



**Please note that Cypress is an Infineon Technologies Company.**

The document following this cover page is marked as “Cypress” document as this is the company that originally developed the product. Please note that Infineon will continue to offer the product to new and existing customers as part of the Infineon product portfolio.

**Continuity of document content**

The fact that Infineon offers the following product as part of the Infineon product portfolio does not lead to any changes to this document. Future revisions will occur when appropriate, and any changes will be set out on the document history page.

**Continuity of ordering part numbers**

Infineon continues to support existing part numbers. Please continue to use the ordering part numbers listed in the datasheet for ordering.

# ModusToolbox® 2.3 Tools Package Release Notes

Production Release

## Overview

ModusToolbox software is a set of tools that enable you to integrate Cypress devices into your existing development methodology. ModusToolbox software consists of various libraries and middleware on GitHub, as well as an IDE and tools package installed on your computer. For more details about what is included with ModusToolbox software, refer to the [ModusToolbox User Guide](#).

This ModusToolbox 2.3 tools package is a complete release. It includes the latest features from all previous releases, including patches. This release does not replace any existing installed releases; it installs alongside them. If you have more than one release installed, refer to the [ModusToolbox Installation Guide](#), Product Versioning section.

This document describes the features and known limitations for the ModusToolbox software provided as part of the ModusToolbox 2.3 tools package included with the installer.

## Contents

Overview .....	1
Contents .....	1
What's Changed.....	2
New EZ-PD Configurator .....	2
Updated KitProg3 Firmware .....	2
Integrated Third Party IDE Support .....	2
Third Party IDE Secure Device Support.....	2
Improved Library Support and Control .....	3
Updated BSP Creation and Update Processes.....	3
Eclipse Update and macOS Big Sur Support.....	3
Windows Multi-User Installation Support.....	3
Update Quick Panel Performance .....	3
Project Creation / Library Management Message Improvements .....	3
Removed Power Estimator Tool.....	3
What's Included.....	4
Supported Tool Chains .....	4
Supported Boards .....	5
Known Issues/Limitations.....	5
Installation.....	5
Project Creator.....	6
Proxy.....	6

Building/Programming/Debugging.....	7
Eclipse IDE .....	11
Library Manager.....	13
Device Configurator .....	14
CapSense Tuner.....	14
Documentation.....	15
Open Source .....	15
Further Reading .....	15

## What's Changed

This release of the ModusToolbox tools package includes the following updates and features:

### ***New EZ-PD Configurator***

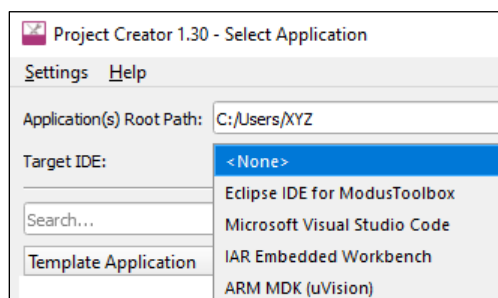
The Easy Power Delivery (EZ-PD) Configurator provides a user-friendly tool for selecting the features and configuring parameters of the PDStack middleware.

### ***Updated KitProg3 Firmware***

The KitProg3 firmware version was updated to deliver GPIO bridging support, along with several bug fixes. Refer to the [KitProg3 User Guide](#) for details on how to update your kit.

### ***Integrated Third Party IDE Support***

For previous versions of ModusToolbox, you could export applications to other IDEs. However, this was only possible through the command line. With this release, the Project Creator tool now includes an option to specify one of several third-party IDEs.



You can still use the command line if you prefer. Either export option generates the appropriate IDE support files and settings.

### ***Third Party IDE Secure Device Support***

The commands to export ModusToolbox applications to IAR EW-ARM and Arm MDK have been updated to populate post-build entries in the generated files with secure tools command(s).

**Note** The Eclipse IDE for ModusToolbox and VS Code already include this support.

### ***Improved Library Support and Control***

In this release, we are now locking the library versions used in your application to the latest release tag available, so that you don't have to worry about locking versions down by yourself. To learn more about how our library versioning works, review the [ModusToolbox User Guide Product Versioning](#) section. We also notify you about the updates available for a given library in the Library Manager. Plus, we've made the default BSP location to be local by default. However, this behavior is also controlled via the manifest, which allows you to change the BSP for a given application without worrying that the changes will propagate to other applications. You can still make the BSP shared using the Library Manager.

### ***Updated BSP Creation and Update Processes***

We have removed the option to change the device or companion device from the Device Configurator. Instead, we have added the command `make update_bsp` for you to be able to change the primary device via the command line. Contact support for details about updating the companion device.

### ***Eclipse Update and macOS Big Sur Support***

All ModusToolbox tools have been validated to be compatible with macOS Big Sur. As a part of that, we have transitioned to a newer version (4.17) of the Eclipse IDE included in the ModusToolbox Tools package.

