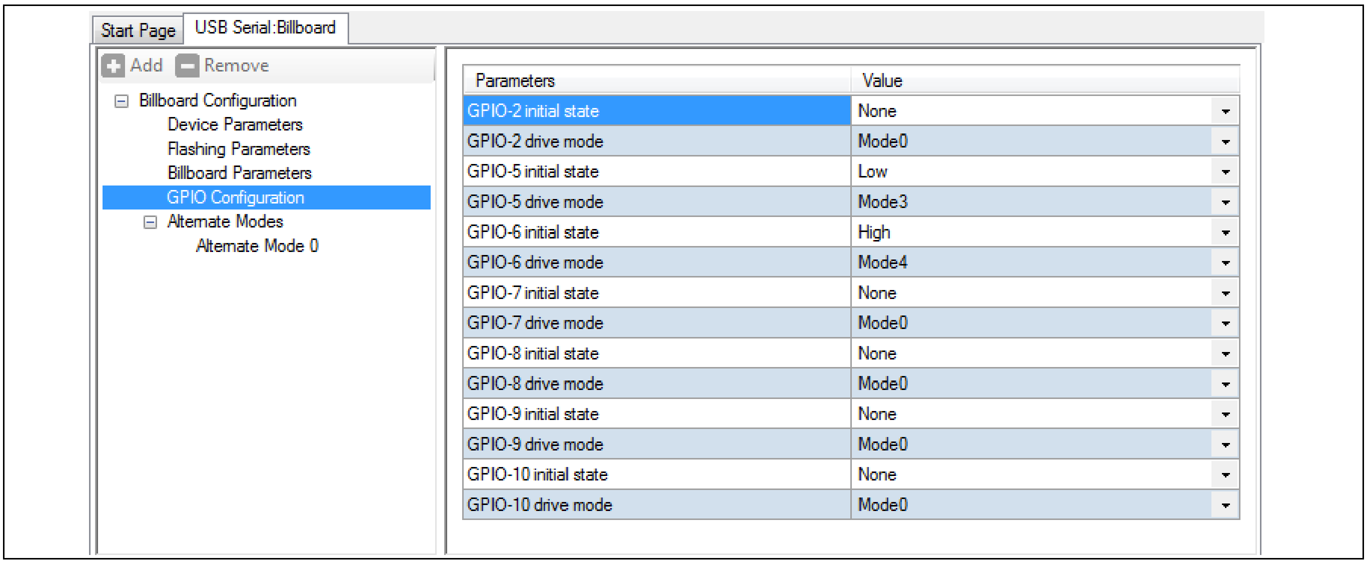
Preferred Mode: 0

Figure 66 Alternate modes in Billboard

4.2.5 GPIO configuration

Unused GPIOs can be configured as input or output GPIOs. For output GPIOs, the initial state and drive mode can be configured. The initial state of an output GPIO can be set as None, Low, or High using the **GPIO-x initial state** field. The drive mode of output GPIOs can be set as one of the following values.

- Mode 0 (Analog mode, High Z)
- Mode 1 (High Z)
- Mode 2 (Strong pull down, Resistive pull up)
- Mode 3 (Resistive pull down, Strong pull up)
- Mode 4 (Strong pull down, Open drain)
- Mode 5 (Open drain, Strong pull up)
- Mode 6 (Strong pull down, Strong pull up)
- Mode 7 (Resistive pull down, Resistive pull up)



Parameters	Value
GPIO-2 initial state	None
GPIO-2 drive mode	Mode0
GPIO-5 initial state	Low
GPIO-5 drive mode	Mode3
GPIO-6 initial state	High
GPIO-6 drive mode	Mode4
GPIO-7 initial state	None
GPIO-7 drive mode	Mode0
GPIO-8 initial state	None
GPIO-8 drive mode	Mode0
GPIO-9 initial state	None
GPIO-9 drive mode	Mode0
GPIO-10 initial state	None
GPIO-10 drive mode	Mode0

Figure 67 GPIO configuration

4.3 Saving the configuration

The above options complete the configuration required for the USB-PD or Billboard solutions. Once the configuration is complete, follow the menu path **File** > **Save As**.

Every configuration will be saved as an XML file. You can open and edit the saved file, as required.

The configuration is also saved in the form of a Cypress defined binary file called the *cyacd* file. The *cyacd* file will be created with the same filename under the same path as the XML file.

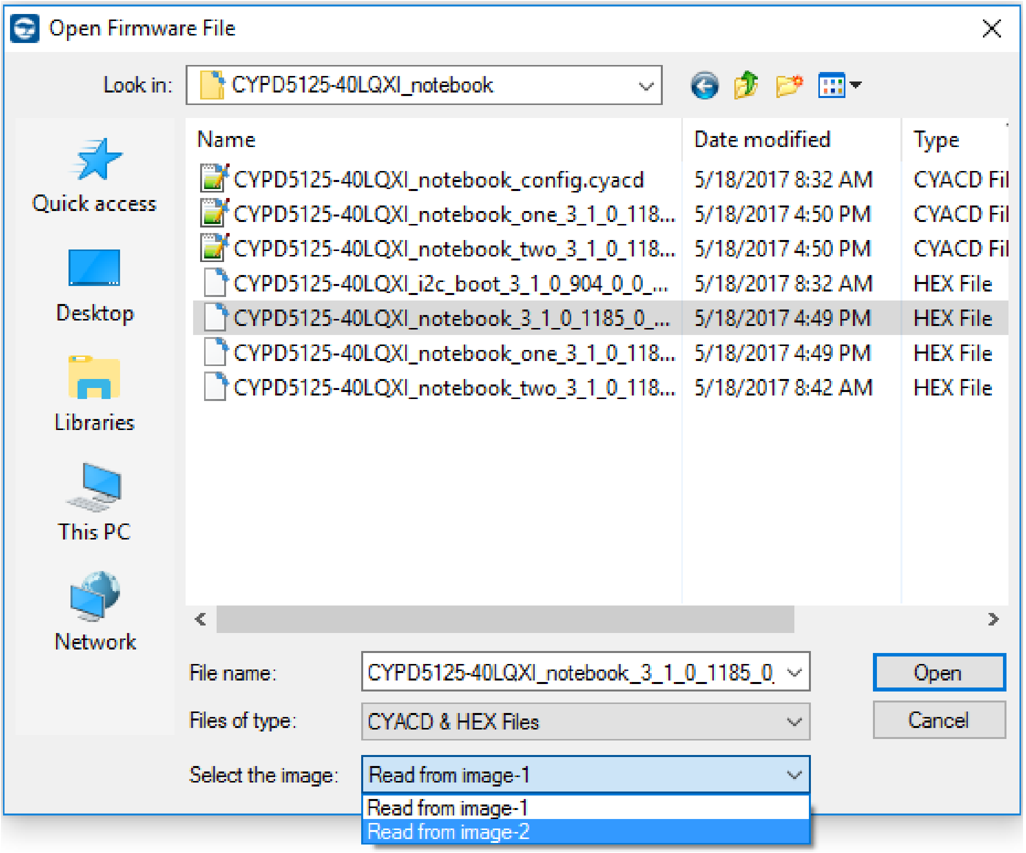
This version of the EZ-PD™ configuration utility also generates a C source file containing the configuration data in the form of an array. This source file can be imported into a CCGx SDK firmware project workspace, so that a firmware binary with the desired default values can be compiled.

4.4 Reading configuration from firmware file

To read the configuration data from the firmware file stored in .hex or .cyacd format, follow the menu path **File** > **Read from Firmware File**.

In CCG1 and CCG2 applications, the *.cyacd* format firmware file does not contain configuration data, hence the utility does not support reading the configuration from the *.cyacd* firmware file of CCG1/2 applications.

In dual firmware applications, the *.hex* format firmware file contains two firmware images and corresponding configurations. EZ-PD™ configuration utility provides an option for selecting the firmware image Id from which configuration has to be read. See **Figure 68**.



Open Firmware File

Look in: CYPD5125-40LQXI_notebook

Name	Date modified	Type
CYPD5125-40LQXI_notebook_config.cyacd	5/18/2017 8:32 AM	CYACD Fil
CYPD5125-40LQXI_notebook_one_3_1_0_118...	5/18/2017 4:50 PM	CYACD Fil
CYPD5125-40LQXI_notebook_two_3_1_0_118...	5/18/2017 4:50 PM	CYACD Fil
CYPD5125-40LQXI_i2c_boot_3_1_0_904_0_0_...	5/18/2017 8:32 AM	HEX File
CYPD5125-40LQXI_notebook_3_1_0_1185_0_...	5/18/2017 4:49 PM	HEX File
CYPD5125-40LQXI_notebook_one_3_1_0_118...	5/18/2017 4:49 PM	HEX File
CYPD5125-40LQXI_notebook_two_3_1_0_118...	5/18/2017 8:42 AM	HEX File

File name: CYPD5125-40LQXI_notebook_3_1_0_1185_0

Files of type: CYACD & HEX Files

Select the image: Read from image-1

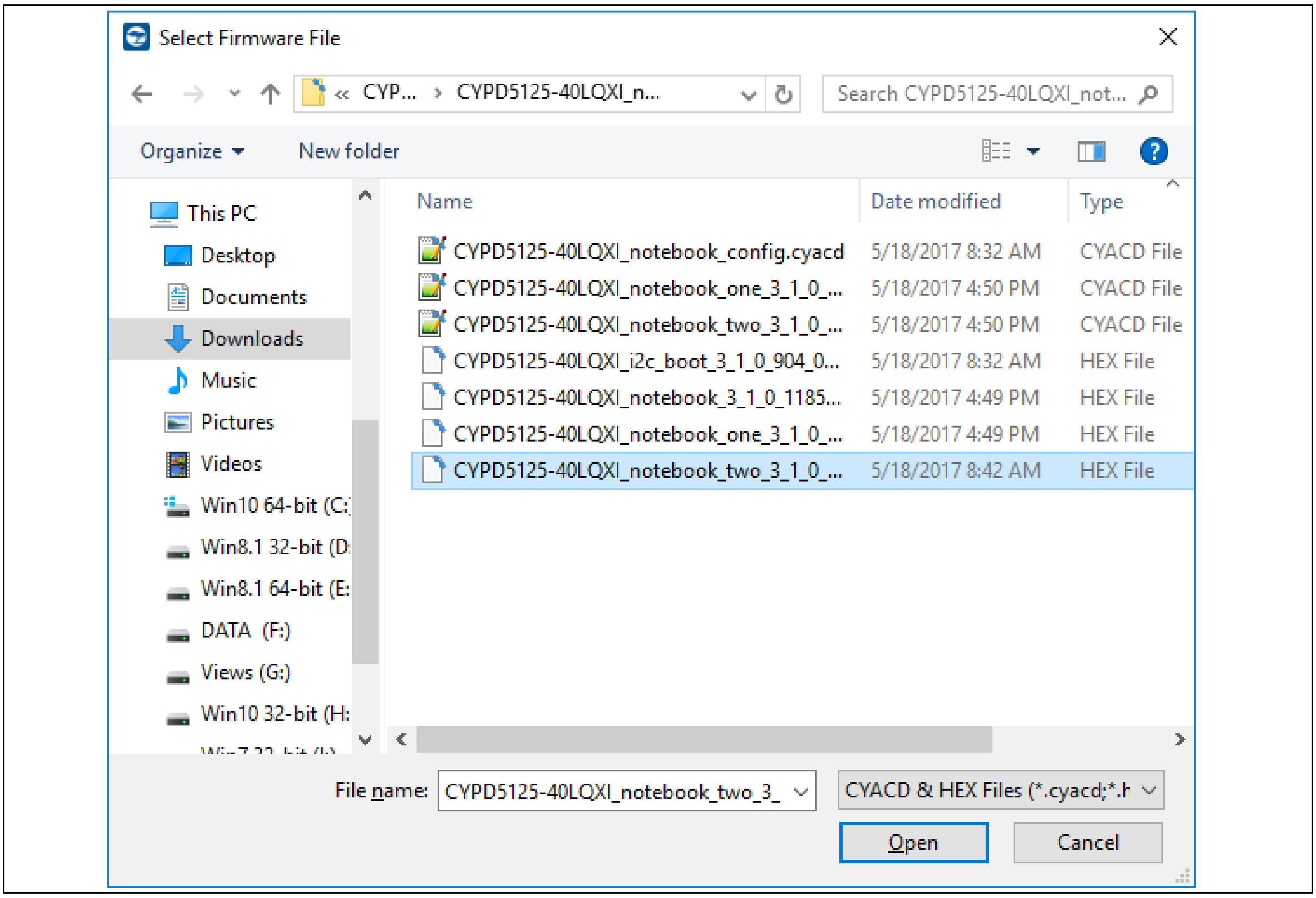
Figure 68 Reading configuration from firmware File

4.5 Saving configuration to firmware file

To save the configuration created or opened in EZ-PD™ configuration utility to a firmware file, follow the menu path **File** > **Save to Firmware File**. The configuration can be saved to either *.cyacd* or *.hex* firmware files.

In CCG1 and CCG2 applications, the *.cyacd* format firmware file does not contain configuration data, hence *.cyacd* firmware file format is not supported.

In dual firmware applications, the *.hex* format firmware file contains two firmware images and corresponding configurations. While saving configurations to dual firmware *.hex* format, EZ-PD™ configuration utility update the configuration of both firmware images.



