



KitProg3 User Guide

Document Number: 002-24616 Rev. *D

Cypress Semiconductor
198 Champion Court
San Jose, CA 95134-1709
www.cypress.com

© Cypress Semiconductor Corporation, 2018-2019. This document is the property of Cypress Semiconductor Corporation and its subsidiaries (“Cypress”). This document, including any software or firmware included or referenced in this document (“Software”), is owned by Cypress under the intellectual property laws and treaties of the United States and other countries worldwide. Cypress reserves all rights under such laws and treaties and does not, except as specifically stated in this paragraph, grant any license under its patents, copyrights, trademarks, or other intellectual property rights. If the Software is not accompanied by a license agreement and you do not otherwise have a written agreement with Cypress governing the use of the Software, then Cypress hereby grants you a personal, non-exclusive, nontransferable license (without the right to sublicense) (1) under its copyright rights in the Software (a) for Software provided in source code form, to modify and reproduce the Software solely for use with Cypress hardware products, only internally within your organization, and (b) to distribute the Software in binary code form externally to end users (either directly or indirectly through resellers and distributors), solely for use on Cypress hardware product units, and (2) under those claims of Cypress’s patents that are infringed by the Software (as provided by Cypress, unmodified) to make, use, distribute, and import the Software solely for use with Cypress hardware products. Any other use, reproduction, modification, translation, or compilation of the Software is prohibited.

TO THE EXTENT PERMITTED BY APPLICABLE LAW, CYPRESS MAKES NO WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, WITH REGARD TO THIS DOCUMENT OR ANY SOFTWARE OR ACCOMPANYING HARDWARE, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE. No computing device can be absolutely secure. Therefore, despite security measures implemented in Cypress hardware or software products, Cypress shall have no liability arising out of any security breach, such as unauthorized access to or use of a Cypress product. CYPRESS DOES NOT REPRESENT, WARRANT, OR GUARANTEE THAT CYPRESS PRODUCTS, OR SYSTEMS CREATED USING CYPRESS PRODUCTS, WILL BE FREE FROM CORRUPTION, ATTACK, VIRUSES, INTERFERENCE, HACKING, DATA LOSS OR THEFT, OR OTHER SECURITY INTRUSION (collectively, “Security Breach”). Cypress disclaims any liability relating to any Security Breach, and you shall and hereby do release Cypress from any claim, damage, or other liability arising from any Security Breach. In addition, the products described in these materials may contain design defects or errors known as errata which may cause the product to deviate from published specifications. To the extent permitted by applicable law, Cypress reserves the right to make changes to this document without further notice. Cypress does not assume any liability arising out of the application or use of any product or circuit described in this document. Any information provided in this document, including any sample design information or programming code, is provided only for reference purposes. It is the responsibility of the user of this document to properly design, program, and test the functionality and safety of any application made of this information and any resulting product. “High-Risk Device” means any device or system whose failure could cause personal injury, death, or property damage. Examples of High-Risk Devices are weapons, nuclear installations, surgical implants, and other medical devices. “Critical Component” means any component of a High-Risk Device whose failure to perform can be reasonably expected to cause, directly or indirectly, the failure of the High-Risk Device, or to affect its safety or effectiveness. Cypress is not liable, in whole or in part, and you shall and hereby do release Cypress from any claim, damage, or other liability arising from any use of a Cypress product as a Critical Component in a High-Risk Device. You shall indemnify and hold Cypress, its directors, officers, employees, agents, affiliates, distributors, and assigns harmless from and against all claims, costs, damages, and expenses, arising out of any claim, including claims for product liability, personal injury or death, or property damage arising from any use of a Cypress product as a Critical Component in a High-Risk Device. Cypress products are not intended or authorized for use as a Critical Component in any High-Risk Device except to the limited extent that (i) Cypress’s published data sheet for the product explicitly states Cypress has qualified the product for use in a specific High-Risk Device, or (ii) Cypress has given you advance written authorization to use the product as a Critical Component in the specific High-Risk Device and you have signed a separate indemnification agreement.

Cypress, the Cypress logo, Spansion, the Spansion logo, and combinations thereof, WICED, PSoC, CapSense, EZ-USB, F-RAM, and Traveo are trademarks or registered trademarks of Cypress in the United States and other countries. For a more complete list of Cypress trademarks, visit cypress.com. Other names and brands may be claimed as property of their respective owners.