### ***Windows Multi-User Installation Support***

ModusToolbox 2.3 installation on Windows has been updated to allow a one-time, admin-enabled installation of the pre-requisite software and drivers, as well as any number of nonprivileged installations of the tools (in the user's space). This allows multiple users of the same machine to use ModusToolbox and also share the same installation among the multiple users on the same Windows machine.

### ***Update Quick Panel Performance***

The Quick Panel has been updated to improve response time when switching between different projects in the Project Explorer.

### ***Project Creation / Library Management Message Improvements***

Various messages in the Project Creator and Library Manager tools have been updated to provide a more user-friendly experience, including:

- Errors, warnings, and success messages are now color-coded.
- Various inaccurate statements have been corrected.

### ***Removed Power Estimator Tool***

For this release, we have temporarily removed the Power Estimator (CyPE) tool. We plan to improve it for a future release. If needed, you can use the tool from a previous release of ModusToolbox.

## What's Included

This release includes the following tools and versions:

Tool Name	Current Release	Previous Release (Patch)
Eclipse IDE for ModusToolbox	2.3.0	2.2.0
Bluetooth Configurator	2.30	2.20
CapSense® Configurator	3.15	3.10 (3.11 patch)
CapSense Tuner	3.15	3.10 (3.11 patch)
Configurator Backend CLI	2.30	2.20 (2.21 patch)
CyMCUEIfTool	1.0	Unchanged
Device Configurator	3.0	2.20 (2.21 patch)
Device Firmware Update (DFU) Host Tool	1.40	1.30
EZ-PD Configurator	1.0	N/A
Firmware Loader	3.2.0 [KitProg3 2.20]	3.1.0 [KitProg3 2.10]
GCC	9.3.1	Unchanged
JRE	11.0.10.9 [version 11]	1.8.0_252 [version 8]
Library Manager	1.30.0	1.2.0
GNU make Build System	1.2	1.2
modus-shell	1.20 (Windows) 1.1.0 (Linux/macOS)	1.1.0
OpenOCD (Cypress-specific)	4.2.0	4.1.0
Project Creator	1.30.0	1.2.0
Proxy Helper	1.1.0	Unchanged
Python (for Windows)	3.7.7.3	3.7.7.2
QSPI Configurator	2.30	2.20
Secure Policy Configurator	1.10	(1.0 patch)
Segment LCD Configurator	1.30	1.20 (1.21 patch)
Smart I/O™ Configurator	3.0	2.20 (2.21 patch)
USB Configurator	2.30	2.20

## Supported Tool Chains

The GCC Arm Embedded toolchain GCC 9.3.1 is installed with the ModusToolbox software. This toolchain has no use restrictions and does not require license activation (it is distributed under the terms of the GNU Public License).

Although not installed with ModusToolbox software, the build system also supports these tool chains for PSoC 6 MCU applications:

- Arm compiler v6.11 (Windows and Linux hosts)
- IAR Embedded Workbench v8.32 minimum (Windows only)

## Supported Boards

The Cypress boards available for use varies with different releases of BSPs and libraries on GitHub. You can see the current list of BSPs in the Project Creator tool using the default manifest URL:

Kit Name	MCU	Connectivity Device
PSoC 4 BSPs		
CY8CKIT-041-41XX	CY8C4146AZI-5433	<none>
CY8CKIT-145-40XX	CY8C4045AZI-5413	<none>
CY8CKIT-149	CY8C4147AZI-5475	<none>
PSOC4-GENERIC	CY8C4548AZI-5485	<none>
PSoC 6 BSPs		
CY8CKIT-062-BLE	CY8C6347BZI-BLD53	<none>
CY8CKIT-06252-43012	CY8C624ABZI-S2D44	CYW43012COWKWBG
CY8CKIT-062-WIFI-BT	CY8C6247BZI-D54	CYW4343WKUBG
CY8CKIT-064B052-4343W	CY8C644ABZI-S2D44	CYW4343WKUBG
CY8CPROTO-062-4343W	CY8C624ABZI-S2D44	CYW4343WKUBG
CY8CPROTO-06253-4343W	CY8C6245LQI-S3D72	CYW4343WKUBG
CY8CPROTO-063-BLE	CYBLE-416045-02	<none>
CY8CPROTO-064B053	CY8C6445LQI-S3D42	<none>
CYSBSYSKIT-01	CY8C624AFNI-S2D43	<none>
CYSBSYSKIT-DEV-01	CY8C624AFNI-S2D43	CYW43012COKFFBH
CYW9P6251-43012EVB-01	CY8C6247BZI-D52	CYW43012CDEKUBG
CYW9P6251-4343EVB-01	CY8C6247BZI-D54	CYW4343WKUBG
PSOC6-GENERIC	CY8C6347BZI-BLD53	<none>
WICED Bluetooth BSPs		
CYBT-213043-EVAL	CYBT-213043-02	CYW20819A1
CYBT-213043-MESH	CYBT-213043-02	CYW20819A1
CYBT-223058-EVAL	CYBT-223058-02	CYW20819A1
CYBT-243053-EVAL	CYBT-243053-02	CYW20820A1
CYBT-253059-EVAL	CYBT-253059-02	CYW20820A1
CYBT-263065-EVAL	CYBT-263065-02	CYW20819A1
CYBT-273063-EVAL	CYBT-273063-02	CYW20819A1
CYBT-343026-EVAL	CYW20706A2	CYW20706A2
CYBT-353027-EVAL	CYW20706A2	CYW20706A2
CYBT-413055-EVAL	CYBT-413055-02	CYW20719B2
CYBT-413061-EVAL	CYBT-413061-02	CYW20719B2
CYBT-423054-EVAL	CYBT-423054-02	CYW20719B2
CYBT-423060-EVAL	CYBT-423060-02	CYW20719B2
CYBT-483056-EVAL	CYBT-483056-02	CYW20719B2
CYBT-483062-EVAL	CYBT-483062-02	CYW20719B2
CYW920706WCDEVAL	CYW20706A2	CYW20706A2
CYW920719B2Q4EVB-01	CYW20719B2KUMLG	CYW20719B2
CYW920721B2EVK-02	CYW20721B2KWB9G	CYW20721B2
CYW920721B2EVK-03	CYW20721B2KUMLG	CYW20721B2
CYW920721M2EVK-01	CYW20721B2KUMLG	CYW20721B2
CYW920721M2EVK-02	CYW20721B2KWB9G	CYW20721B2
CYW920735Q60EVB-01	CYW20735B1	CYW20735B1
CYW920819EVB-02	CYW20819A1KFBG	CYW20819A1
CYW920819REF-KB-01	CYW20819A1KFB1G	CYW20819A1
CYW920820EVB-02	CYW20820A1KFBG	CYW20820A1
CYW920835REF-RCU-01	CYW20835B1	CYW20835B1
CYW943012BTEVK-01	CYW43012COWKWBG	CYW43012C0
CYW98920EVB-01	CYW89820BWM1G	CYW20820A1
CYW9M2BASE-43012BT	CYW43012C0	CYW43012C0
XMC BSPs		
KIT_XMC14_BOOT_001	XMC1404-Q064x0200	<none>
KIT_XMC47_RELAX_V1	XMC4700-F144x2048	<none>

**Note** Additional boards will be made available on an ongoing basis.

## Known Issues/Limitations

This section lists the known issues/limitations of this release:

### Installation

Problem	Workaround
On common Linux distributions, the serial UART ports (usually /dev/ttySx or /dev/ttyUSBx devices) belong to the root user and to the dialout and plugdev groups. Standard users are not allowed to access these devices.	<p>An easy way to allow the current user access to the Linux machine's serial ports is by adding the user to the dialout or plugdev group. This can be done using the following command:</p> <pre>\$sudo usermod -a -G dialout,plugdev \$USER</pre> <p><b>Note</b> For this command to take effect, the user must log out and then log back in.</p>

## Project Creator

Problem	Workaround
Using the Project Creator Target IDE feature, the application that was exported to a third-party IDE doesn't work in that IDE.	There are limitations to the export process. For example IAR and $\mu$ Vision are not supported in macOS or Linux. In other cases, you need to update configuration settings in the third-party IDE. Refer to the "Export to IDEs" chapter in the <a href="#">ModusToolbox User Guide</a> for more details.
In the default case, project creation and library updates require internet access. In some cases, using a VPN may adversely affect the time required for these operations due to a delay in DNS server response times for external IP addresses such as github.com.	<p>If you experience issues such as this, project creation and library update times can be improved by disabling the VPN connection during those operations.</p> <p>To determine if this is an issue for you, open a terminal window and enter "nslookup github.com". If the response is not immediate, then you may benefit from disabling your VPN connection.</p>
<p>When trying to create a new project you may not see any BSPs or applications to select, as well as the following error message:</p> <p>Unable to open file at  <a href="http://github.com/cypresssemiconductorco/mtb-super-manifest/raw/v2.X/mtb-super-manifest-fv2.xml">http://github.com/cypresssemiconductorco/mtb-super-manifest/raw/v2.X/mtb-super-manifest-fv2.xml</a></p> <p>Some boards and applications may be missing. Check the console for a detailed error message.</p>	<p>Verify that you have a network connection and see the instructions in the <a href="#">Installation</a> problem section above concerning proxy settings.</p> <p>This may also be caused by your work environment not having access to GitHub. Refer to <a href="#">KBA230953</a> for details on how to work around this problem.</p>