Name	Date modified	Type
CYPD5125-40LQXI_notebook_config.cyacd	5/18/2017 8:32 AM	CYACD File
CYPD5125-40LQXI_notebook_one_3_1_0_...	5/18/2017 4:50 PM	CYACD File
CYPD5125-40LQXI_notebook_two_3_1_0_...	5/18/2017 4:50 PM	CYACD File
CYPD5125-40LQXI_i2c_boot_3_1_0_904_0...	5/18/2017 8:32 AM	HEX File
CYPD5125-40LQXI_notebook_3_1_0_1185...	5/18/2017 4:49 PM	HEX File
CYPD5125-40LQXI_notebook_one_3_1_0_...	5/18/2017 4:49 PM	HEX File
CYPD5125-40LQXI_notebook_two_3_1_0_...	5/18/2017 8:42 AM	HEX File

File name: CYPD5125-40LQXI_notebook_two_3_

CYACD & HEX Files (*.cyacd;*.h

Figure 69 Saving configuration to firmware file

4.6 Configuring the device

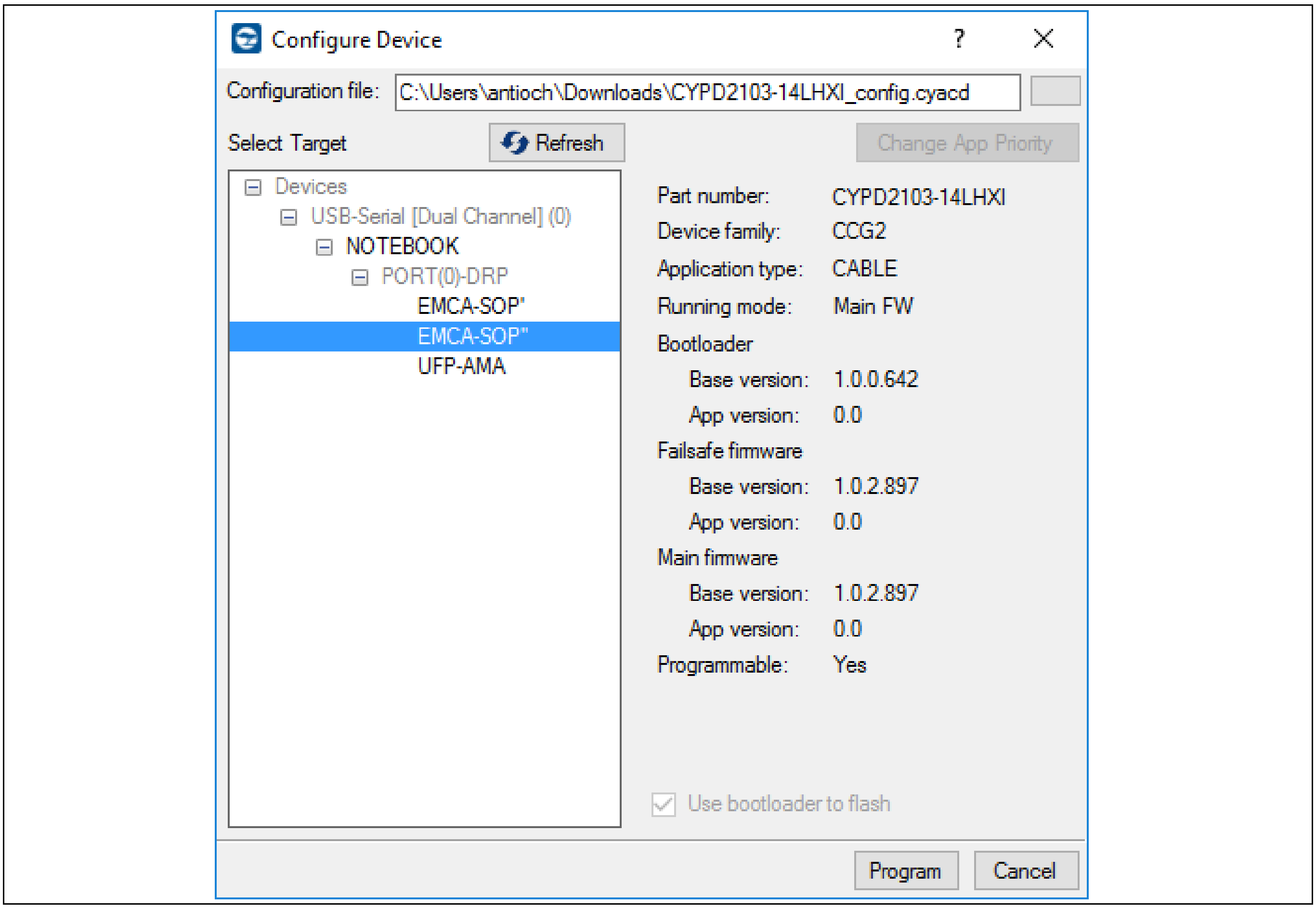
After entering and verifying the relevant values, follow the menu path **Tools** > **Configure Device** to program the configuration to the device.

On the left pane of the Configure Device dialog, the devices are displayed in tree structure, representing the way they are connected. Select a device to view its properties on the right pane,

Click **Program** to initiate programming of the device.

Note: *The status bar at the bottom of the dialog displays the state of the device interface. The program option is only accepted when the device interface is in the ready state.*

Click **Refresh** to start a fresh scan of EZ-PD™ devices connected to the PC host.



Configure Device

Configuration file: C:\Users\antioch\Downloads\CYPD2103-14LHXI_config.cyacd

Part number: CYPD2103-14LHXI
Device family: CCG2
Application type: CABLE
Running mode: Main FW
Bootloader
Base version: 1.0.0.642
App version: 0.0
Failsafe firmware
Base version: 1.0.2.897
App version: 0.0
Main firmware
Base version: 1.0.2.897
App version: 0.0
Programmable: Yes

☑ Use bootloader to flash

Figure 70 **Configuring device dialog box**

4.6.1 Extensions for dual firmware devices

The flash architecture is different for different CCG device families. CCG1/2 devices have only 32 KB of flash memory. CCG3PA devices have 64 KB flash memory and CCG7D devices have 128 KB flash memory. CCG1/2, CCG3PA, PAG1S, and CCG7D devices make use of a single firmware binary with flash access (firmware or configuration update) supported through a dedicated boot-loader application. CCG3/4/5 devices have 128 KB of flash memory, allowing the corresponding solutions to implement a robust dual-firmware solution.

In CCG1, CCG2, CCG3PA, and PAG1S controllers, configuration or firmware update can only be performed by switching the device to a special boot-loader mode. The USB-PD port functionality is not supported in boot-loader mode; which means that these operations will disrupt the USB-PD port operation.

In dual firmware devices, the internal device flash contains two copies of the firmware which can mutually update each other. The USB-PD port operation can continue unaffected while the flash update is in progress. You can reset the dual firmware device at an appropriate time to switch to the newly updated firmware or configuration.

4.6.1.1 Single firmware mode (legacy boot mode)

Since CCG1, CCG2, CCG3PA, and PAG1S devices only support device programming through the boot-loader, only one programming option is available for these devices.

When programming starts, the utility will request the device to jump to bootloader mode where all PD capabilities are disabled. Once the configuration update is complete, the device will be reset into the firmware application mode.

4.6.1.2 Dual firmware mode

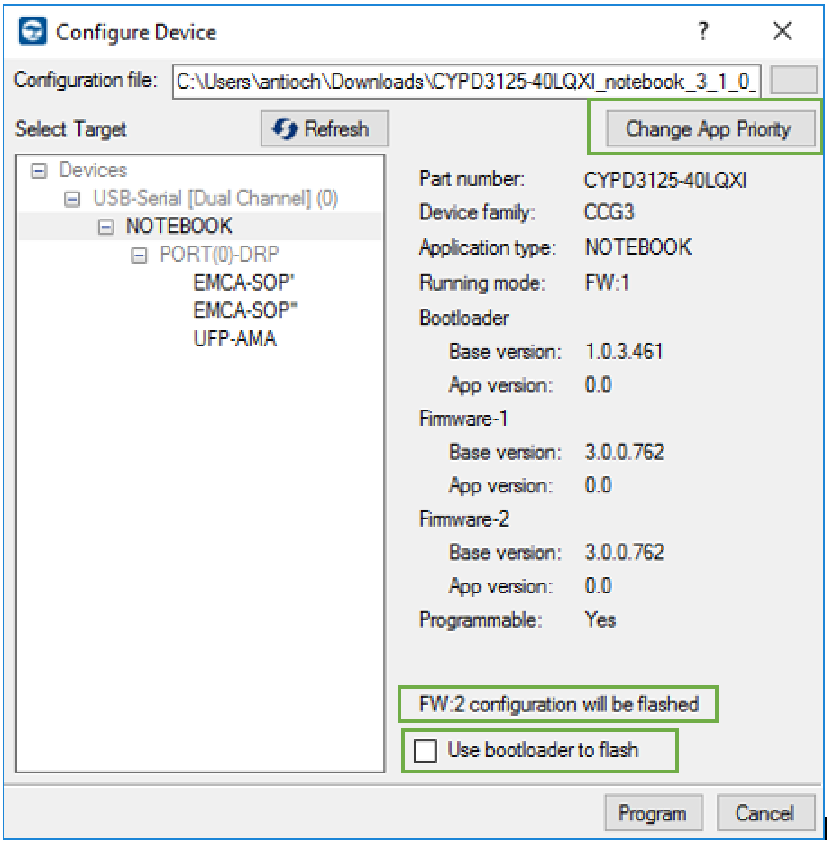
If the device has dual firmware, you can configure the device in either of the following methods:

- Using firmware to flash

  The default behavior of the utility is to configure the alternate firmware without jumping to bootloader. This option will be available only if both firmware copies in the flash are valid (running Firmware is FW1/FW2 and alternate firmware is FW2/FW1). The new configuration will be applied to the alternate firmware image, and the active firmware binary is switched at the end of the configuration update. This will cause the current alternate firmware to become the new running firmware.

  When you click **Program**, the running firmware will update the configuration table of the alternate firmware without interrupting the activities of the CCGx controller. The device will be left in the current state (so as to not disturb the PD port operation) and will start running the alternate firmware when it next goes through a reset.

  If the alternate firmware version is older than the running firmware, the utility will pop-up a warning indicating that the firmware will roll back to an older version at the end of the operation. You have an option to continue or abort the operation at this stage.



Configuration file: C:\Users\antioch\Downloads\CYPD3125-40LQXI_notebook_3_1_0_

Part number: CYPD3125-40LQXI
Device family: CCG3
Application type: NOTEBOOK
Running mode: FW:1
Bootloader
Base version: 1.0.3.461
App version: 0.0
Firmware-1
Base version: 3.0.0.762
App version: 0.0
Firmware-2
Base version: 3.0.0.762
App version: 0.0
Programmable: Yes

FW:2 configuration will be flashed

□ Use bootloader to flash

Figure 71 Configure device dialog box for dual firmware device

- Using bootloader to flash

  This option is available only when at least one valid FW is present (Running firmware is FW1/FW2). This option is used when you want to update the configuration of the active firmware binary without wanting to switch to the other firmware copy.

  The flash update is done by stopping the PD port activity, switching to the boot-loader mode, and then updating the configuration table. The active firmware version will remain the same at the end of the flash update and device reset.

Change App Priority

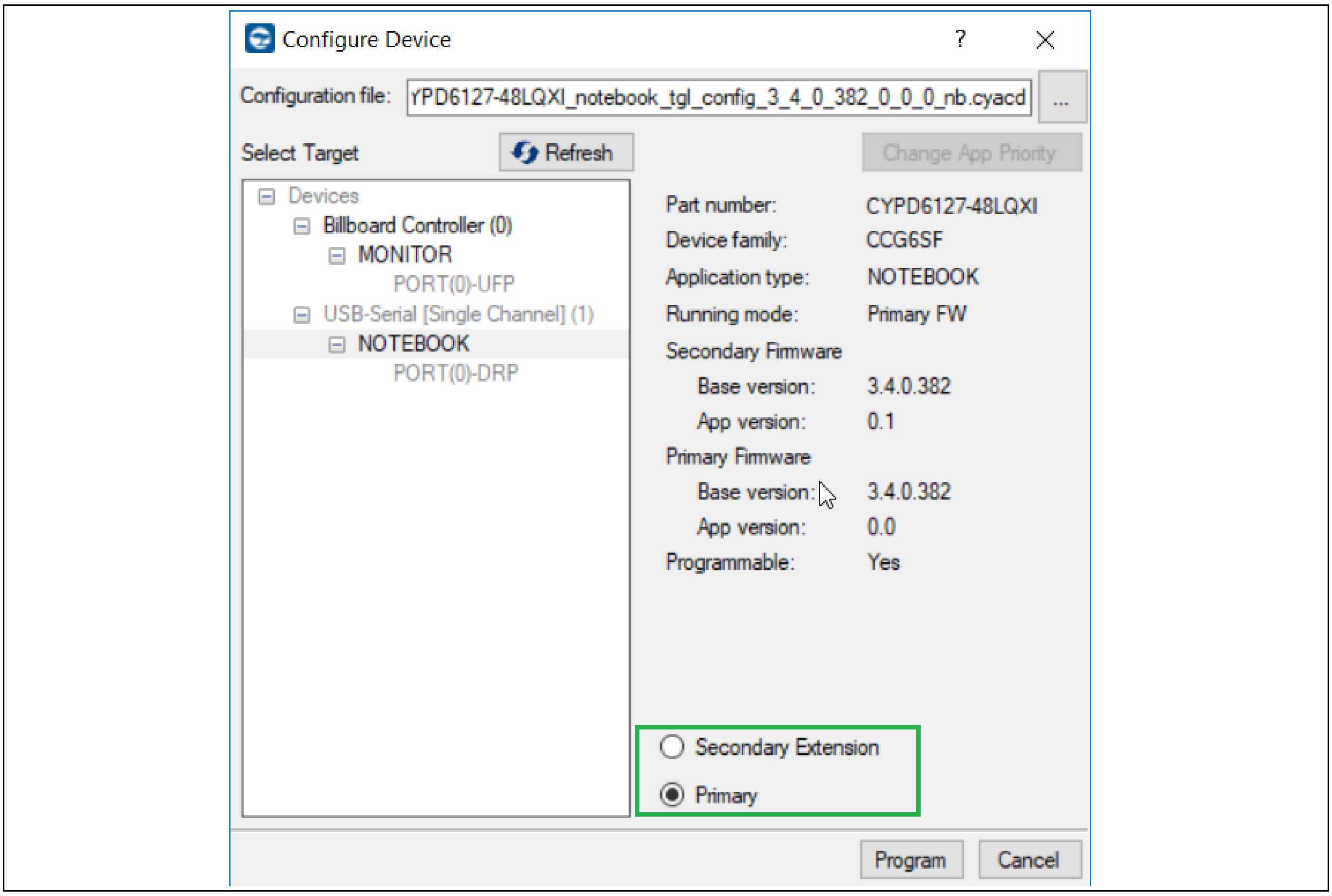
If the device supports dual firmware functionality, running firmware can be switched to alternate firmware by clicking **Change App Priority**. This operation will be successfully performed only when alternate firmware is valid.