Contents



1. Introduction.....	4
1.1 What Is In this Manual	5
1.2 KitProg3 Tools Support and Compatibility	5
2. Installing and Using KitProg3.....	6
2.1 Installing KitProg3.....	6
2.2 Using KitProg3.....	6
3. KitProg3 Design.....	10
3.1 Supported Kits	10
3.2 Operating Speeds.....	10
4. DAPLink Mode	11
4.1 Supported Kits	11
4.2 Mbed Ecosystem	11
4.3 Features	11
4.4 How To	12
5. KitProg3 vs. KitProg2.....	13
5.1 Feature Comparison.....	13
5.2 Upgrading to KitProg3	13
5.3 Downgrading to KitProg2.....	14
6. Troubleshooting	15
Revision History.....	16

1. Introduction



KitProg3 is the Cypress low-level communication firmware for programming and debugging. It provides communication between a programming tool (such as Cypress Programmer or PSoC® Programmer™) and a target, such as a PSoC 6 MCU device. KitProg3 supports a variety of Cypress development kits. It is also the communication firmware found in the MiniProg4 debug probe.

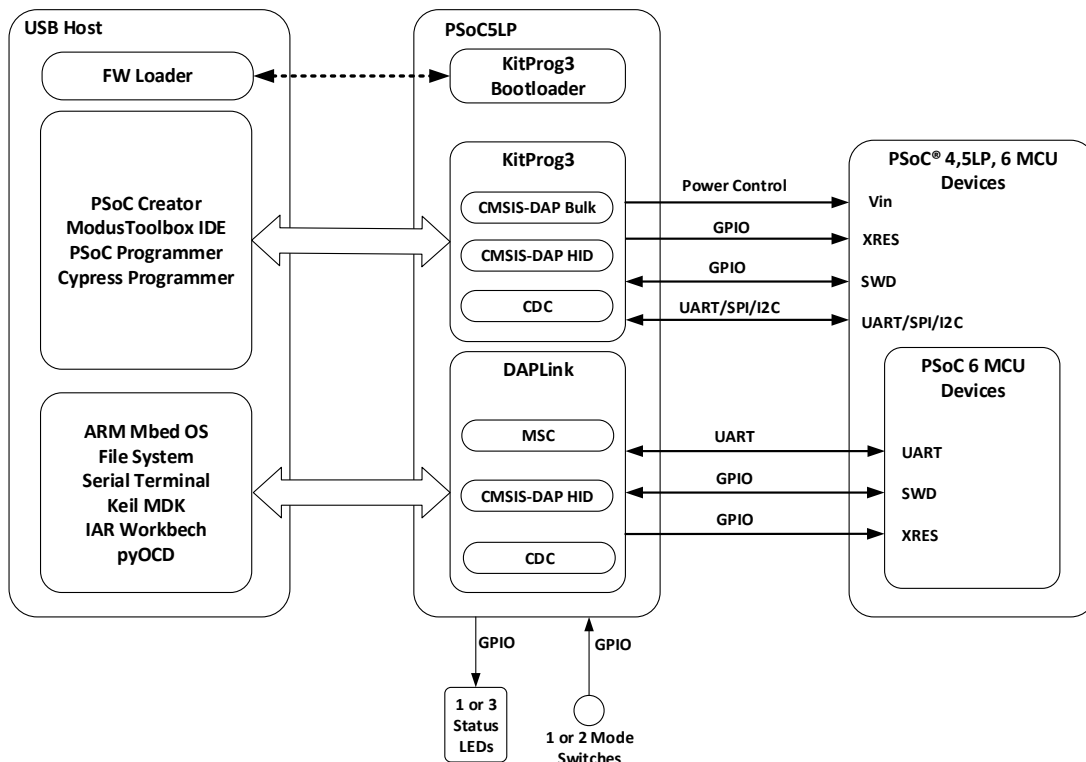
Cypress development kits have KitProg firmware installed to provide the necessary communication between the host and target. As a result, you plug the kit into your host computer – programming and debugging just work.

KitProg3 uses the industry-standard Serial Wire Debug (SWD) protocol. It uses CMSIS-DAP V2.0.0 and V1.2.0 as the Bulk and HID endpoints transport mechanisms. CMSIS-DAP is also an industry standard. KitProg3 implements USB Bulk endpoints for faster communication. It also supports HID endpoints for use cases that require them, but communication is slower. Out of the box, KitProg3 uses Bulk endpoints.

KitProg3 also supports bridging: USB-UART, USB-I2C, and USB-SPI.

The KitProg3 package also includes the Arm® Mbed™ DAPLink that enables programming and debugging applications (IoT) for Arm Cortex® CPUs (PSoC 6 MCU devices only). DAPLink is platform-independent and provides Drag-and-Drop programming via a Mass Storage Controller (MSC), CMSIS-DAP debugging (HID endpoints), and a virtual serial port via USB Communications Device Class (CDC). You can switch between KitProg3 and DAPLink with a simple push of a button.

Figure 1-1. KitProg3 High-Level Architecture



1.1 What Is In this Manual

This user guide provides comprehensive information about KitProg3 in PSoC development kits:

[Section 2: Installing and Using KitProg3](#) – provides all the information you need to get up and running for the common use cases

[Section 3: KitProg3 Design](#) – includes full details about KitProg3 User interface, mode switching and status LEDs behavior.