## Proxy

Problem	Workaround
<p>When trying to create a new project, you may see the following error message:</p> <p>Unable to open file at  <a href="http://github.com/cypresssemiconductorco/mtb-super-manifest/raw/v2.X/mtb-super-manifest.xml">http://github.com/cypresssemiconductorco/mtb-super-manifest/raw/v2.X/mtb-super-manifest.xml</a>.</p> <p>Some boards and apps may be missing. Check the logfile for a detailed error message.</p>	<p>This can happen if you are behind a firewall and do not have your proxy settings configured. You must set your HTTP_PROXY and HTTPS_PROXY environment variables or use the Proxy Helper tool from the <b>Settings</b> menu. You can also find it in:</p> <p>&lt;install_path&gt;/ModusToolbox/tools_2.3/proxy-helper</p>
<p>In some cases, incorrect proxy settings in the Project Creator or Library Manager tool can prevent the proxy server settings from being edited to correct values.</p>	<p>Run the Proxy Helper tool from the <b>Settings</b> menu to reset the proxy mode to direct. The tool is also located in:</p> <p>&lt;install_path&gt;/ModusToolbox/tools_2.3/proxy-helper</p> <p>For example:</p> <pre>./proxy-helper --config set mode=direct</pre>

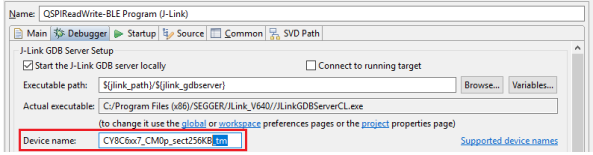
## Building/Programming/Debugging

Problem	Workaround
<p>The ModusToolbox build process stops at some points, such as during auto-discovery, and it takes longer than other times.</p>	<p>This is likely due to anti-virus software scanning various ModusToolbox directories. To resolve this, change your anti-virus settings to exclude the following as appropriate for your installation (where ~ is your home directory):</p> <ul style="list-style-type: none"> <li>• Installation directory (e.g., ~/ModusToolbox/)</li> <li>• Settings directory (e.g., ~/.modustoolbox/)</li> <li>• Applications directories (e.g., ~/MyExamples/)</li> <li>• Eclipse IDE workspace directory (e.g., ~/mtw/)</li> </ul>
<p>For PSoC 6A-256K MCU "ERROR: Timeout while restoring target, core does not stop. (PC = 0x100009FC, XPSR = 0x01000003, SP = 0x08002EFC)!" can appear after several executing of Program launch configuration via Jlink probe by using JTAG interface.</p>	<p>Set Type of Initial Reset and Halt to '0' for Program Jlink launch configuration in Startup tab</p>
<p>J-Link launch configurations do not work for a PMG1-related project.</p>	<p>Use KitProg3 launch configs</p>
<p>While debugging an application built with the GCC toolchain, some breakpoints cannot be reached as the application code referenced by the breakpoints is inlined. <b>Note</b> The GDB behavior described above has no functional impact on the application execution.</p>	<p>Switch the optimization level from <code>-Og</code> (default in the <code>CONFIG=Debug</code> mode) to <code>-O0</code>, by editing the application Makefile:</p> <pre>CONFIG=Custom CFLAGS=-O0</pre> <p>Changing the optimization flag to <code>-O0</code> may impact the application behavior or memory usage. We recommend disabling the compiler optimizations only for debugging purposes.</p>
<p>PSoC 64 Secure MCUs do not support the programming and debugging via JTAG interface. It is because of the specifics of PSoC 64 MCUs - JTAG pins are configured after unpredictable amount of time after what causes a lot of warnings during programming via JTAG.</p>	<p>Use SWD interface with PSoC 64 Secure MCUs in Eclipse and VS Code IDEs</p>
<p>Unable to acquire PSoC 4 target with J-Link probe if target is in PROTECTED state. J-Link probe does not support device acquisition in Test Mode. This prevents invocation of SROM API making it impossible to re-program or erase the chip in PROTECTED state.</p>	<p>The workaround is to unprotect/erase the chip using MiniProg4 or KitProg3 probe.</p>
<p>A "Failed to read memory at &lt;address&gt;" message may appear in "Memory View" of the Eclipse IDE when connected via the MiniProg4 probe and try to read memory from invalid address.</p>	<p>You can safely ignore this error</p>
<p>In VS Code, usability behavior might be observed after a Restart operation with PSoC 64 Secure MCUs. For example, "Failed to launch OpenOCD GDB Server: Timeout" error may appear when KitProg3 Attach launch config is running, or if debug session is broken with J-Link.</p>	<p>No reliable workaround</p>
<p>In VS Code, "Exception has occurred" error might appear when you execute Attach launch configuration. Such errors started to be visible in VS Code - 1.53.2.</p>	<p>You can safely ignore this error.</p>