4.6.2 Extensions for hybrid architecture applications

The Hybrid flash architecture scheme is currently supported on CCG6DF and CCG6SF devices. There are two major parts to this solution:

1. Primary: This is the main firmware application which supports the necessary Type-C, USB-PD, Alternate Mode, and HPI features. The primary does not have any firmware update capabilities and transfers control to the secondary base when any update is required.
2. Secondary
   a) **Secondary Base:** This is a bootloader application. The secondary base is kept write protected in the CCG device flash and cannot be modified. Since all firmware updates are performed through the secondary base, it also supports a minimal set of Type-C and USB-PD features so that the notebook can be charged while the firmware update is being performed.
   b) **Secondary Extension:** This is an optional binary which augments the Secondary Base application. Since the Secondary Base is write protected, it cannot be updated to add any new functionality that may be desired while the firmware updates are being performed. In such cases, a Secondary Extension application which acts as a set of wrappers around the Secondary Base can be brought into the device flash and executed.

In hybrid architecture solutions, all flash read/write operations are performed through secondary base firmware. The utility provides the option to select the firmware, **Secondary Extension** or **Primary**, for which the configuration should be updated.



Configure Device

Configuration file: YPD6127-48LQXI_notebook_tgl_config_3_4_0_382_0_0_0_nb.cyacd

Part number: CYPD6127-48LQXI
Device family: CCG6SF
Application type: NOTEBOOK
Running mode: Primary FW
Secondary Firmware
Base version: 3.4.0.382
App version: 0.1
Primary Firmware
Base version: 3.4.0.382
App version: 0.0
Programmable: Yes

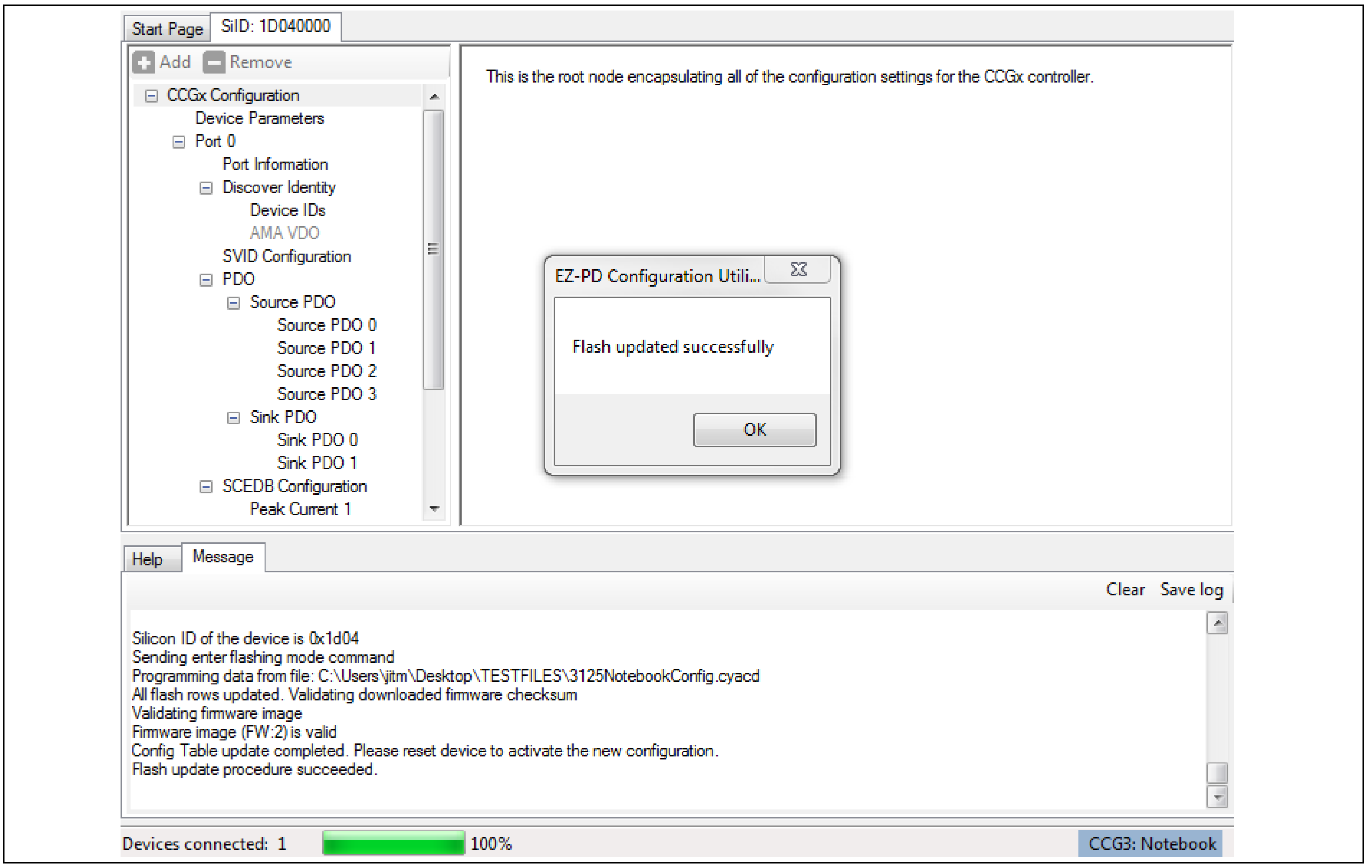
○ Secondary Extension
● Primary

Figure 72 Configure device dialog box for hybrid architecture application

Note:

1. *The tree-view lists all devices that are detected using the USB and CC connections. Since a USB-PD Client is connected through USB to one USB-Serial Device and through CC to the USB-PD Host, the same device could be listed twice.*
2. *Properties of the EMCA and UFP controllers can only be retrieved if the target device supports the CC based flash read/write operations. If the device does not support these operations, the corresponding properties will not be displayed and will be BLANK.*

The configuration is expected to take about 30 seconds and the messages indicating the progress are displayed in the message box. A progress bar also indicates the progress of the operation.



This is the root node encapsulating all of the configuration settings for the CCGx controller.

Flash updated successfully

```
Silicon ID of the device is 0x1d04
Sending enter flashing mode command
Programming data from file: C:\Users\jitm\Desktop\TESTFILES\3125NotebookConfig.cyacd
All flash rows updated. Validating downloaded firmware checksum
Validating firmware image
Firmware image (FW:2) is valid
Config Table update completed. Please reset device to activate the new configuration.
Flash update procedure succeeded.
```

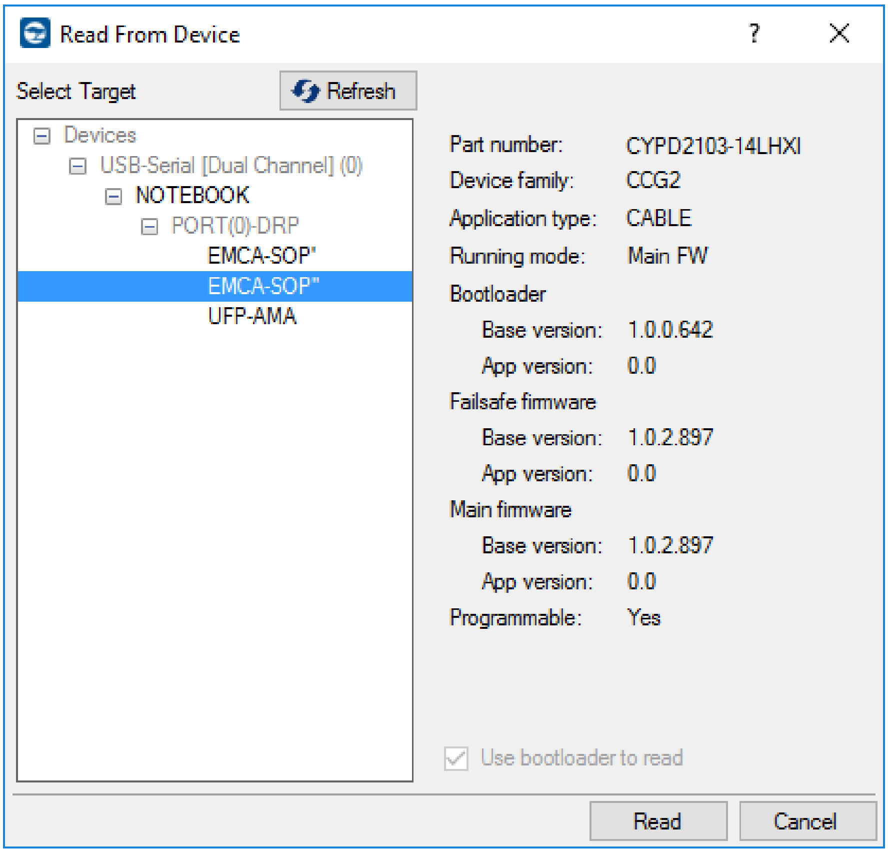
Figure 73 Configure device completion

4.7 Reading configuration from device

To read the Type-C configuration stored on the EZ-PD™ controller connected to the computer and display it, follow the menu path **File** > **Read from Device**.

The Read dialog is similar to the Configure Device dialog and displays the connected devices in a tree structure. You can select a device to read data from.

The configuration read from the device will be displayed using the same screens used for completing the configuration. You can edit, save, program the parameter values back to the controller.



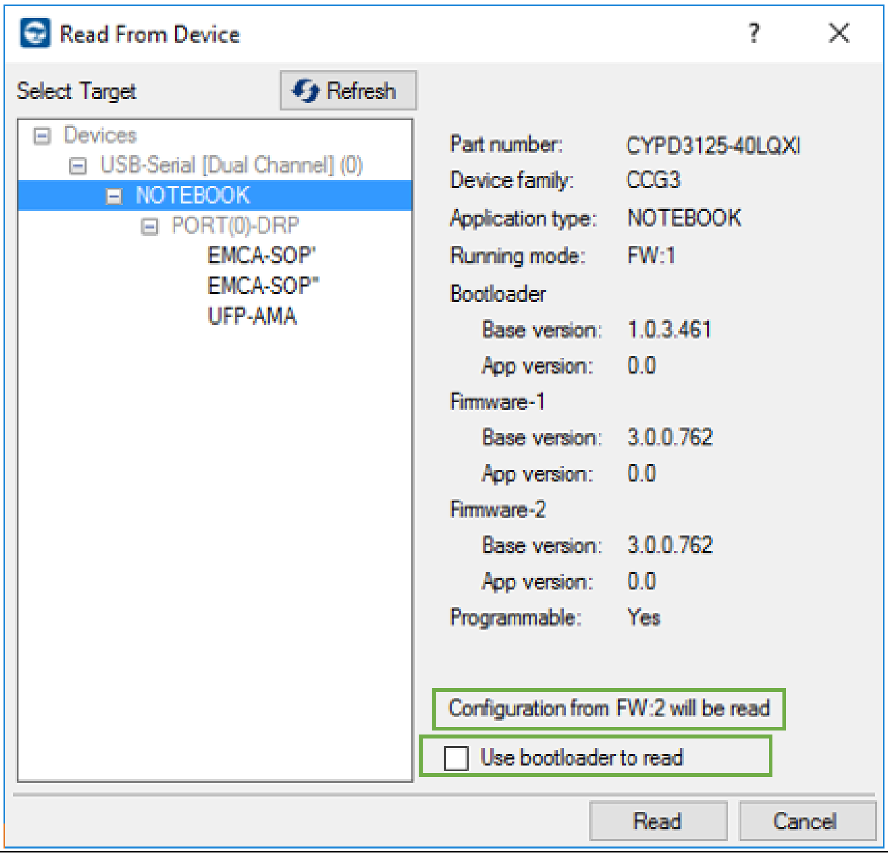
Part number: CYPD2103-14LHXI
Device family: CCG2
Application type: CABLE
Running mode: Main FW
Bootloader
Base version: 1.0.0.642
App version: 0.0
Failsafe firmware
Base version: 1.0.2.897
App version: 0.0
Main firmware
Base version: 1.0.2.897
App version: 0.0
Programmable: Yes

☑ Use bootloader to read

Figure 74 Read device dialog

4.7.1 Extensions for dual firmware devices

As in the case of device configuration, the dual bank firmware implementation has implications on the read functionality as well.