[Section 4: DAPLink Mode](#) – includes details on how to upgrade kit firmware to KitProg3 with DAPLink, how to switch to DAPLink mode, and useful references to Arm resources

[Section 5: KitProg3 vs KitProg2](#) – includes details about how to tell what's installed, the differences, and how to upgrade a kit to KitProg3.

In case of any issues, see the [Troubleshooting](#) section.

1.2 KitProg3 Tools Support and Compatibility

KitProg3 is supported by the following tool combinations:

- User mode
 - [ModusToolbox IDE](#) and [Cypress Programmer](#)
 - [PSoC Creator](#) and [PSoC Programmer](#) (v 3.28)
- DAPLink mode
 - [ModusToolbox](#)
 - [Mbed CLI](#)
 - [pyOCD](#)
 - [uVision](#)

Table 1-1. KitProg Compatibility

KitProg	IDE	Programmer	Bridging Tools
KitProg3	ModusToolbox IDE PSoC Creator™	Cypress Programmer PSoC Programmer	Bridge Control Panel (PSoC Programmer)* CapSense® Tuner (PSoC Creator and ModusToolbox IDE)
KitProg2	PSoC Creator	PSoC Programmer	

* Note that the Bridge Control Panel is not supported by Cypress Programmer or ModusToolbox IDE.

Table 1-2. KitProg Modes

Mode	USB devices	Features
User mode	CMSIS-DAP HID CMSIS-DAP Bulk CDC UART	CMSIS-DAP Programming/Debugging I2C/SPI/UART Bridging Voltage control
DAPLink mode	Mass Storage Device CDC UART CMSIS-DAP HID	CMSIS-DAP Programming/Debugging UART Bridging Drag-And-Drop Programming

2. Installing and Using KitProg3



To use KitProg3, you need one or more of the following tools:

- [ModusToolbox IDE](#) and [Cypress Programmer](#)
- [PSoC Creator](#) and [PSoC Programmer](#)
- Bridge Control Panel (BCP) from Cypress for USB-I2C and USB-SPI bridging
- A terminal emulator for USB-UART bridging
- A supported kit (see [Supported Kits](#))

KitProg3 is communication firmware used by these tools. Bridge Control Panel is installed with PSoC Programmer. PSoC Programmer versions before 3.28 do not support KitProg3.

2.1 Installing KitProg3

Install ModusToolbox IDE, Cypress Programmer or PSoC Programmer before using any kit with KitProg3. Any required driver is installed by the tools that use KitProg3. There is no separate installer for KitProg3.

You can also get the latest version of Firmware Loader at the [Cypress GitHub repository](#). The Firmware Loader does not install any drivers, but you can use it to upgrade (or downgrade) the KitProg firmware on a kit.

The supported Cypress kits have either KitProg3 or KitProg2 already installed. See [Upgrading to KitProg3](#) to learn how to tell what's installed, and how to upgrade.

When you plug in a kit, depending upon your circumstances and host operating system, you may see a message that drivers are being installed.

KitProg3 enumerates as a USB Composite Device, and as a USB-UART port.

2.2 Using KitProg3

You do not use KitProg3 directly. You use a programming tool or IDE that automatically connects to and uses KitProg3. In most cases, KitProg3 is completely transparent.

2.2.1 Connecting

Plug in the kit. Use the USB cable that came with the kit and connect the host computer to the kit. KitProg3 is powered via the USB cable.

When you plug in the kit, an amber LED turns ON. On most kits, this is LED2. On MiniProg4, it is labeled **Mode**. The precise designation varies by kit.

If the LED is steady (out of box behavior), KitProg3 is using Bulk endpoints for faster communication. If the LED is ramping, KitProg3 is using HID endpoints, which means slower communication. If the LED is OFF, KitProg3 is in DAPLink mode.

KitProg3 from the factory defaults to Bulk endpoints because they are faster. You can switch KitProg3 between Bulk and HID endpoints should you need to. See [Mode Switching](#).

Launch your programming tool or the ModusToolbox IDE. The tool connects to KitProg3 automatically. The KitProg3 connection appears in the UI of the programming tool. Note that at this time, Cypress programming tools do not recognize the kit if it is in DAPLink mode. Switch the kit to Bulk or HID, and the tool can see and work with the kit. See [Mode Switching](#).