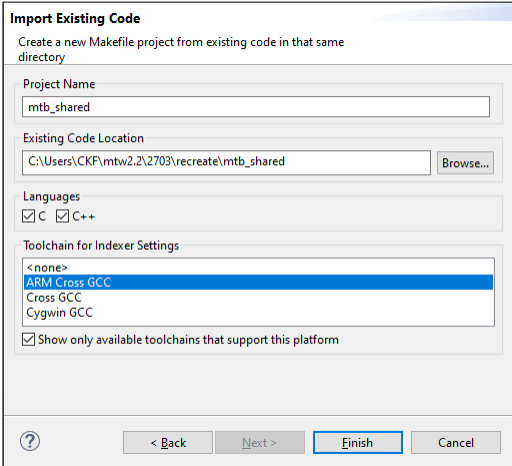
Problem	Workaround
<p>In VS Code, "Failed to launch undefined GDB Server: Timeout." error might appear if to launch Debug J-Link launch configuration with KIT_XMC47_RELAX_V1 after several 'step over', 'step into' operations during the previous Debug session what was stopped.</p>	<p>Re-run the launch configuration again after failure</p>
<p>In VS Code, the J-Link launch configuration does not stop at Reset_Handler after 'Restart', but at some other line of code.</p> <p>For PSoC 6 MCUs, this is associated with any "monitor" command in GDB. For example, if you press the Restart button, the Cortex-Debug plugin sends "monitor halt" and "monitor reset" commands to the GDB causing GDB to go out-of-sync with the target.</p> <p>GDB does not know what the "monitor xxx" command does. It does not expect that "monitor reset" will change the state of the target.</p> <p>For XMC and PSoC 4 MCUs, such behavior is caused by the using "Cortex M0" target for PSoC 4 and "Cortex-M4" for XMC devices as the workaround for other issue with Attach in SEGGER SW.</p>	<p>Step through code to enter the Reset_Handler.</p>
<p>When attempting to Program/Debug with Miniprogr4 or J-Link probes in JTAG mode, the tool displays errors similar to the following:</p> <pre> Info : KitProg3: Pipelined transfers enabled Info : VTarget = 3.276 V Info : clock speed 2000 kHz Error: JTAG scan chain interrogation failed: all zeroes Error: Check JTAG interface, timings, target power, etc. Error: Trying to use configured scan chain anyway... Error: psoc6.cpu: IR capture error; saw 0x00 not 0x01 Warn : Bypassing JTAG setup events due to errors Error: Invalid ACK (0) in DAP response Error: Invalid ACK (0) in DAP response Error: Invalid ACK (0) in DAP response           </pre>	<p>These errors are likely caused by the device being in sleep mode. Open the Device Configurator, and set the <b>System Idle Power Mode</b> setting to 'Active' to turn off DeepSleep mode. JTAG is not available in sleep mode. Alternately, use SWD to acquire the target, and then switch to JTAG.</p>
<p>If an application includes multiple BSPs that span major version numbers (for example, v1.x and v2.x) as local dependencies, there will be build conflicts. This is because of changes in the names of the BSP's dependencies.</p>	<p>There are two options:</p> <ol style="list-style-type: none"> <li>1) The application can be updated to use the new MTB flow with the BSPs (and dependencies) marked as shared.</li> <li>2) All local BSPs within a application must have the same major version number. Once the application is updated to only have a single major version of the BSP, any of the dependent libraries that were pulled down by the removed BSP version need to be removed from the application. This includes removing any <i>.lib</i> or <i>.mtb</i> files. The problematic dependencies that must be removed include the following: <ul style="list-style-type: none"> <li>1.x BSP dependencies: psoc6pdl, psoc6hal</li> <li>2.x BSP dependencies: mtb-pdl, mtb-hal</li> </ul> </li> </ol>