Part number: CYPD3125-40LQXI
Device family: CCG3
Application type: NOTEBOOK
Running mode: FW:1
Bootloader
Base version: 1.0.3.461
App version: 0.0
Firmware-1
Base version: 3.0.0.762
App version: 0.0
Firmware-2
Base version: 3.0.0.762
App version: 0.0
Programmable: Yes

Configuration from FW:2 will be read

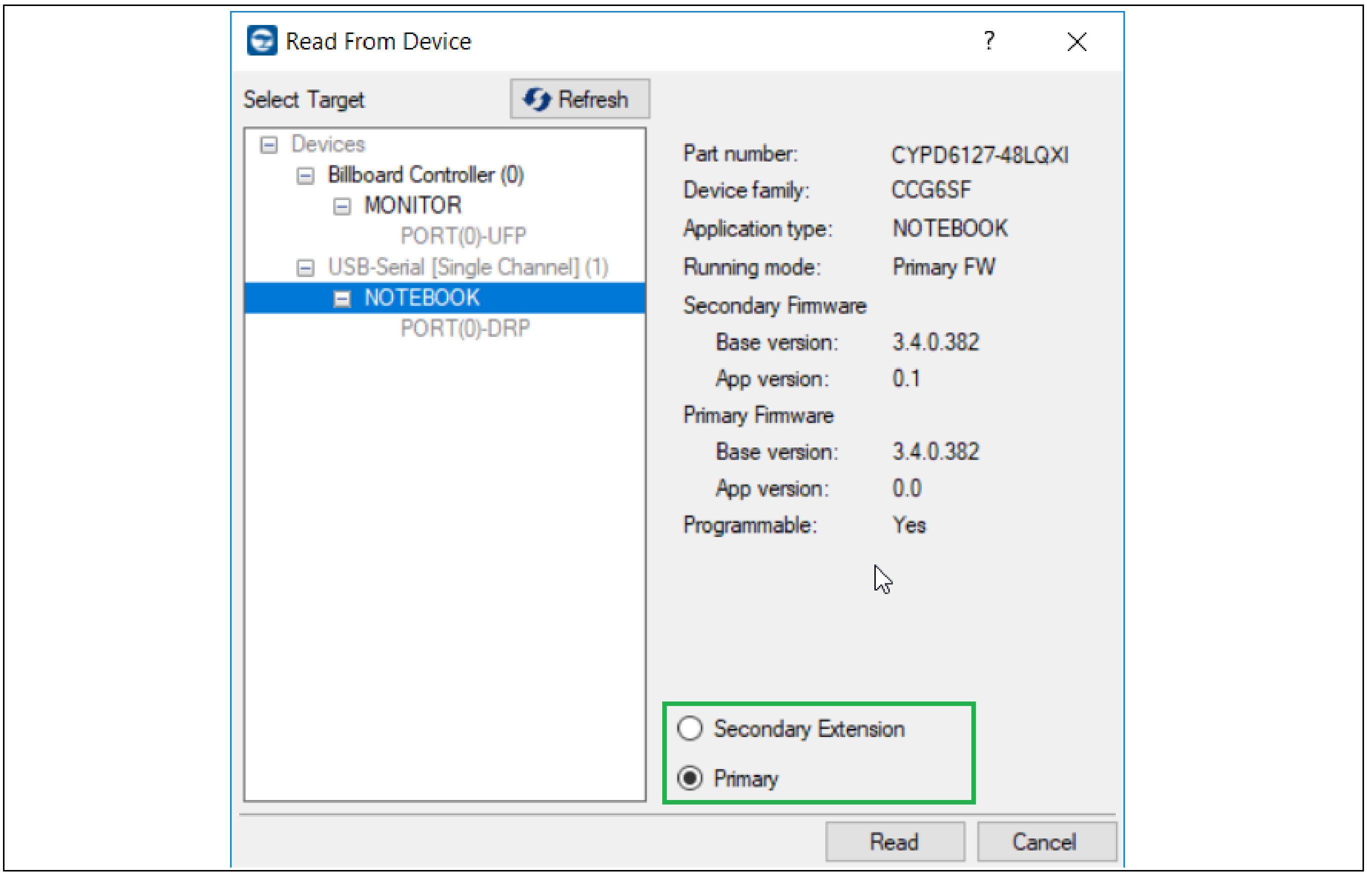
□ Use bootloader to read

Figure 75 Read device dialog box for dual firmware devices

The default behavior of the utility is to read and return the configuration associated with the alternate firmware. You can use the **Use bootloader to read** option, shown in **Figure 75**, to trigger a read of the configuration table associated with the running firmware. However, this operation will require the CCG device to disable the USB-PD connections and revert to bootloader mode for doing the flash reads.

4.7.2 Extensions for hybrid architecture applications

In case of hybrid architecture applications, configuration read operation is performed through secondary base firmware. The utility provides the option to select the firmware, **Secondary Extension** or **Primary**, from which the configuration data should be read.


Read From Device

Select Target

- Devices
  - Billboard Controller (0)
    - MONITOR
      - PORT(0)-UFP
  - USB-Serial [Single Channel] (1)
    - NOTEBOOK
      - PORT(0)-DRP

Part number: CYPD6127-48LQXI
Device family: CCG6SF
Application type: NOTEBOOK
Running mode: Primary FW
Secondary Firmware
Base version: 3.4.0.382
App version: 0.1
Primary Firmware
Base version: 3.4.0.382
App version: 0.0
Programmable: Yes

○ Secondary Extension
● Primary

Figure 76 Read device dialog box for hybrid architecture applications

4.8 Firmware update

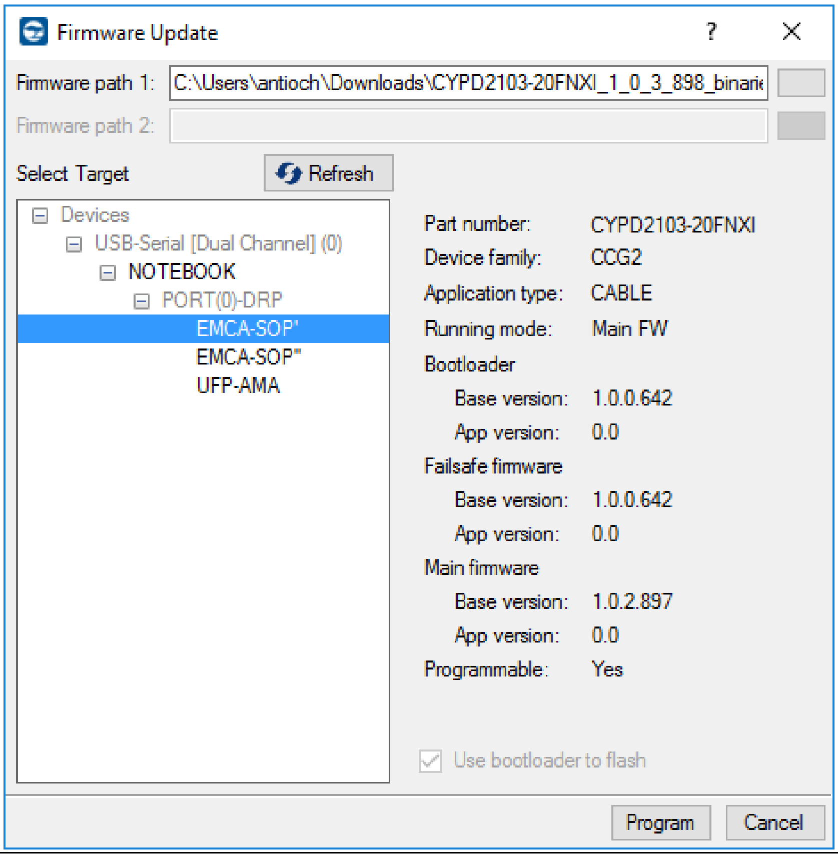
To update the firmware running on the EZ-PD™ or Billboard controller follow the menu path **Tools** > **Firmware Update**.

As in the case of device configuration and configuration load, the firmware update can be performed on any of the supported devices. Select the controller using the Target Device option and select the firmware binary file.

Firmware for the CCGx controllers are provided in *.cyacd* format, and firmware for the Billboard controllers are provided in *.img* format. Sample firmware binaries for each application and standard part numbers are provided under the *Firmware* folder in the utility installation.

The firmware update over CC will take about two minutes to complete. Progress messages will be displayed in the Messages window and a progress bar also shows the progress of the firmware update process.

Attention: Do not disconnect the devices from the computer while the firmware update is in progress.



Firmware Update

Firmware path 1: C:\Users\antioch\Downloads\CYPD2103-20FNXI_1_0_3_898_binarie

Firmware path 2:

Select Target

Part number: CYPD2103-20FNXI

Device family: CCG2

Application type: CABLE

Running mode: Main FW

Bootloader

Base version: 1.0.0.642

App version: 0.0

Failsafe firmware

Base version: 1.0.0.642

App version: 0.0

Main firmware

Base version: 1.0.2.897

App version: 0.0

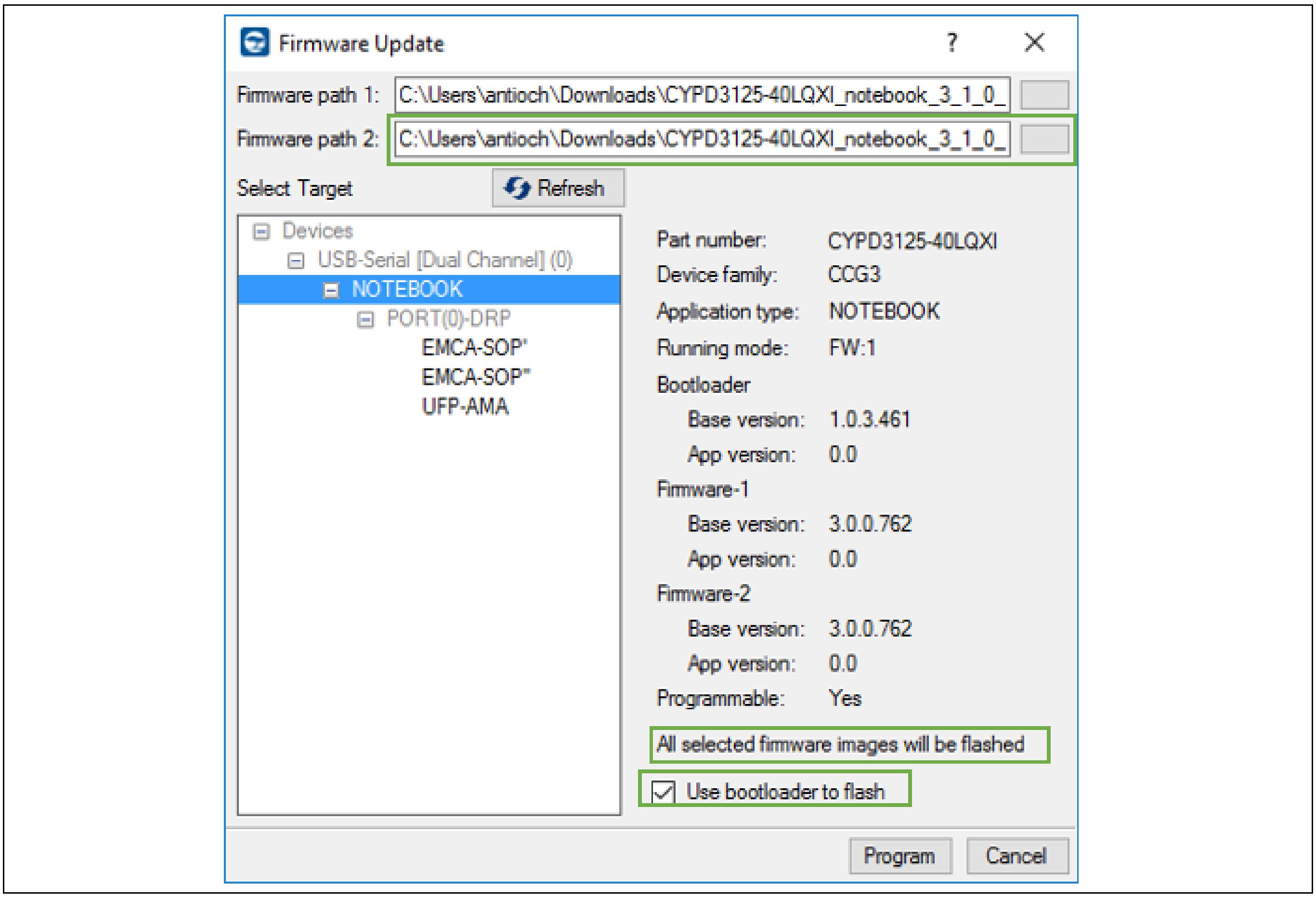
Programmable: Yes

☑ Use bootloader to flash

Figure 77 Firmware update dialog

4.8.1 Extensions for dual firmware devices

You can specify one or both firmware binaries through the firmware update dialog, and the appropriate binary will be used for the update operation.



Firmware path 1: C:\Users\antioch\Downloads\CYPD3125-40LQXI_notebook_3_1_0_

Firmware path 2: C:\Users\antioch\Downloads\CYPD3125-40LQXI_notebook_3_1_0_

Part number:	CYPD3125-40LQXI
Device family:	CCG3
Application type:	NOTEBOOK
Running mode:	FW:1
Bootloader	
Base version:	1.0.3.461
App version:	0.0
Firmware-1	
Base version:	3.0.0.762
App version:	0.0
Firmware-2	
Base version:	3.0.0.762
App version:	0.0
Programmable:	Yes

All selected firmware images will be flashed

☑ Use bootloader to flash

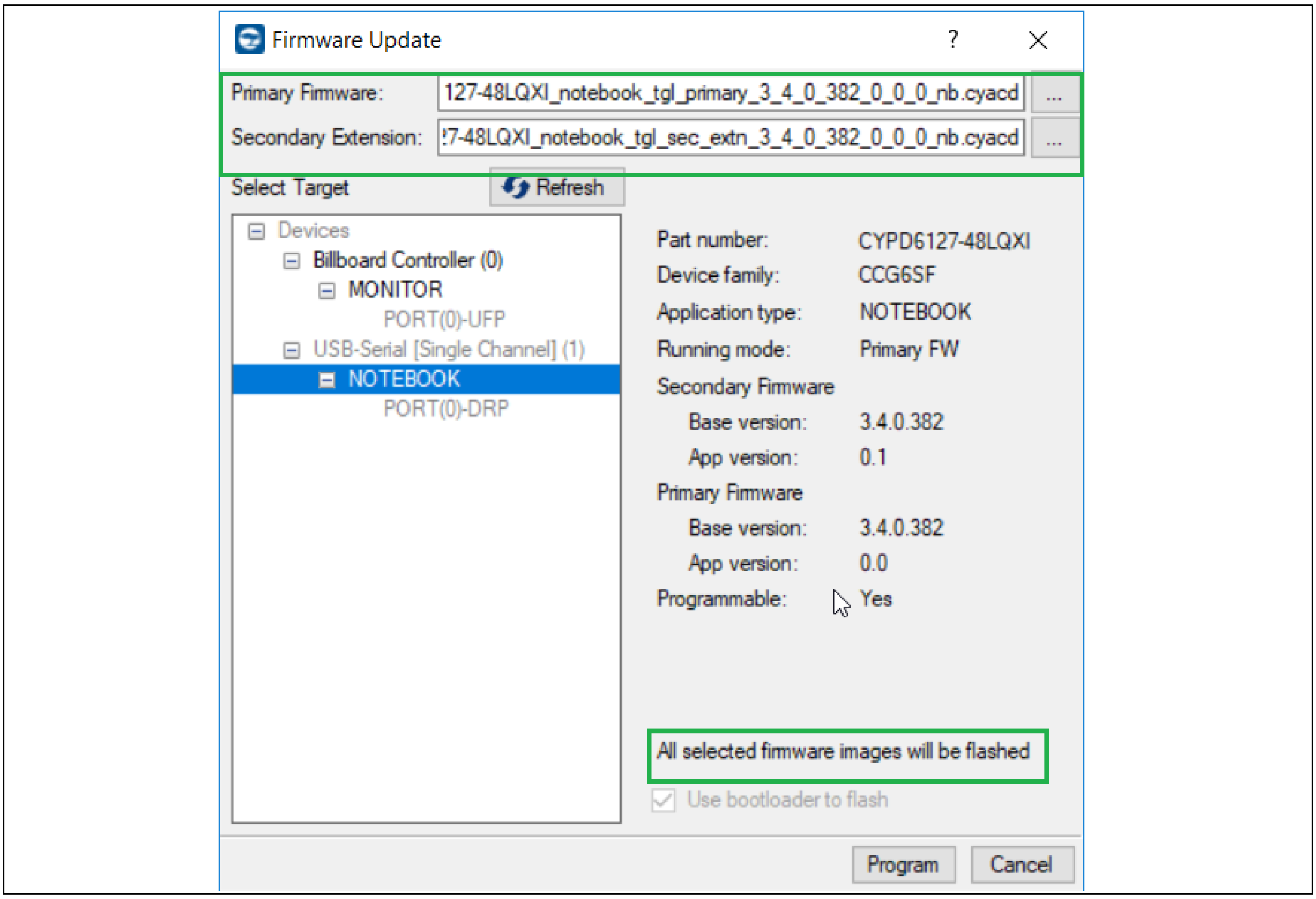
Figure 78 Firmware update dialog for dual firmware devices

If the device is currently running FW1, the utility will select the FW2 binary from the two files passed in, and update FW2 with the new version. If you do not provide the FW2 binary, an error will be reported.