Figure 2-1. Connected via Cypress Programmer

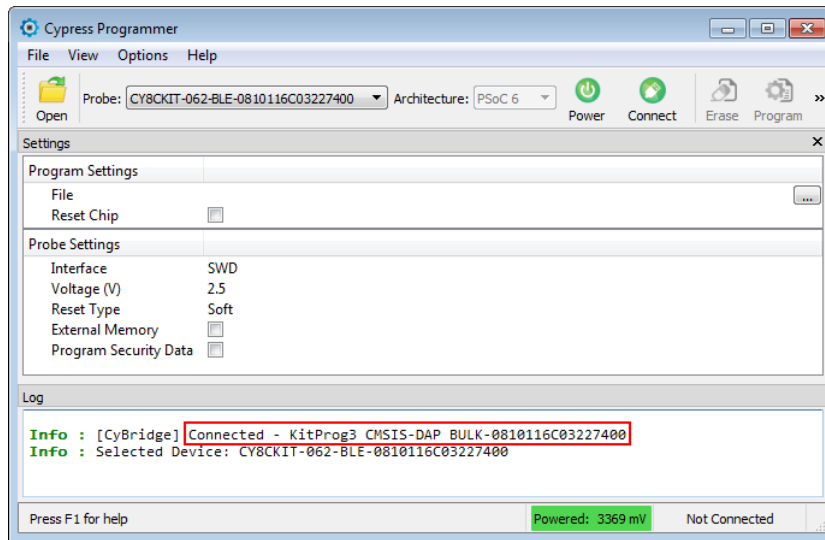
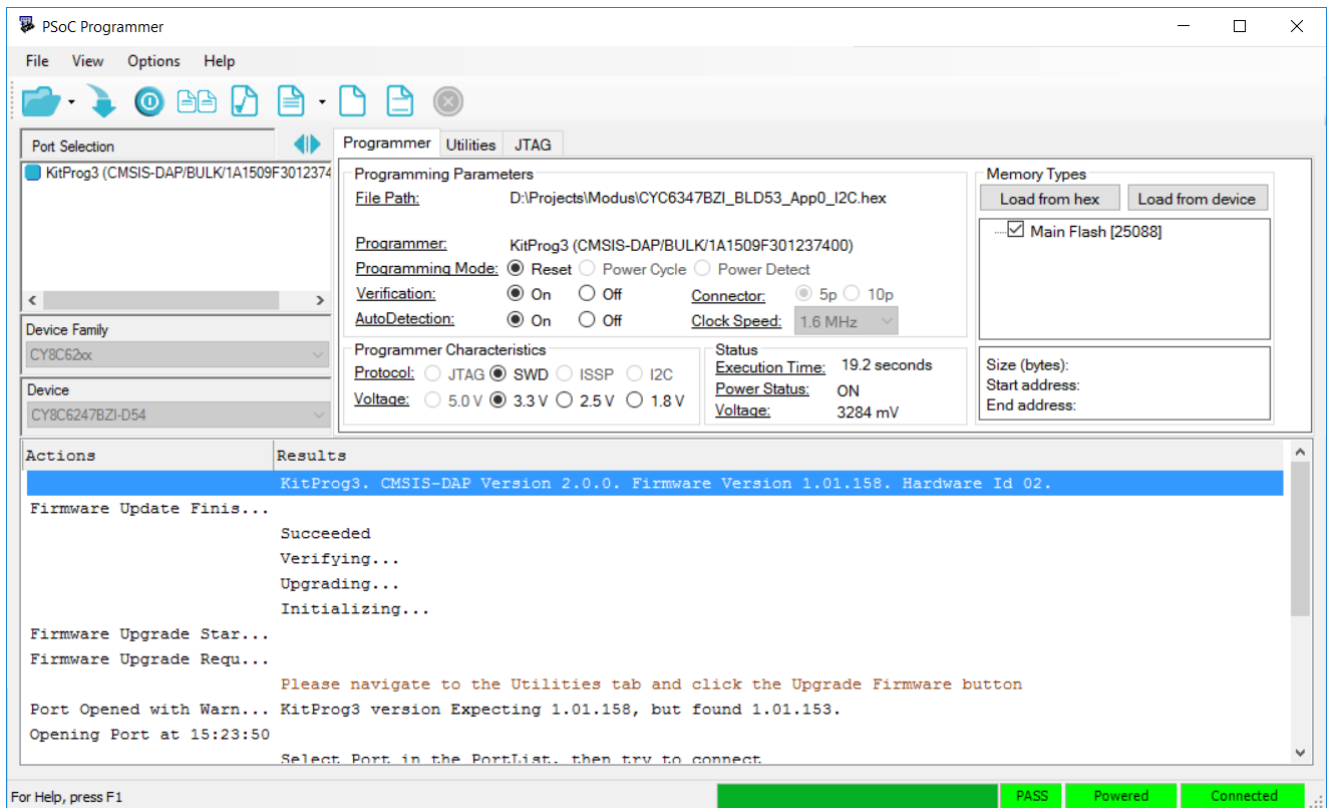


Figure 2-2. Connected via PSoC Programmer

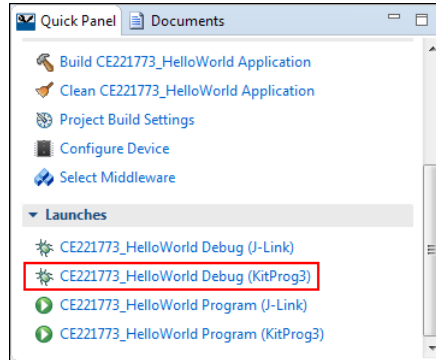


2.2.2 Programming and Debugging

Use the program or debug commands in your tool. See your IDE's documentation for details.