Problem	Workaround
<p>KitProg3 and J-Link launch configurations do not work when you change the build configuration from Debug to Release or vice-versa. This applies to the Eclipse IDE for ModusToolbox and Visual Studio (VS) Code.</p> <p>Launch configurations contain the path to the executable, and that changes for each build configuration.</p>	<p>After you change from Debug to Release or vice-versa, click the “Generate Launches” command in the Quick Panel.</p> <p>You can alternately run the following command to create new launch configurations:</p> <ul style="list-style-type: none"> <li>• <code>make eclipse</code> for Eclipse IDE for ModusToolbox</li> <li>• <code>make vscode</code> for VS Code</li> </ul> <p>Be aware that this option overwrites any changes you may have made to the configurations.</p>
<p>On macOS for a PSoC 64 or PSoC 6A project, using the Eclipse IDE for ModusToolbox, the “Attach” launch configuration fails every second time it is used:</p> <ul style="list-style-type: none"> <li>• for an RTOS-based project</li> <li>• if “Attach” is launched when the target is halted on breakpoint</li> </ul>	<p>Reset the device each time you run the “Attach” configuration.</p>
<p>For the Arm Cortex CM4 core, some code in <code>main()</code> executes before the debugger stops at start of <code>main()</code>. This means that some code executes twice; once before the debugger stops execution, and again after the debugger resets the program counter to the start of <code>main()</code> and you start debugging.</p>	<p>If you observe this issue, and it affects your application, put a delay loop at the start of <code>main()</code> to allow time for debugging subsystem initialization. The following code tests for the presence of the debugger before entering the delay loop:</p> <pre>int main(void) {     /* If an active debugger is detected, give     it some time to execute SYSRESETREQ     * otherwise, the application runs before     the debugger is fully operational */     if (CoreDebug-&gt;DHCSR &amp;         CoreDebug_DHCSR_C_DEBUGEN_Msk)     {         Cy_SysLib_Delay(400u);     } }</pre> <p>See <a href="#">KBA231071</a> for details.</p>
<p>Junk characters might be observed in a UART terminal during programming of the connected kit or after programming is completed.</p>	<p>Clear UART buffers after programming is completed.</p>
<p>On some occasions, the Eclipse IDE will fail to run launch configurations with various errors, such as: “XXX has encountered a problem. Debug session already started. Terminate the first one before restarting”</p>	<p>Depending on the steps taken prior to launching a configuration, there are several reasons this may occur. The easiest way to resolve the issue is to restart Eclipse.</p>
<p>There is a programming error for Cypress platforms that connect via FTDI on macOS Catalina. The boards include:</p> <ul style="list-style-type: none"> <li>• CYW920819EVB-02</li> <li>• CYW920820EVB-02</li> <li>• CYW920719B2Q40EVB-01</li> <li>• CYW920721B2EVK-01</li> <li>• CYW920721B2EVK-02</li> <li>• CYW920721B2EVK-03</li> <li>• CYW989820EVB-01</li> <li>• CYW920706WCDEVAL</li> <li>• CYW920735Q60EVB-01</li> </ul>	<p>This only happens in macOS Catalina 10.15.5 because of a serial port detection error. The macOS Catalina FTDI driver is missing the necessary device identification information.</p> <p>To resolve this issue, update to macOS Catalina 10.15.6 or newer.</p>

Problem	Workaround
A "JTAG-DP STICKY ERROR" message may appear in the IDE when connected via the JTAG interface of a MiniProg4 probe in CMSIS-DAP HID mode.	You can safely ignore this error, or switch the MiniProg4 to CMSIS-DAP bulk mode.
JTAG performance on MiniProg4 may be significantly slower than SWD.	There is no workaround except using the SWD interface if JTAG performance is not acceptable.
You must manually reset after programming PSoC 6 kits when using GDB SEGGER + Jlink + JTAG interface.	Update each of the following Launch Configurations under the <b>Debugger</b> tab. In the Device Name field, delete the "_tm" suffix.  <ul style="list-style-type: none"> <li>• "&lt;app-name&gt; Erase (JLink)"</li> <li>• "&lt;app-name&gt; Program (JLink)"</li> </ul>
Starting from KitProg3 v2.10, when KitProg3 is in CMSIS-DAP Bulk mode, it is not possible to debug and use USB-I2C/SPI bridging (for example, in the CapSense Tuner, Bridge Control Panel) at the same time. This affects Windows OS only. It does not affect Linux or macOS users.	If you would like to use debug and USB-I2C/SPI bridging at the same time, there are two possible workarounds: <ul style="list-style-type: none"> <li>• If performance for programming and debug is not critical, switch KitProg3 to <b>CMSIS-DAP HID</b> mode via the <a href="#">fw-loader</a> utility. Firmware Loader is installed with ModusToolbox software, and is available separately on <a href="#">GitHub</a>.</li> <li>• If you need faster performance for programming and debug, use the onboard KitProg3 for programming purposes and MiniProg4 for bridging purposes or vice versa. Both devices can be in <b>CMSIS-DAP bulk</b> mode.</li> </ul> Details are in <a href="#">KBA231025</a> .
Starting from KitProg3 v2.10, in some cases Windows 7 does not recognize the KitProg3 bridge. So the USB-I2C/SPI bridge devices are not available in either CMSIS-DAP HID or CMSIS-DAP bulk mode.	Install a digitally signed driver manually from the Windows Update Catalog. Follow steps from <a href="#">KBA231026</a> .
In Linux OS, with KitProg3 in CMSIS-DAP HID mode, a debug session in ModusToolbox can be destroyed if you use the Firmware Loader --device-list command while debugging. This is limitation of hidapi library used on Linux. MacOS and Windows OSs are not impacted.	If you have a debug session running, don't use the firmware loader tool.
KitProg3 v2.10 or later is installed as part of the ModusToolbox 2.2 or later tools package. This version of KitProg3 will not work with PSoC Creator 4.3 or PSoC Programmer 3.28.7.	If you updated your kit to KitProg3 v2.10 and wish to use the kit with PSoC Creator 4.3 and PSoC Programmer 3.28.7, get a previous version of fw-loader (with an earlier version of KitProg3) and update the kit. The fw-loader tool is available here: <a href="https://github.com/cypresssemiconductorco/Firmware-loader">github.com/cypresssemiconductorco/Firmware-loader</a>  If this is not urgent, you can wait for a newer PSoC Programmer version with support for KitProg3 v2.10, expected soon.