It is possible to update both FW1 and FW2 binaries at once using **Use bootloader for flashing** as shown in **Figure 78**. In this case, CCG will disable all PD ports and revert to bootloader mode so that both firmware banks can be updated.

4.8.2 Extensions for hybrid architecture applications

In case of hybrid architecture applications, firmware update operation is performed through secondary base firmware. To update firmware, browse and select the primary, secondary extension, or both firmware files. Then, click **Program**. The utility will update all selected firmware images.



Primary Firmware: 127-48LQXI_notebook_tgl_primary_3_4_0_382_0_0_0_nb.cyacd

Secondary Extension: !7-48LQXI_notebook_tgl_sec_extn_3_4_0_382_0_0_0_nb.cyacd

Part number:	CYPD6127-48LQXI
Device family:	CCG6SF
Application type:	NOTEBOOK
Running mode:	Primary FW
Secondary Firmware	
Base version:	3.4.0.382
App version:	0.1
Primary Firmware	
Base version:	3.4.0.382
App version:	0.0
Programmable:	Yes

All selected firmware images will be flashed

☑ Use bootloader to flash

Figure 79 Firmware update dialog for hybrid architecture application

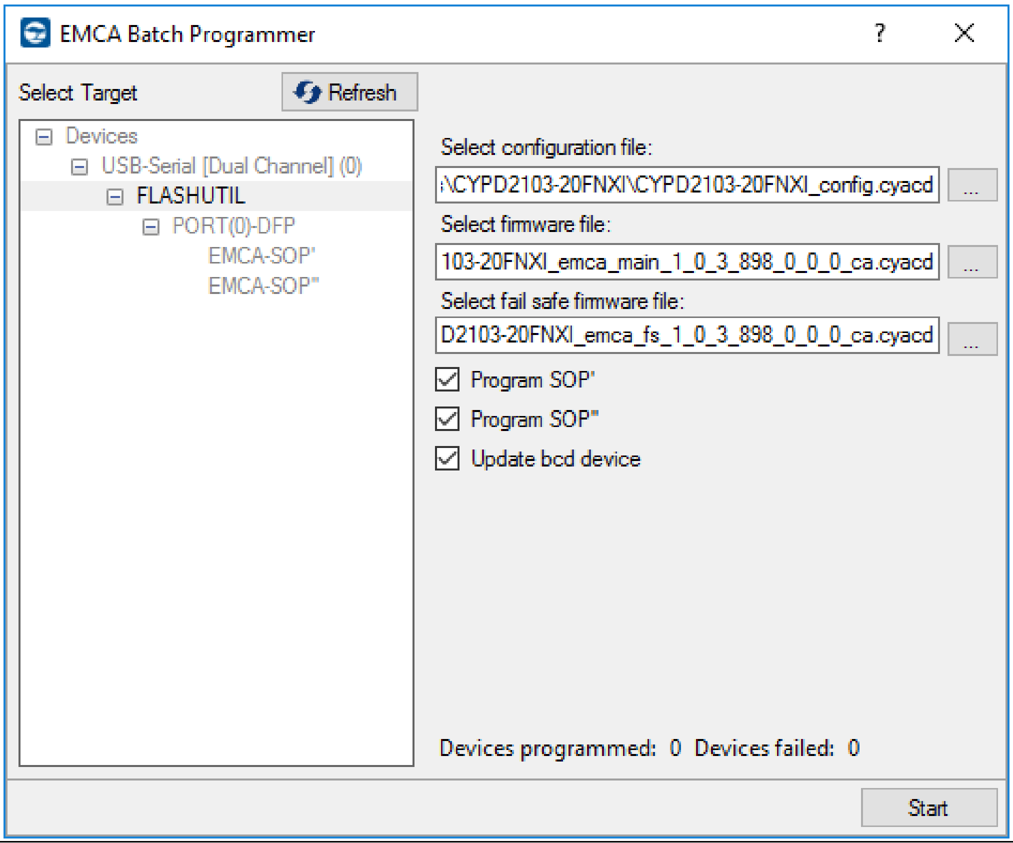
4.9 EMCA batch programming

The EZ-PD™ configuration utility supports a special Batch Programming mode for EMCA devices. This batch programming mode is designed to program the same configuration settings and/or firmware binaries into several USB Type-C cables without going through separate UI operations.

The required settings can be selected at the start of the session, and then the cables to be programmed can be plugged in one-by-one. The utility automatically detects the cable plug-in, programs the CCGx controller and then prompts the user to plug in a new cable.

The batch programmer requires the cables to be connected to a CCG1 Host or Client board (CY4504 or CY4503) running a specially designed flashing firmware. This firmware can be found at the path Firmware\FlashUtil within the utility installation. Separate files are provided for use on the CY4503 and CY4504 kits respectively. The CCG1 host programmed with this firmware will show up on the UI screen as "DFP-FLASHUTIL".

The EMCA Batch Programmer Dialog is used for the EMCA batch programming operation.



Select configuration file: :\CYPD2103-20FNXI\CYPD2103-20FNXI_config.cyacd

Select firmware file: 103-20FNXI_emca_main_1_0_3_898_0_0_0_ca.cyacd

Select fail safe firmware file: D2103-20FNXI_emca_fs_1_0_3_898_0_0_0_ca.cyacd

☑ Program SOP'

☑ Program SOP''

☑ Update bcd device

Devices programmed: 0 Devices failed: 0

Figure 80 EMCA batch programmer dialog

To start the programming, select the FLASHUTIL device to be used for programming. Then, select the configuration file, firmware file, and failsafe firmware file. Select the EMCA controllers to be programmed (near-end SOP', far-end SOP'', or both). The **Update bcd device** option is used to specify that the bcd Device field in the Product VDO should be incremented for each device programmed.

Programming begins once you click **Start**.

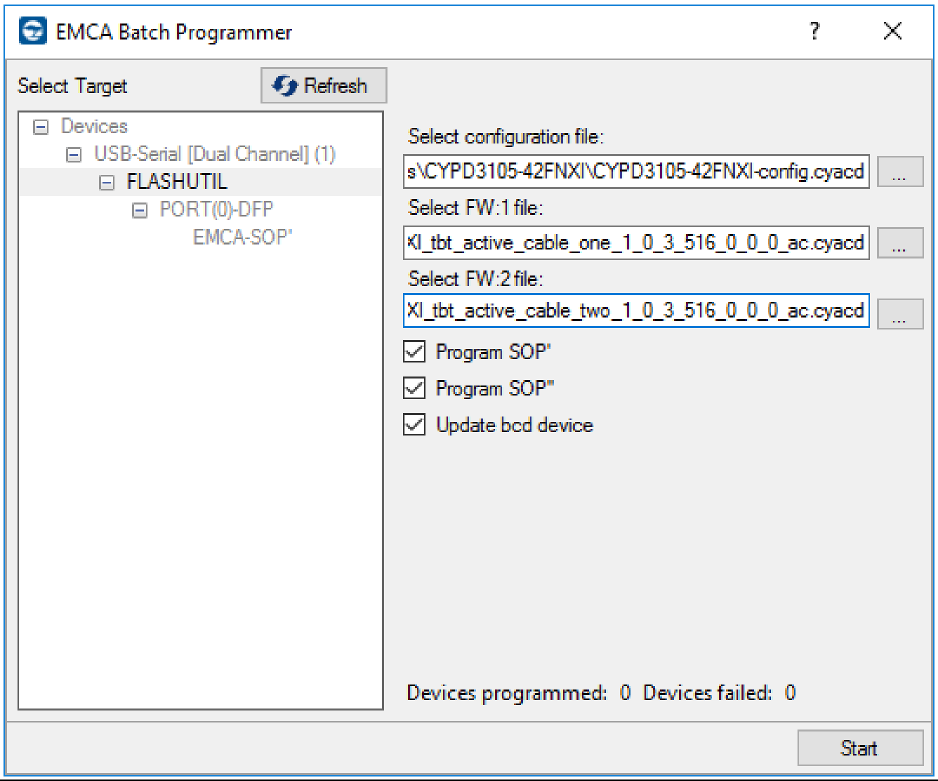
Unlike other operations such as Configure Device, the Batch Programmer dialog stays active while the programming is in progress. You need to explicitly dismiss the dialog, The dialog displays the number of devices programmed and the current status of the operation on the status bar at the bottom. When the status message says “Please plug in new EMCA cable”, remove the current EMCA cable and plug-in a new cable to be programmed. The operation continues until the you press **Stop**, or a fatal error (FLASHUTIL controller error) occurs.

4.9.1 Extension for CCG3 based EMCA programming

If the EMCA controllers to be programmed use CCG3 devices, additional parameters need to be specified as part of the batch programming flow. The following operations can be performed:

- Firmware update of one copy of the firmware
- Firmware update of both copies of firmware
- Configuration update of valid copies of firmware.
- Firmware and configuration update on one or both copies of firmware.

All of these operations are performed through the CCG boot-loader and will involve a device reset.



Select configuration file: s\CYPD3105-42FNXI\CYPD3105-42FNXI-config.cyacd

Select FW:1 file: XI_tbt_active_cable_one_1_0_3_516_0_0_0_ac.cyacd

Select FW:2 file: XI_tbt_active_cable_two_1_0_3_516_0_0_0_ac.cyacd

☑ Program SOP'

☑ Program SOP''

☑ Update bcd device

Devices programmed: 0 Devices failed: 0

Figure 81 EMCA batch programmer dialog for CCG3 devices

4.10 Dongle batch programming

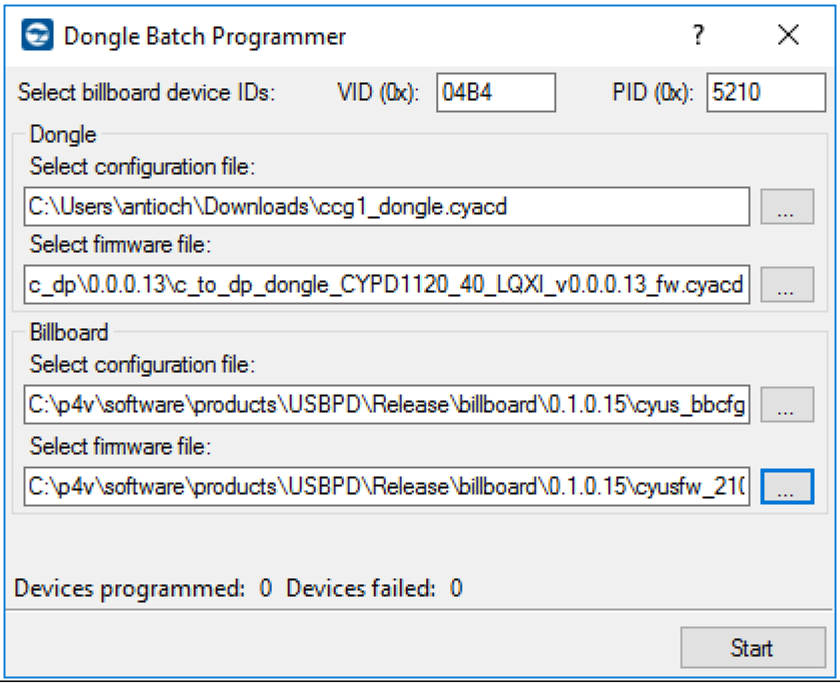
A special batch programming mode is supported for Type-C to Display dongle applications as well. The Dongle Batch Programmer dialog allows users to program multiple Type-C to Display dongles without repeated actions on the UI.

The files to be programmed include:

- Configuration file for the CCGx AMA controller (generated through the utility).
- Firmware binary for the CCGx AMA controller.
- Configuration file for the USB-Serial Billboard controller (generated through the utility)
- Firmware binary for the USB-Serial Billboard controller.

You must select at least one of these files for programming.

The programming is done through the USB-Serial Billboard controller. The Billboard controller is expected to support a HID USB device interface through which the programming request is initiated. The USB device IDs (Vendor and Product ID) associated with the HID device to be used for programming should be specified.



Select billboard device IDs: VID (0x): 04B4 PID (0x): 5210

Dongle

Select configuration file: C:\Users\antioch\Downloads\ccg1_dongle.cyacd

Select firmware file: c_dp\0.0.0.13\c_to_dp_dongle_CYPD1120_40_LQXI_v0.0.0.13_fw.cyacd

Billboard

Select configuration file: C:\p4v\software\products\USBPD\Release\billboard\0.1.0.15\cyus_bbcfg

Select firmware file: C:\p4v\software\products\USBPD\Release\billboard\0.1.0.15\cyusfw_21(

Devices programmed: 0 Devices failed: 0

Figure 82 Dongle batch programming dialog

Once the programming starts, the utility will perform the following operations:

- Wait for a Billboard device with the specified device IDs to be connected.
- Switch the Billboard device to programming mode.
- Program the CCGx dongle controller if required.
- Program the Billboard controller if required.
- Request you to plug in the next dongle device.