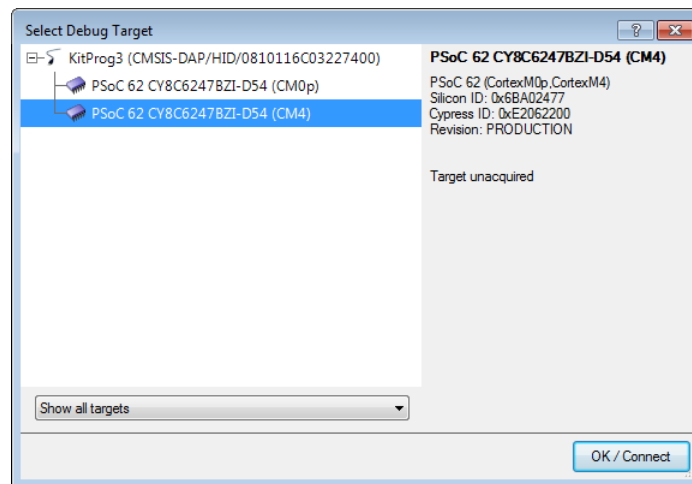
For example, for ModusToolbox IDE, click the **Debug (KitProg3)** link in the **Quick Panel**. To program the device without debugging, use the **Program (KitProg3)** link.

Figure 2-3. Launching a Debug Session in ModusToolbox IDE



For PSoC Creator, use the **Debug** or **Program** commands in the **Debug** menu. Then select your target and click **OK**.

Figure 2-4. Selecting the Debug Target in PSoC Creator



2.2.3 Mode Switching

KitProg3 supports Bulk and HID endpoints. Switch to HID endpoints if you need to for your design or hardware. Otherwise, stay in Bulk mode for best performance.

Each supported kit has a **Mode Select** switch. Press the switch to change operation between Bulk and HID endpoints. When in Bulk mode, the amber LED is ON and steady. When in HID mode, the amber LED ramps up and down. In each mode, bridging (USB-I2C, USB-SPI, or USB-UART) is available while debugging. See [KitProg3 LEDs](#) for information on how KitProg uses the LEDs.

The precise designation for the mode switch varies based on the kit. For example, on some Cypress Pioneer kits and the MiniProg4 debug probe, it is labeled **Mode Select**. Use your kit documentation if you can't find the switch.

KitProg3 also has DAPLink and Bootloader modes. Getting into these modes is slightly different.

In and out of DAPLink mode

To switch to DAPLink, on a kit with one mode switch, press the switch for more than two seconds. Press the mode switch again (no need to hold it down) to return to normal operation.

On a kit with a second mode switch, press that switch to switch to DAPLink. Press again to return to normal operation. The specific designation and location of the second mode switch varies based on the kit. On the MiniProg4 debug probe, it is labeled **Custom App**.

In DAPLink mode, the amber LED switches OFF. Other LEDs may light up (depending on active communication with PC). See [KitProg3 LEDs](#) for information on how KitProg uses the LEDs.

When you switch out of DAPLink mode, KitProg3 resumes its previous user state (either Bulk or HID endpoints).

In and out of Bootloader mode

If your KitProg3 image is corrupted, you can use bootloader mode to update the firmware. See [Installing KitProg3](#) for information on where to get the KitProg3 image.

To get into Bootloader mode, press the mode switch *while plugging in the board*.

To get out of Bootloader mode and return to normal operation, unplug the kit and reconnect.

2.2.4 KitProg3 LEDs

The KitProg3 user interface is limited to one or two mode switches and one or three status LEDs, depending upon the kit. The name and location of the mode switch(es) vary per kit. See the kit documentation to understand what switches and LEDs on the kit are used for KitProg3. See [Mode Switching](#) for Information on how to use the mode switches.

The table describes how KitProg3 uses LEDs to let you know what's going on. In the User Modes, green means success and red means there was a problem.

Table 2-1. Status LEDs

	Programming Mode	Programming Status	Three LED Kit			Single LED Kit
			Amber LED	Green LED	Red LED	Amber LED
User modes	CMSIS-DAP Bulk	Programming	ON	8 Hz	OFF	8 Hz
		Success		ON	OFF	ON
		Error		OFF	ON	2 Hz
	CMSIS-DAP HID	Programming	Ramping 1 Hz	8 Hz	OFF	8 Hz
		Success		ON	OFF	Ramping 1 Hz
		Error		OFF	ON	2 Hz
Advanced modes	Bootloader		1 Hz	N/A	N/A	1 Hz
	DAPLink		OFF	Flashes when the USB MSC interface is active	Flashes when the USB CDC interface is active	OFF

3. KitProg3 Design



KitProg3 firmware is designed to run on specific hardware using a PSoC 5LP device. The hardware design is unchanged between KitProg2 and KitProg3. As a result, any kit that supports KitProg2 can be upgraded to KitProg3.