## Eclipse IDE

Problem	Workaround
<p>Sometimes when importing an application using the Eclipse IDE, a message displays stating the following:</p> <p><i>Unable to add the shared library project at "&lt;workspace_location&gt;/mtb_shared" to the workspace. That project is for reference only; functionality is not affected.</i></p>	<p>This message can be safely ignored.</p>
<p>For some macOS machines running on M1 processors, the Eclipse IDE may fail to start with a message such as:</p> <p><i>ModusToolbox quit unexpectedly.</i></p> <p>The message details also state that the "attachment of code signature supplement failed."</p>	<p>The only solution to this problem at this time is to uninstall and reinstall ModusToolbox.</p> <p>We are investigating this issue for the next release.</p>
<p>In the updated Eclipse IDE for this release, the <b>News</b> tab has been removed. However, it may appear if you open an older release application in the updated IDE. If so, it may contain an error message like:</p> <p>"Unable to access the news site. Please check your internet connection then click on the button below to try again."</p>	<p>The web page that provided content to the <b>News</b> tab is no longer maintained. You can safely ignore the message about site access.</p> <p>If you want to remove the <b>News</b> tab from this workspace, select <b>Window &gt; Perspective &gt; Reset Perspective...</b></p>
<p>Some projects that run pre- or post-build scripts, for example BTSDK or XMC, when imported in the Eclipse IDE on macOS and Linux fail to build if the project archive contains executable files.</p> <p>If an existing project is exported from Eclipse using <b>File &gt; Export &gt; General &gt; Archive File</b>, and then the resultant .zip file is imported into another workspace using <b>File &gt; Import &gt; General &gt; Existing Projects into Workspace</b>, any files that had the executable permission bit set in the original project will have the executable bit stripped in the new imported project.</p> <p>This can vary from project to project, but can include executables necessary to the build process, which can cause the imported project to fail to build, or other possible side effects depending on what executables may be in the archive.</p>	<p>Identify what files were executable in the original project (pre-export) and then manually fix them in the new imported project with the following terminal command:</p> <pre>chmod +x &lt;filename&gt;</pre>
<p>For a ModusToolbox 2.2 application that includes a shared directory, such as mtb_shared, when you use the <b>Build All</b> command you may see an error such as:</p> <pre>**** Build of configuration Debug for project mtb_shared **** make: *** No rule to make target 'all'. Stop. Build Failed. 1 errors, 0 warnings.</pre>	<p>This problem has been addressed for version 2.3 applications and it only occurs for version 2.2 applications with a shared directory.</p> <p>To stop this message from occurring, open the properties dialog for the shared directory (<b>Properties &gt; C/C++ Build</b>) and change the <b>Build command</b> field to the following:</p> <pre>echo Skipping \${ProjName}; nothing to build for:</pre>
<p>For projects that use external resources located outside of the application folder root (such as mtb_shared, wiced_bt sdk, or 3rd-party libraries), some IDE code browsing and analysis features (such as resolving includes or opening object declarations for objects and headers located in those external resources) may not be usable or show unresolved includes immediately after project creation.</p>	<p>These will be resolved after building the project, as the IDE parses the output messages from the build to find paths to those resources.</p>