The utility will wait for a maximum of one minute for each dongle to be plugged in. If a dongle is not detected within this period, the batch programming will be aborted. You can restart the operation later.

4.11 Manufacturing test kit option

The EZ-PD™ configuration utility supports MTK Option, a special mode to configure and test CMG1 EMCA devices.

The MTK consists of the MTK UI, MTK firmware, and MTK hardware.

The MTK UI will be invoked from the EZ-PD™ configuration utility. The MTK Firmware will be available as part of the EZ-PD™ configuration utility installation package.

The MTK UI supports configuring and testing of up to 10 CMG1 cables in parallel. This option is designed to be used in factories during the manufacturing cycle. **Figure 83** shows the setup for configuring and testing 10 cables.

The MTK hardware consists of the CY4532 EZ-PD™ CCG3PA EVK Power board (ensure the CY4532 EZ-PD™ CCG3PA EVK Main board is detached from the power board). For more details, see this **link**.

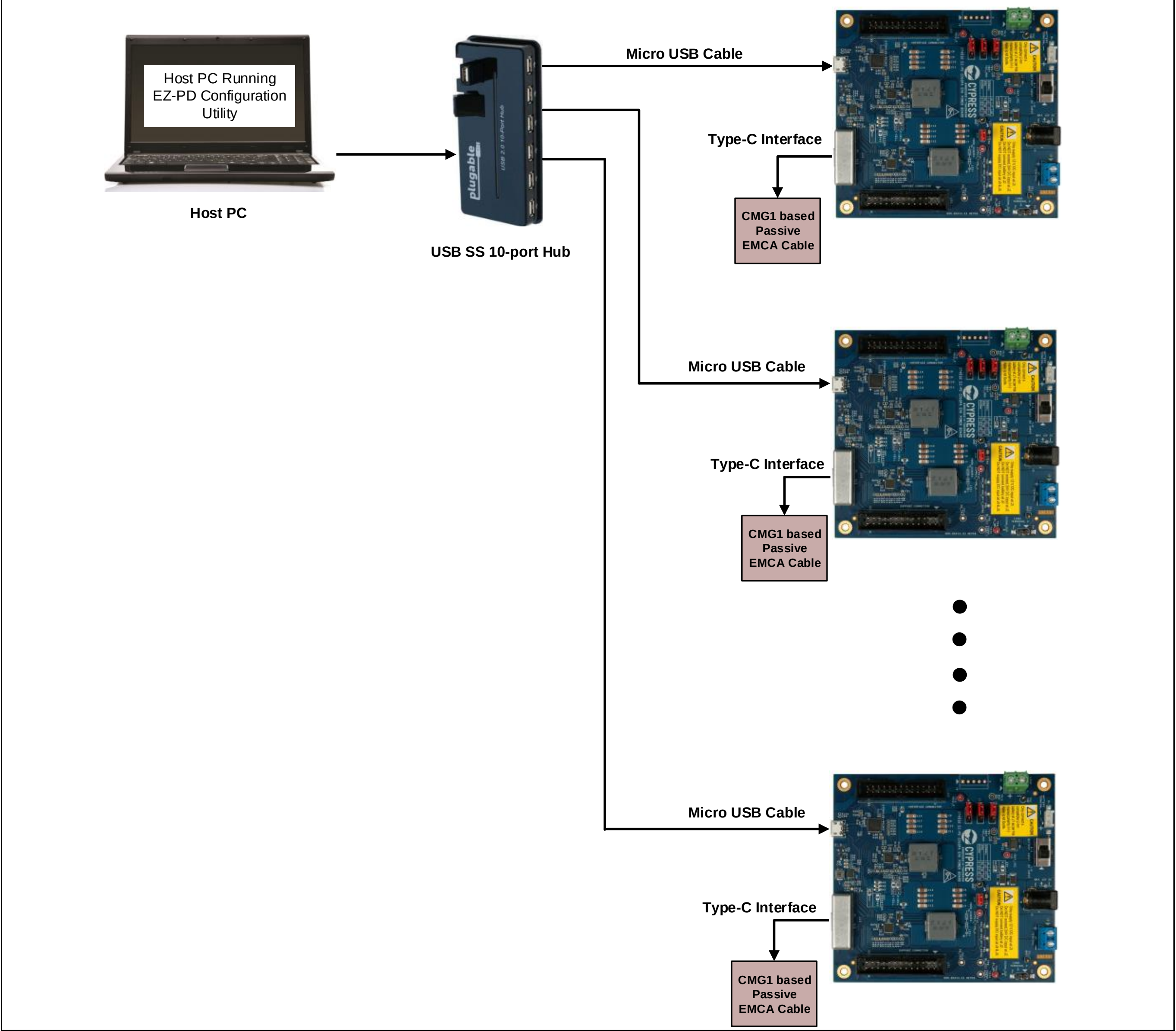


Figure 83 CMG1 configure/test setup

The following operations are required to configure and test the CMG1 EMCA devices with the MTK:

- Download MTK firmware
- Configure MTK Tester ID
- Create CMG1 configuration
- MTK operation

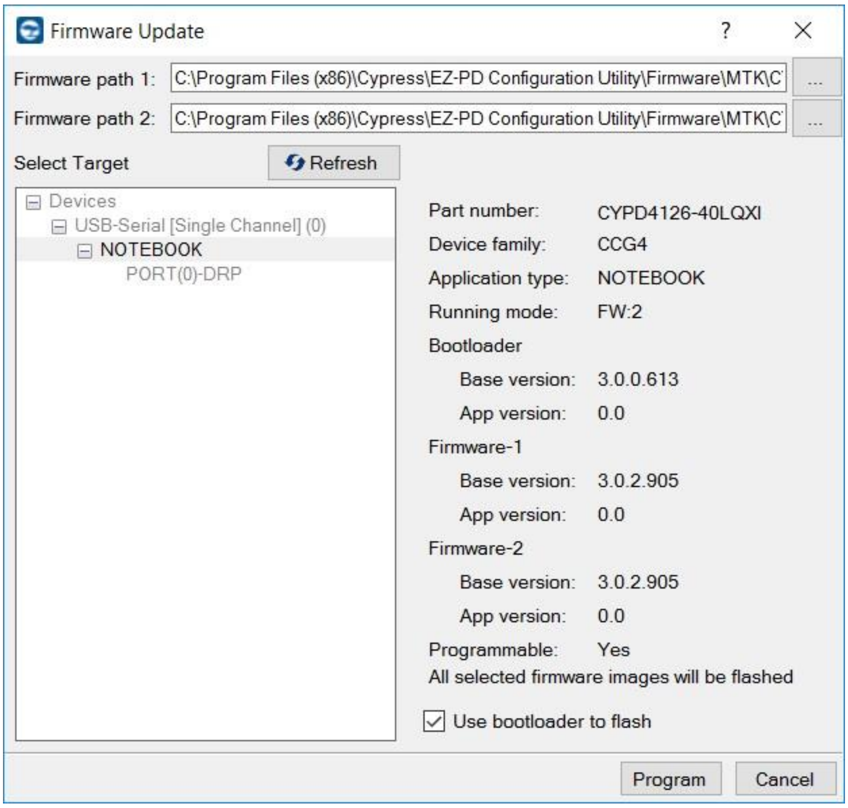
4.11.1 Download MTK firmware

See **Programming manufacturing test kit hardware** for the hardware setup to download the firmware The MTK firmware is available in the *Firmware\MTK* folder in the EZ-PD™ configuration utility installation.

Download both the firmware images *CYPD4126-40LQXI_mtk_1.cyacd* and *CYPD4126-40LQXI_mtk_2.cyacd* via the EZ-PD™ configuration utility.

To update the firmware:

1. Select **Firmware Update** and load appropriate firmware path for the **Firmware path 1** and **Firmware path 2**.
2. Select **Use bootloader to flash**, as shown in **Figure 84**, and program the CY4532 power board.



Firmware path 1: C:\Program Files (x86)\Cypress\EZ-PD Configuration Utility\Firmware\MTK\C

Firmware path 2: C:\Program Files (x86)\Cypress\EZ-PD Configuration Utility\Firmware\MTK\C

Select Target

Part number: CYPD4126-40LQXI

Device family: CCG4

Application type: NOTEBOOK

Running mode: FW:2

Bootloader

Base version: 3.0.0.613

App version: 0.0

Firmware-1

Base version: 3.0.2.905

App version: 0.0

Firmware-2

Base version: 3.0.2.905

App version: 0.0

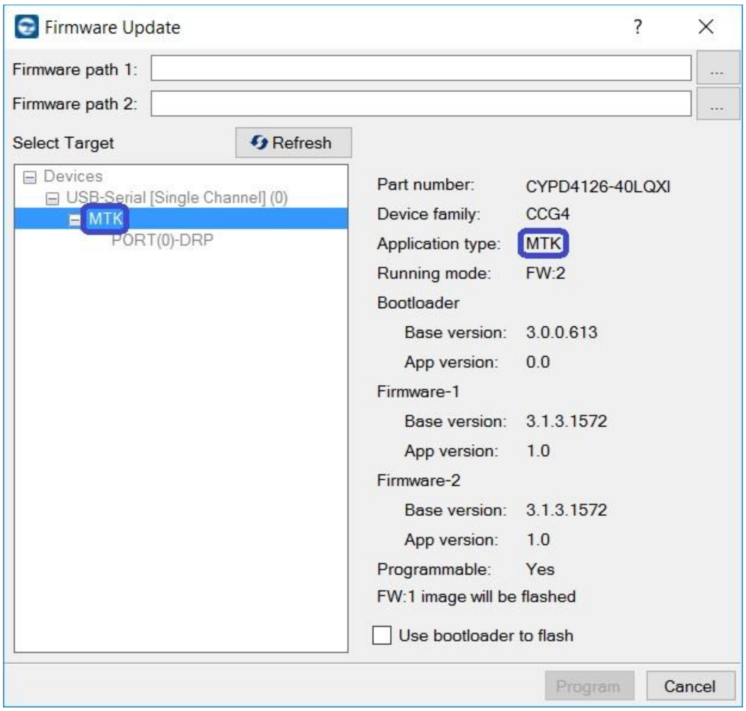
Programmable: Yes

All selected firmware images will be flashed

☑ Use bootloader to flash

Figure 84 MTK firmware update

Once the firmware images are updated successfully, the Application Type in Firmware update window will change to MTK as shown in **Figure 85**.



Firmware Update

Firmware path 1:

Firmware path 2:

Select Target

Part number: CYPD4126-40LQXI

Device family: CCG4

Application type: MTK

Running mode: FW:2

Bootloader

Base version: 3.0.0.613

App version: 0.0

Firmware-1

Base version: 3.1.3.1572

App version: 1.0

Firmware-2

Base version: 3.1.3.1572

App version: 1.0

Programmable: Yes

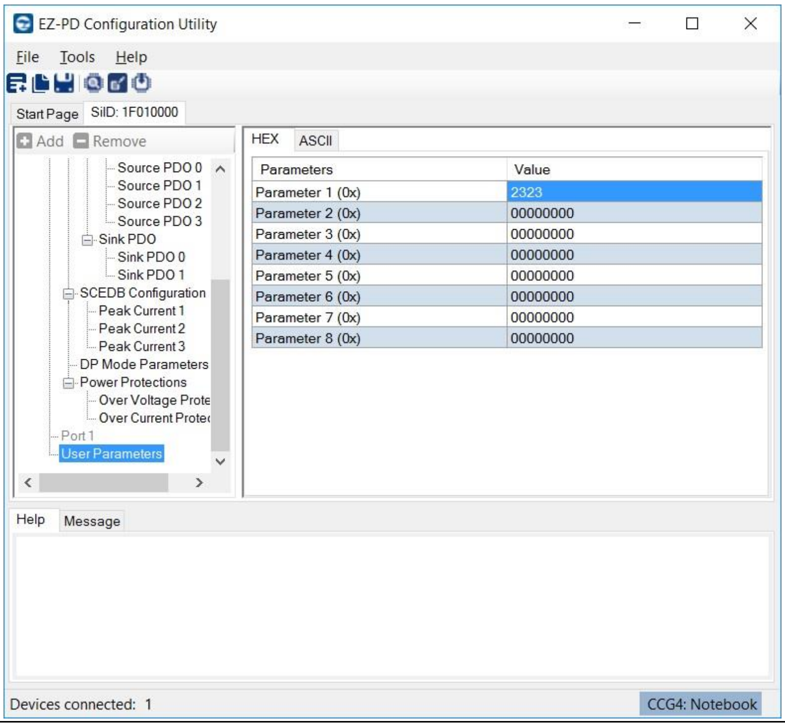
FW:1 image will be flashed

□ Use bootloader to flash

Figure 85 MTK firmware update success

4.11.2 Configure MTK tester ID

To uniquely identify the MTK hardware, you can provide a tester ID by configuring the user parameter of the MTK hardware configuration. To configure the tester ID, first read the configuration information from the MTK hardware. Follow the menu path **File** > **Read from Device** (see **Figure 74**). From the User Parameters section, update **Parameter 1** with the 2-byte tester ID as shown in **Figure 86**.