To support a variety of Cypress kits with varying feature sets and capabilities, KitProg3 firmware uses an internal hardware ID and adjusts the available features to fit the capabilities of the kit.

3.1 Supported Kits

Table 3-1 lists the development kits that support KitProg3.

Note: Kits released with KitProg2 can be upgraded to KitProg3. See [Upgrading to KitProg3](#).

Table 3-1. Kit Support

Development Kits	Mode Switches
CY8CKIT-041-40XX PSoC 4 S-Series Pioneer Kit	One
CY8CKIT-041-41XX PSoC 4100S Pioneer Kit	One
CY8CKIT-048 PSoC Analog Coprocessor Pioneer Kit	One
CY8CKIT-145-40XX PSoC 4 S-Series Prototyping Kit	One
CY8CKIT-146 PSoC 4200DS Prototyping Kit	One
CY8CKIT-147 PSoC 4100PS Prototyping Kit	One
CY8CKIT-148 PSoC 4700S Inductive Sensing Evaluation Kit	One
CY8CKIT-149 PSoC 4100S Plus Prototyping Kit	One
CY8CKIT-062 BLE Pioneer Kit	Two
CY8CKIT-062-WiFi-BT PSoC 6 WiFi-BT Pioneer Kit	Two
CY8CKIT-005 MiniProg4 Program and Debug Kit	Two
CY8CPROTO-062-4343W PSoC 6 Wi-Fi BT Prototyping Kit	One
CY8CKIT-062-4343W PSoC 6 Wi-Fi BT Pioneer Kit	Two

3.2 Operating Speeds

Table 3-2. KitProg3 Operating Speeds

Functionality	Supported Speed	Units	Comments
Programmer	1.6	MHz	-
USB-UART Bridge	1200, 2400, 4800, 9600, 19200, 38400, 57600, 115200, 250000, 500000, 1000000, 2000000, 3000000	Baud	Data bits – 8, Parity – None, Stop bits - 1
USB-I2C Bridge	50, 100, 400, 1000	kHz	-
USB-SPI Bridge	50–6000	kHz	-

4. DAPLink Mode



Arm Mbed DAPLink is open-source software that provides alternative platform-independent programming/debugging interfaces between the target application and host PC. KitProg3 includes DAPLink.

DAPLink is supported by:

- ModusToolbox IDE and Cypress Programmer
- Mbed Online IDE (Arm® Mbed Enabled™ technology)
- Mbed CLI (Arm® Mbed Enabled™ technology)
- Any IDE that supports CMSIS-DAP HID protocol

4.1 Supported Kits

Table 4-1 lists the development kits that support the [Arm® Mbed Enabled™](#) program. Other KitProg3-based kits will have DAPLink mode available, but it will not work with Mbed Ecosystem.

Table 4-1. Arm Mbed Enabled Kit Support

Development Kits	Mode Switches	Mbed support
CY8CKIT-062 BLE Pioneer Kit	Two	Yes
CY8CKIT-062-WiFi-BT PSoC 6 WiFi-BT Pioneer Kit	Two	Yes
CY8CPROTO-062-4343W PSoC 6 Wi-Fi BT Prototyping Kit	One	Yes
CY8CKIT-062-4343W PSoC 6 Wi-Fi BT Pioneer Kit	Two	Yes

4.2 Mbed Ecosystem

To use DAPLink, you need to upgrade kit firmware to KitProg3 v1.10 or later. See [How-To: Update DAPLink FW](#) section for details.

See the *Mbed OS to ModusToolbox Flow* section of the ModusToolbox User Guide for Mbed Ecosystem. You can download it on [ModusToolbox IDE](#) website (Documentation tab).

4.3 Features

DAPLink provides three interfaces: drag-and-drop programming, a serial port, and debugging support.

4.3.1 Drag-and-Drop Programming

Program the target PSoC 6 MCU device by copying or saving a file in one of the supported formats to the DAPLink drive. Upon completion, the drive remounts. If a failure occurs, the file FAIL.TXT appears on the drive containing information about the failure.

Supported file formats:

- Raw binary file
- Intel Hex

You can control DAPLink with [MSD Commands](#).

Note: DAPLink provided with KitProg3 does not implement the **start_bl.act** MSD command because it uses the KitProg3 Bootloader.

4.3.2 Serial Port

The serial port is connected directly to the target PSoC 6 MCU device allowing for bidirectional communication. It also allows the target to be reset by sending a break command over the serial port.

Table 4-2. UART Parameters and Speed

Functionality	Supported Speed	Units	Comments
USB-UART Bridge	1200, 2400, 4800, 9600, 19200, 38400, 57600, 115200, 250000, 500000, 1000000, 2000000, 3000000	Baud	Data bits – 8, Parity – None, Stop bits - 1

4.3.3 Debugging

You can debug with any IDE that supports the CMSIS-DAP protocol. Some tools capable of debugging are:

- [ModusToolbox](#)
- [Mbed CLI](#)
- [uVision](#)

4.3.4 User Interface

See [KitProg3 LEDs](#) to learn how they are used to let you know what's going on with DAPLink.

DAPLink UART settings are: Data bits – 8, Parity – None, Stop bits – 1.

See [Mode Switching](#) for instructions on how to get into and out of DAPLink mode.

4.4 How To

4.4.1 Switch to and from DAPLink Mode

See [Mode Switching](#).

4.4.2 Update DAPLink Firmware

DAPLink is part of the latest KitProg3 package. See [Installing KitProg3](#) for information on where to get KitProg3. When you update KitProg to a version that includes DAPLink, DAPLink is installed.

If you wish to upgrade KitProg to a version that includes DAPLink, you need a tool called Firmware Loader. ModusToolbox software includes this tool to update Cypress kits with KitProg3 firmware, and to switch the KitProg firmware from KitProg2 to KitProg3, and back. The default install directory of this tool is:

```
<ModusToolbox Directory>\tools\fw-loader-2.1\bin\
```

Note: On a Linux machine, you must first run the `udev_rules/install_rules.sh` script, before the first run of the fw-loader tool.

Use the appropriate commands on your host OS to open a command line window in the Firmware Loader `bin` directory and run `fw-loader`.

Use the following options, as needed:

- `--help` or `/?` (or no arguments) – Display a list of supported commands with their descriptions.
- `--device-list` – Display a list of connected devices.
- `--update-kp3 [device-name]` – Update the FW of the specified device name to latest KitProg3

If only one device is connected, you don't need to specify the device name.

See the *KitProg Firmware Loader* section of the ModusToolbox IDE User Guide for more details about KitProg Firmware Loader. You can download it on [ModusToolbox IDE](#) website (Documentation tab).

5. KitProg3 vs. KitProg2



Use this chapter to understand the differences and decide which to use.

5.1 Feature Comparison

Table 5-1. KitProg Feature Comparison

Feature	DAPLink	KitProg3	KitProg2
Protocol	Serial Wire Debug (SWD)	Serial Wire Debug (SWD)	Serial Wire Debug (SWD)
Transport Mechanism	CMSIS DAP v1.10	CMSIS DAP v2.0.0 CMSIS DAP v1.2.0	CMSIS DAP v1.1.0 Proprietary
USB Mass Storage Device	Yes	No	Yes
USB Endpoints	HID, CDC, MSC	Bulk and HID	HID only
IDE Support	ModusToolbox IDE Mbed CLI uVision	ModusToolbox IDE PSoC Creator IDE	PSoC Creator
Programmer Support	N/A	Cypress Programmer PSoC Programmer	PSoC Programmer
Kit support	See Arm Mbed Enabled™ Kit Support	The same, see Supported Kits	

This comparison does not include programming speed because that number depends upon several variables, such as the target flash memory (type and size), programming tool overhead, and the transport mechanism. These vary widely from tool to tool, and kit to kit. However, KitProg3 is 2-4x faster than KitProg2 using the CMSIS-DAP transport mechanism, because it uses Bulk endpoints for faster data transfer.

5.2 Upgrading to KitProg3

You can upgrade the kit firmware by using one of these:

- PSoC Programmer (doesn't include latest KitProg3 FW)
- Cypress Programmer (doesn't include latest KitProg3 FW)
- FW Loader Tool (includes latest KitProg3 FW with DAPLink)

This section describes how to upgrade kit firmware to KitProg3 by using PSoC Programmer or Cypress Programmer. See [How-To: Update DAPLink FW](#) for instructions on how to upgrade kit FW to latest KitProg3 FW with DAPLink.

Because tools are released individually and on their own schedule, the most recent release of a programming tool may not have the very latest version of KitProg3 to install and update. You can also use the FW Loader tool that comes with ModusToolbox software to upgrade the kit.

Use Cypress Programmer or PSoC Programmer to connect to a kit. The tool connection tells you what firmware is on the kit. If KitProg2 is installed on the kit, the tool notifies you, and gives you the option to upgrade firmware. Click **OK** to leave the firmware unchanged.

Click **Upgrade Firmware** and KitProg3 is loaded into the kit. The Cypress Programmer or PSoC Programmer log window provides progress information and confirms connection to the new KitProg3 firmware on the kit.