EZ-PD Configuration Utility

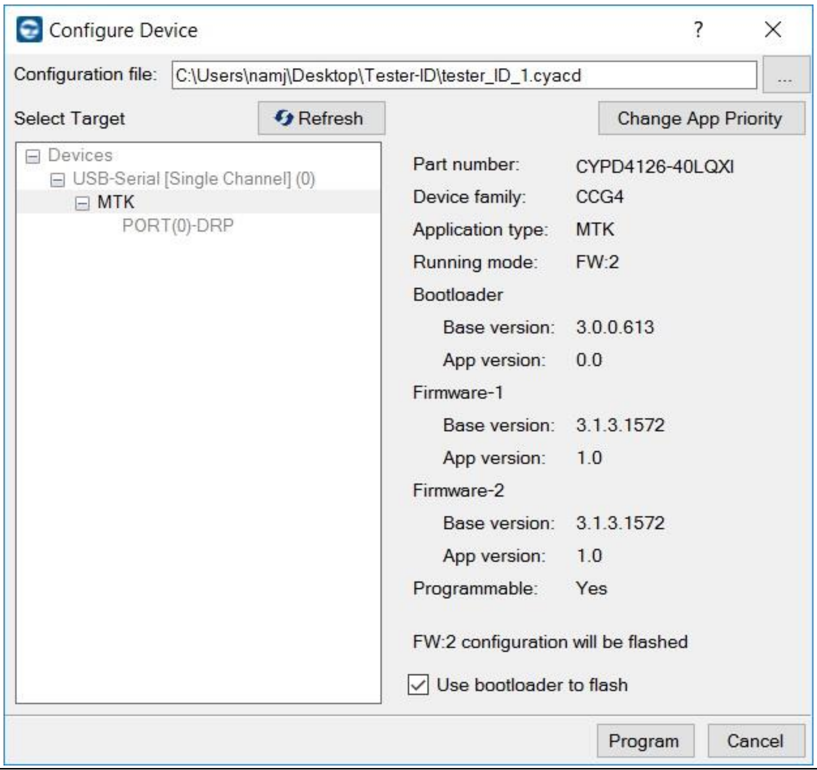
Parameters	Value
Parameter 1 (0x)	2323
Parameter 2 (0x)	00000000
Parameter 3 (0x)	00000000
Parameter 4 (0x)	00000000
Parameter 5 (0x)	00000000
Parameter 6 (0x)	00000000
Parameter 7 (0x)	00000000
Parameter 8 (0x)	00000000

Devices connected: 1

CCG4: Notebook

Figure 86 Configure MTK tester ID

Save the configuration and program the saved configuration as shown in the **Figure 87**. If the programming is successful, the programmed tester ID will be displayed in the Tester ID column of the MTK display window (see **Figure 90**).


Configure Device
?
Configuration file: C:\Users\namj\Desktop\Tester-ID\tester_ID_1.cyacd
...
Select Target
Refresh
Change App Priority
Devices
USB-Serial [Single Channel] (0)
MTK
PORT(0)-DRP

Part number: CYPD4126-40LQXI
Device family: CCG4
Application type: MTK
Running mode: FW:2
Bootloader
Base version: 3.0.0.613
App version: 0.0
Firmware-1
Base version: 3.1.3.1572
App version: 1.0
Firmware-2
Base version: 3.1.3.1572
App version: 1.0
Programmable: Yes

FW:2 configuration will be flashed

☑ Use bootloader to flash

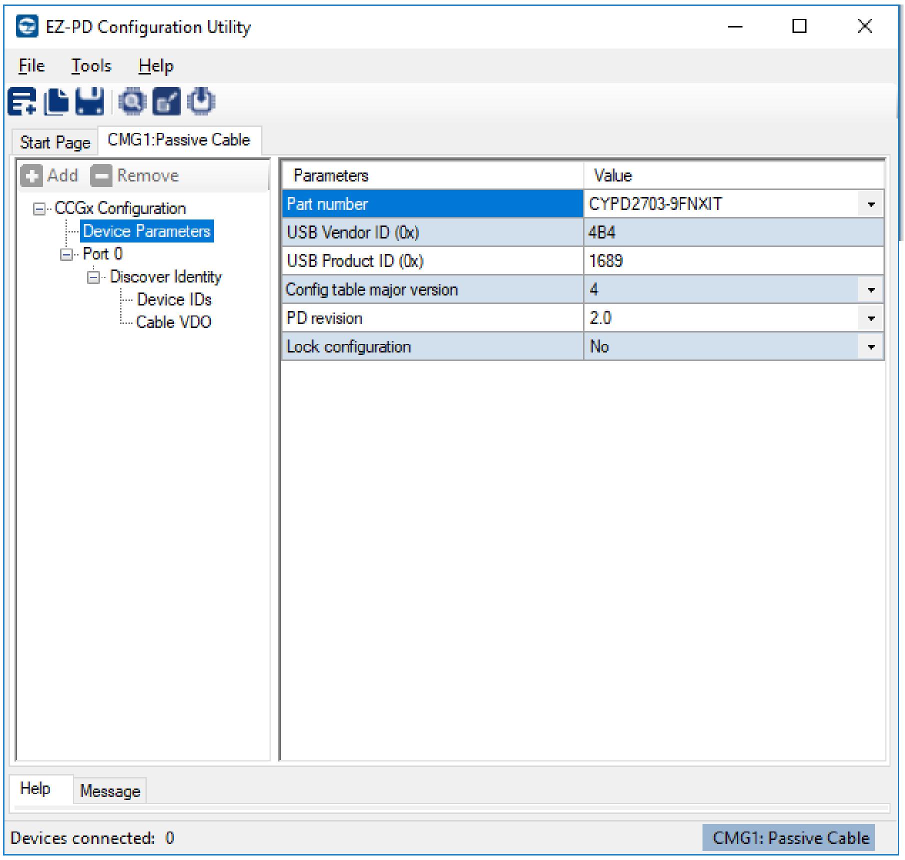
Figure 87 Program tester ID

4.11.3 Create CMG1 configuration

Create a new configuration project with **CMG1** as the **Device Type** (see **Creating configuration** for more details).

You can select a group to view the parameters and the corresponding values on the right pane. To modify the parameters, you can either select the values from the drop-down list or enter the values in the textbox. The utility performs validity checks on the values entered in the textbox.

The CMG1 Configuration options are shown in **Figure 88**.



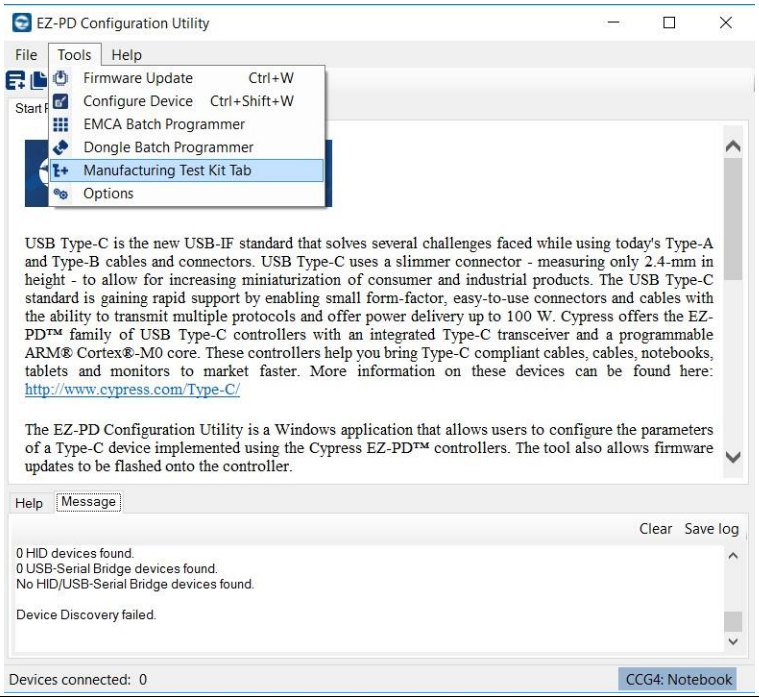
Parameters	Value
Part number	CYPD2703-9FNXIT
USB Vendor ID (0x)	4B4
USB Product ID (0x)	1689
Config table major version	4
PD revision	2.0
Lock configuration	No

Figure 88 CMG1 configuration parameters

Note: Selecting ***Yes*** *for the* ***Lock configuration*** *option will treat the CMG1 device as a one-time configurable part.*

4.11.4 MTK operation

To launch the MTK UI from the EZ-PD™ configuration utility, follow the menu path **Tools** > **Manufacturing Test Kit Tab**.



USB Type-C is the new USB-IF standard that solves several challenges faced while using today's Type-A and Type-B cables and connectors. USB Type-C uses a slimmer connector - measuring only 2.4-mm in height - to allow for increasing miniaturization of consumer and industrial products. The USB Type-C standard is gaining rapid support by enabling small form-factor, easy-to-use connectors and cables with the ability to transmit multiple protocols and offer power delivery up to 100 W. Cypress offers the EZ-PD™ family of USB Type-C controllers with an integrated Type-C transceiver and a programmable ARM® Cortex®-M0 core. These controllers help you bring Type-C compliant cables, cables, notebooks, tablets and monitors to market faster. More information on these devices can be found here: http://www.cypress.com/Type-C/

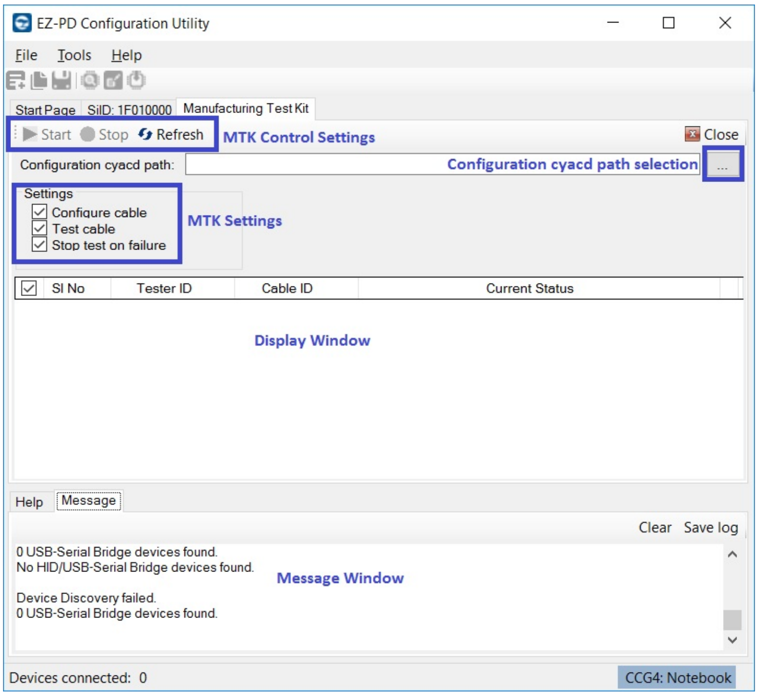
The EZ-PD Configuration Utility is a Windows application that allows users to configure the parameters of a Type-C device implemented using the Cypress EZ-PD™ controllers. The tool also allows firmware updates to be flashed onto the controller.

```
0 HID devices found.
0 USB-Serial Bridge devices found.
No HID/USB-Serial Bridge devices found.

Device Discovery failed.
```

Figure 89 Manufacturing test kit tab option

The Manufacturing Test Kit GUI is displayed as shown in the **Figure 90**. The MTK UI has a Display window, Message window, and MTK control and settings options.



Settings
☑ Configure cable
☑ Test cable
☑ Stop test on failure

```
0 USB-Serial Bridge devices found.
No HID/USB-Serial Bridge devices found.

Device Discovery failed.
0 USB-Serial Bridge devices found.
```

Figure 90 MTK GUI layout

The MTK control options are listed in **Table 3**.

Table 3 MTK control options

MTK control	Description
Start	The MTK **Start** option will be enabled only when the correct configuration *cyacd* is loaded. Clicking **Start** will initiate the detecting and testing of the CMG1 devices.
Stop	The MTK **Stop** option will complete the current test in progress and stop testing and detecting CMG1 devices.
Refresh	The **Refresh** option will detect the MTK hardware connected to the PC.

The MTK Settings are listed in **Table 4**.

Table 4 MTK settings

MTK control	Description
Configure cable	Configures the CMG1 device based on the selected configuration *cyacd* file.
Test cable	Sends the Discover Identity PD command and verifies the response with the loaded configuration *cyacd* file to declare Pass/Fail.
Stop test on failure	When a failure is detected by one of the MTKs, the current testing on the rest of detected CMG1 devices will complete and the MTK will stop detecting and testing CMG1 devices. If this option is not checked, then only the MTK that has detected a failure will stop detecting the CMG1 device. The remaining MTKs will continue to detect and test CMG1 devices.

The various fields of the display window are described in **Table 5**.

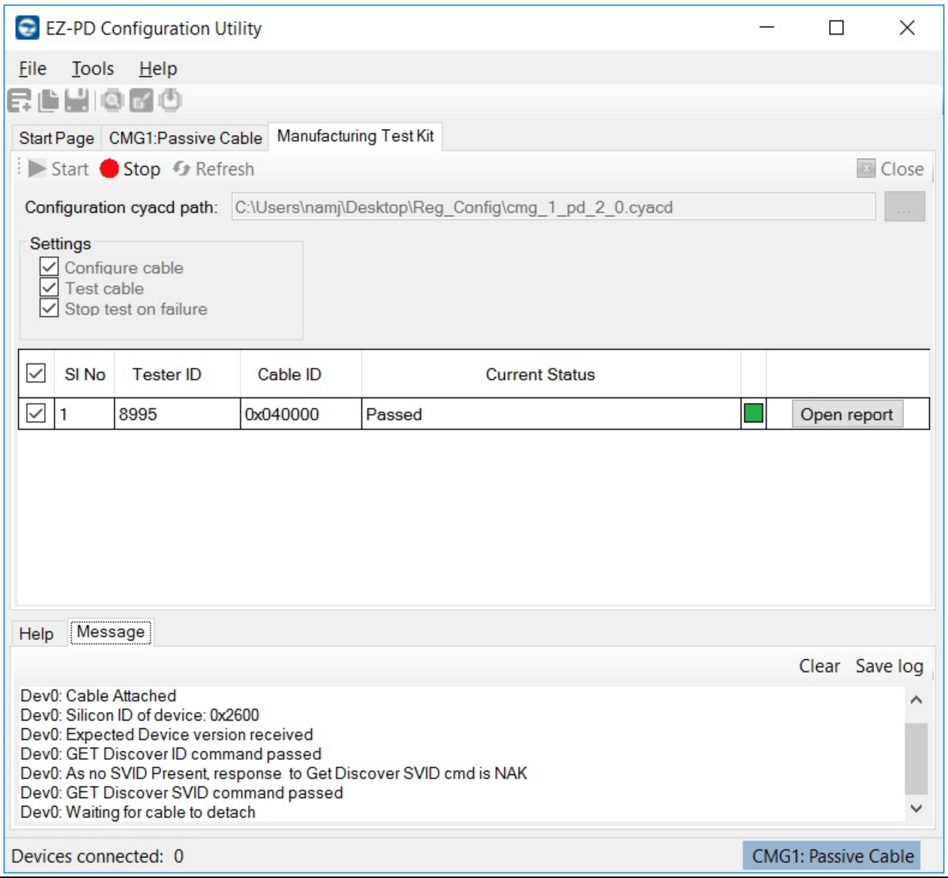
Table 5 Fields in display window

MTK control	Description
Sl No	Serial number of the connected MTK hardware
Tester ID	Unique tester ID of the MTK hardware
Cable ID	Unique cable ID of the connected CMG1 device
Current Status	The Current Status can be one of the following status: • Cable detached – Yellow LED (CMG1 device not attached) • Cable attached – Yellow LED (CMG1 device attached) • Configuring – Yellow LED (CMG1 device configuration in progress) • Testing – Yellow LED (CMG1 device testing in progress) • Passed – Green LED (Test passed) • Failed – Red LED (Test failed)

The logs/reports are maintained for each MTK. Select **Open report** to open the report. Select Save log to save a consolidated log.