Figure 5-1. Upgrading to KitProg3 in Cypress Programmer

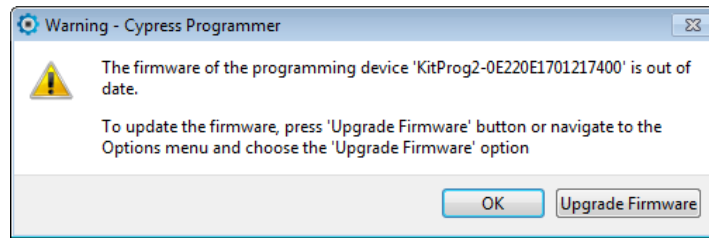
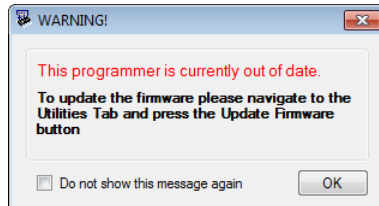


Figure 5-2. KitProg3 Firmware Update Warning in PSoC Programmer



5.3 Downgrading to KitProg2

This is only necessary if you have upgraded a kit to KitProg3, and you then wish to use an older version of PSoC Programmer (before v 3.28) with PSoC Creator.

ModusToolbox software includes a command-line tool named “fw-loader” to update Cypress kits and switch the KitProg firmware from KitProg2 to KitProg3, and back. The default install directory of this tool is:

```
<ModusToolbox Directory>\tools\fw-loader-2.1\bin\
```

Note: On a Linux machine, you must first run the `udev_rules/install_rules.sh` script, before the first run of the fw-loader tool.

Use the appropriate commands on your host OS to open a command line window in that directory and run `fw-loader`.

Use the following options, as needed:

- `--help` or `/?` (or no arguments) – Display a list of supported commands with their descriptions.
- `--device-list` – Display a list of connected devices.
- `--update-kp3 [device-name]` – Update the FW of the specified device name to KitProg3.
- `--update-kp2 [device-name]` – Update the FW of the specified device name to KitProg2. If only one device is connected, you don’t need to specify the device name.
- Please look to the Section 8 from ModusToolbox IDE User Guide for more details about KitProg Firmware Loader. You can download it on the [ModusToolbox IDE](#) website (Documentation tab).

The PSoC Programmer installation contains a hex file for KitProg2 in the root directory. You can also program the KitProg2 hex file manually using a MiniProg3 probe.

6. Troubleshooting



This section lists known issues, along with any workaround.

I use Windows 7. When in Bulk mode (amber LED on), the kit is not recognized by the programming tool, and debug does not work with ModusToolbox or PSoC Creator IDE. If I switch to HID mode, it works.

This is a known issue with Windows 7 driver updates. Instead of using the correct driver, Windows update installs the HP Printer (BIDI) driver *if the machine is connected to the Internet*. As a result, KitProg3 will not work when in Bulk mode. The problematic HP Printer driver is no longer available in the Windows update system. However, it is possible that the wrong driver was installed at an earlier time.

To fix this issue, follow these steps:

1. Uninstall the driver from Device Manager.
2. Close any internet connection.
3. Attach the kit to your computer and rescan the device in Device Manager.

How do I recover a corrupted KitProg3 image?

Although unlikely, it is possible to corrupt the KitProg3 image, for example, if a firmware update is interrupted.

To fix this issue, follow these steps.

1. Put KitProg3 into bootloader mode. (Press the mode switch while plugging in the kit.) Then follow the instructions for your programmer.
2. Launch Cypress Programmer. Cypress Programmer automatically updates the KitProg3 firmware.

OR

2. Launch PSoC Programmer. Update KitProg3 firmware via **Utilities > Update firmware** option.

Mbed CLI interface shows errors and warning while running Mbed on Cypress kits.

This is a known issue caused by Mbed OS installation problems. See the *Install and configure Mbed CLI* section of the ModusToolbox IDE User Guide. You can download it on the [ModusToolbox IDE](#) website (Documentation tab).

Revision History



Document Revision History

Document Title: KitProg3 User Guide			
Document Number: 002-24616			
Revision	Issue Date	Origin of Change	Description of Change
**	10/26/2018	JETT	New kit guide.
*A	11/08/2018	SRDS	Updated Introduction: Updated Description. Updated KitProg3 Tools Support and Compatibility: Updated Installing and Using KitProg3: Updated Using KitProg3: Updated Programming and Debugging: Updated description. Updated Mode Switching: Updated description. Updated KitProg3 Design: Updated KitProg3 : Updated description. Updated KitProg3 vs. KitProg2: Updated Downgrading to KitProg2: Updated description. Updated Troubleshooting: Updated description.
*B	11/22/2018	JETT/VITA	Added PSoC Creator/PSoC Programmer information
*C	02/19/2019	GRYT/VITA	Updates throughout for KitProg3 v1.1 and DAPLink Updated Introduction. Added DAPLink Mode. Updated KitProg3 User Interface. Updated Troubleshooting. Updated KitProg3 Operating Speeds. Updated KitProg3 Status LEDs. Updated Downgrading to KitProg2. Updated Upgrading to KitProg3.
*D	2/21/2019	JETT	Remove listing of an unsupported kit