The MKT UI can detect a maximum of 10 MTK hardware. Each MTK hardware should be programmed and configured with a unique Tester ID.

The CMG1 devices will be configured from the configuration the *cyacd* file that is provided.



Configuration cyacd path: C:\Users\namj\Desktop\Reg_Config\cmg_1_pd_2_0.cyacd

Settings

- ☑ Configure cable
- ☑ Test cable
- ☑ Stop test on failure

☑	Sl No	Tester ID	Cable ID	Current Status		
☑	1	8995	0x040000	Passed	🟩	Open report

```
Dev0: Cable Attached
Dev0: Silicon ID of device: 0x2600
Dev0: Expected Device version received
Dev0: GET Discover ID command passed
Dev0: As no SVID Present, response  to Get Discover SVID cmd is NAK
Dev0: GET Discover SVID command passed
Dev0: Waiting for cable to detach
```

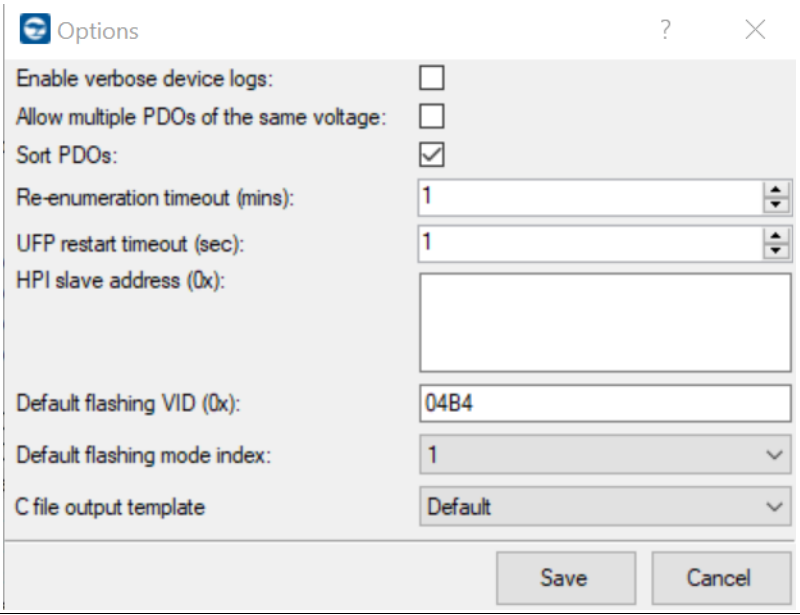
Devices connected: 0

CMG1: Passive Cable

Figure 91 MTK operation

4.12 Utility options

To change the options, follow the menu path **Tools** > **Options**. Figure 92 shows the available options.



Enable verbose device logs: □

Allow multiple PDOs of the same voltage: □

Sort PDOs: ☑

Re-enumeration timeout (mins): 1

UFP restart timeout (sec): 1

HPI slave address (0x):

Default flashing VID (0x): 04B4

Default flashing mode index: 1

C file output template Default

Figure 92 Options menu

4.12.1 Enable device verbose logs

Select this option to view detailed logs. The logs displayed in the **Messages** tab can be functional logs required to identify status, or verbose logs that can be used for debugging.

4.12.2 Allow multiple PDOs of the same voltage

Select this option to create a configuration with multiple PDOs of the same voltage.

4.12.3 Sort PDOs

Select this option to sort Source and Sink PDOs.

4.12.4 Re-enumeration timeout

Programming an USB-Serial based Billboard device involves sending a command, which causes the device to disconnect and reconnect in manufacturing mode. When this mode switch happens for the first time on a particular USB host port, there could be delays in binding the device to the Cypress USB driver. The **Re-enumeration timeout** parameter specifies the maximum amount of time that the utility will wait for detecting the billboard device in USB Serial Manufacturing mode. If this time is shorter than the delay in driver binding, the flashing operation in billboard device will fail.

4.12.5 UFP restart timeout

Programming UFP device over CC interface involves sending reset device command, which causes UFP device and flashing controller to restart. Some UFP devices take longer to restart, which leads to failure of power contract between UFP and flashing controller. The **UFP restart timeout** parameter specifies the amount of time that the utility will wait to get UFP device ready after restart.

4.12.6 HPI slave address

The CCGx devices are connected as I2C slave devices to the USB-Serial Bridge used as a bridge for programming. The utility looks for and detects CCGx devices that are configured with the default I2C slave addresses of 0x08, 0x12, and 0x44. If the target design uses I2C slave addresses that is not on this list, you can enter the additional addresses in the **HPI slave address** textbox. You can add any number of I2C addresses, separated by spaces.

4.12.7 Default flashing VID and index

CCGx devices in EMCA and power adapter applications are programmed through the CC interface. The flash programming commands in these applications are sent in the form of Unstructured Vendor Data Messages (UVDM) commands. These UVDMs will only be supported by CCGx when it has entered a dedicated flashing alternate mode. CCGx firmware uses the Cypress VID (0x04b4) and alternate mode index 1 as the flashing alternate mode by default. If the target design uses a different VID or mode index, specify the corresponding values using the **Default flashing VID** and **Default flashing mode index** settings.

4.12.8 C File output template

This setting can be used for selecting a template for generating the output *.c* configuration file. The *.c* configuration file generated while saving configuration can be in either of the following formats.

- **Default format:** In the default format, the *.c* configuration file contains the configuration table section along with the following function definitions.

```
const pd_config_t * get_pd_config(void)
{
    return (pd_config_t *)&gl_config_table;
}
```

  and

```
const pd_port_config_t * get_pd_port_config(uint8_t port)
{
    return (pd_port_config_t *)&(((pd_config_t *)gl_config_table)-
>port_conf[port]);
}
```

- **CCGx Auto application format:** In the auto application format, the *.c* configuration file does not contain the above function definitions. It contains only the configuration table section.

Use the **CCGx Auto** template only for auto applications. For all remaining CCGx applications, select the **Default** template.

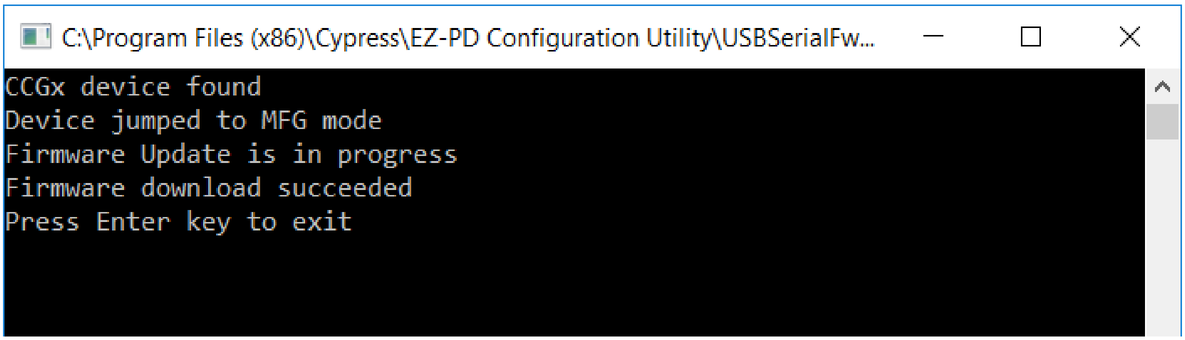
4.13 Firmware download for USB serial

A separate firmware update tool *CyUSBSerial_FwUpdate_Tool.exe* is available in *C:\Program Files (x86)\Cypress\EZ-PD Configuration Utility\USBSerialFwDownloadTool*. This tool is used to update the firmware for the USB Serial. The improved USB serial firmware is intended for the CY4532 Kit and will reduce the firmware download time over the CC line.

Follow these steps to download the improved USB serial firmware:

1. Power the CY4532 kit and connect to the Windows PC.
2. In the Device Manager, check if the CY4532 kit is enumerated and bound to the Cypress driver.
3. Double-click the *CyUSBSerial_FwUpdate_Tool.exe* tool.

After the firmware is successfully downloaded, the message, shown in **Figure 93**, will be displayed in the command window.


C:\Program Files (x86)\Cypress\EZ-PD Configuration Utility\USBSerialFw...

```
CCGx device found
Device jumped to MFG mode
Firmware Update is in progress
Firmware download succeeded
Press Enter key to exit
```

Figure 93 USB serial firmware download success

4.14 Help

The EZ-PD™ configuration utility provides the following help options:

- Windows Help document, which describes the various UI screens and their usage. You can access this document from the Help Topics menu.
- A Context Sensitive help button (?), which is part of every UI window. Clicking this button will display documentation for that particular UI.
- Small snippets of help texts are displayed in the **Help** tab at the bottom of the dialog, and tooltips are displayed when you hover the mouse over the GUI components.

Revision history

Version	Date	Description
**	2017-11-23	Initial release
*A	2018-04-02	Added Section 3.2.6 for Programming MTK Hardware Added Section 4.11 for CMG1 Manufacturing Test Kit operation.
*B	2018-07-16	Added Section 4.13 for USB Serial Firmware Download.
*C	2018-11-29	Updated section 4.1 to add Thunderbolt Host Configuration and reference to CCGx Host/Power SDK User Guides. Added section 4.1.14 to capture Thunderbolt Host Configuration node
*D	2019-08-05	Updated Sections 1.1, 2.1.1, 4, 4.1.2, 4.6.1 for adding support for PAG1S/CCG5C/CCG6/ACG1F family. Figures updated: 2-1, 2-2, 2-3, 4-2, 4-3, 4-4, 4-10, 4-19, 4-24, 4-26, 4-35, 4-57
*E	2019-09-26	Added section 4.14 Blackbox Utility for CCG3PA
*F	2020-03-26	Removed Section 4.14 Updated Sections 1.1, 2.1.1, 4, 4.1.15 for adding support for CCG6DF and CCG6SF family. Added sections 4.1.9, 4.1.16, 4.1.17, 4.1.18 and 4.1.19 to capture SKEDB Configurations, Base Alternate Mode Configuration, Custom Alternate Mode Configuration, Custom Host Configuration and ICL/TGL Configurations respectively. Figure updated: 4-36 Update Disclaimer on Page 2
*G	2020-06-19	Added Sections Extensions for hybrid architecture applications, and Extensions for hybrid architecture applications for hybrid architecture applications support Added Figure 4-55, Figure 4-59, and Figure 4-62 Updated Figure 3-1 Added note in Section Programming over CC channel: ‘The CC Firmware Update feature is not intended for field firmware update’
*H	2020-09-22	Updated “Introduction” chapter. Updated “EZ-PD™ Device Family”: Updated description. Updated “EZ-PD Configuration Utility Overview” chapter: Updated “Usage Flow”: Updated “Create Configuration”: Updated table. Updated “GUI Usage” chapter: Updated description. Updated “USB-PD Device Configuration”: Updated description. Updated “Power Settings”: Updated figure “Power Settings Parameter Configuration”. Added “Auto Configuration”. Updated “Thunderbolt Host Configurations”:

Revision history

Version	Date	Description
		Updated description.
*I	2020-12-03	Moved to Infineon Template Updated **Figure 16**, **Figure 20**, **Figure 21**, **Figure 23**, and **Figure 51** Updated **USB-PD device configuration** Added **BCR configuration**
*J	2021-03-11	Updated **Figure 29**, **Figure 34**, **Figure 45**, **Figure 49**, and **Figure 92**. Updated Section **4.6.1**. Added **Sort PDOs**.
*K	2021-05-24	Updated **Figure 1**, **Figure 16**, **Figure 19**, **Figure 21**, **Figure 46**, **Figure 47**, and **Figure 49** Added section **4.1.21**

Trademarks

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

Edition 2021-05-24

Published by

Infineon Technologies AG

81726 Munich, Germany

Do you have a question about this document?

Go to www.cypress.com/support

Document reference

002-22139 Rev. *K

IMPORTANT NOTICE

The information given in this document shall in no event be regarded as a guarantee of conditions or characteristics (“Beschaffenheitsgarantie”) .

With respect to any examples, hints or any typical values stated herein and/or any information regarding the application of the product, Infineon Technologies hereby disclaims any and all warranties and liabilities of any kind, including without limitation warranties of non-infringement of intellectual property rights of any third party.

In addition, any information given in this document is subject to customer’s compliance with its obligations stated in this document and any applicable legal requirements, norms and standards concerning customer’s products and any use of the product of Infineon Technologies in customer’s applications.

The data contained in this document is exclusively intended for technically trained staff. It is the responsibility of customer’s technical departments to evaluate the suitability of the product for the intended application and the completeness of the product information given in this document with respect to such application.

For further information on the product, technology, delivery terms and conditions and prices please contact your nearest Infineon Technologies office (**www.infineon.com**).

WARNINGS

Due to technical requirements products may contain dangerous substances. For information on the types in question please contact your nearest Infineon Technologies office.

Except as otherwise explicitly approved by Infineon Technologies in a written document signed by authorized representatives of Infineon Technologies, Infineon Technologies’ products may not be used in any applications where a failure of the product or any consequences of the use thereof can reasonably be expected to result in personal injury.