Problem	Workaround
<p>On macOS Catalina, some IDE GUI windows/elements do not display correctly in some cases.</p>	<p>Resize the window or scroll to fix the display problem. In some cases, use the Refresh option.</p> <p>Cypress is investigating these issues and will address them in a future release.</p>
<p>If you delete the shared library directory (named <i>mtb_shared</i>, by default) from disk and then regenerate it using <code>make getlibs</code> or the Library Manager, the directory will not be restored properly for use in the Eclipse IDE.</p> <p>This is because several files required by the Eclipse IDE are not restored as they were when the application was created.</p> <p><b>Note</b> You can delete the <i>mtb_shared</i> directory at any time because it can be recreated. You might do this when sharing the application, for example. The shared library directory only contains files that are already controlled and versioned, so you should NOT check it into a revision control system.</p>	<p>After regenerating the <i>mtb_shared</i> directory and assorted libraries, open the Eclipse IDE and follow these steps:</p> <ol style="list-style-type: none"> <li>1. Delete the <i>mtb_shared</i> folder shown in the IDE Project Explorer. Do <b>NOT</b> select the check box "Delete project contents on disk" (if you do, you will have to regenerate it again).</li> <li>2. Select <b>File &gt; Import &gt; C/C++ &gt; Existing Code as Makefile Project</b> and click <b>Next &gt;</b>.</li> </ol>  <ol style="list-style-type: none"> <li>a. Under <b>Existing Code Location</b>, click <b>Browse...</b>, navigate to the application's root directory, select the <i>mtb_shared</i> folder, and click <b>Select Folder</b>.</li> <li>b. Under <b>Toolchain for Indexer Settings</b>, select <b>ARM Cross GCC</b>.</li> <li>c. Click <b>Finish</b>.</li> </ol> <ol style="list-style-type: none"> <li>3. After the import completes, build the application.</li> </ol>
<p>Some applications when imported into the Eclipse IDE from Mbed OS fail to build with an error, such as:</p> <pre>cc1.exe fatal error: .mbed_config.h: No such file or directory</pre>	<p>This happens because the application is generating a path to the <i>mbed_config.h</i> header file that make build system cannot find. Fix this by removing the relative path <code>.\</code> in the makefile. Change this:</p> <pre>ASM_FLAGS += -include ASM_FLAGS += .\mbed_config.h</pre> <p>To this:</p> <pre>ASM_FLAGS += -include ASM_FLAGS += mbed_config.h</pre>
<p>On Windows, when building a project and/or programming the device, the IDE reports one or more errors similar to the following:</p> <pre>*** fatal error - cygheap base mismatch detected - 0x18032C408/0x18032D408</pre>	<p>This occurs because you likely have multiple versions of Cygwin in your build environment path. The IDE uses a version of Cygwin in the ModusToolbox installation directory. Remove the instance of "C:\cygwin64\bin" from your path.</p>

Problem	Workaround
<p>Markdown (*.md) files do not render correctly in the IDE. For example, tables do not show rows and columns. Also, the IDE may show an error for the file such as: "Cannot resolve element with id 'figure-1'".</p>	<p>This is a known issue with viewing markdown files in Eclipse. Various rendering errors can be safely ignored. Cypress recommends using an external editor, such as Visual Studio Code or Typora to view markdown files.</p>
<p>When creating a new application, the IDE attempts to open any found readme.md files. Occasionally (and randomly), some of these files are opened in an external text editor instead of in the IDE.</p>	<p>This appears to be an Eclipse bug with no workaround. However, since *.md files do not render well in Eclipse (as noted above), you should use an external editor, such as Visual Studio Code or Typora, and set that editor as the default.</p>
<p>Sometimes, the Eclipse "egit" plugin locks directories. This prevents the Library Manager from removing BSPs/libraries from these locked directories. When this happens, you will see an error message in the Library Manager console indicating that permission is denied for removing a particular directory.</p>	<p>Before running the Library Manager from the Eclipse IDE, right-click on the project and select <b>Team &gt; Disconnect</b>. When you are done with the Library Manager, go to <b>Team &gt; Share Project</b> and select the correct project to reconnect.</p> <p><b>Note</b> Various projects may be set up differently, and the process to use the Team options will vary as well.</p>
<p>The IDE <b>Project &gt; Build Configurations &gt; Active</b> menu item to set Debug or Release mode is non-functional. This is due to the project's Makefile.</p>	<p>To set Debug or Release mode, edit the project's Makefile, which contains the following:</p> <pre># Default build configuration. Options include: # # Debug -- build with minimal optimizations, focus on debugging. # Release -- build with full optimizations CONFIG=Debug</pre>
<p>If you include external folders/files in your application, the Eclipse IDE will occasionally unselect your application project in the Project Explorer. However, it will leave the <b>Launch</b> section populated with links in the Quick Panel. Using those Launch links may result in a reported error of "Could not resolve cy_prj_path."</p>	<p>To resolve this, click on the appropriate project for your application in the Project Explorer, and then click on the <b>Launch</b> link again.</p>

## Library Manager

Problem	Workaround
<p>Issue using a VPN.</p>	<p>See <a href="#">Project Creator</a>.</p>
<p>After running the Library Manager, the files shown in the Eclipse IDE Project Explorer are not updated.</p>	<p>To see the updated files, click on the project in the IDE Project Explorer and press [F5] to refresh the Eclipse view.</p>
<p>Opening multiple instances of the Library Manager at the same time for the same application could cause confusion and indeterminate states in applications.</p>	<p>Do not open multiple instances of the Library Manager for the same application. This issue will be addressed in a later release.</p>
<p>Sometimes when you click <b>Update</b> in the Library Manager, the tool will begin to process your request and then appear to freeze.</p>	<p>Terminate the Library Manager and relaunch it. This issue will be addressed in a future release.</p>

Problem	Workaround
<p>When you remove a library item, the Library Manager deletes the associated directory from the “libraries” directory (typically this is called “libs”). On Windows, if there are any processes that have a lock on that directory, or a file in that directory, the directory removal will not work completely. The Library Manager will remove most of the directory contents and will also mark the library item as removed.</p>	<p>To remove the library completely, you must release the lock on the folder or file, and then manually delete the directory. The steps to release the lock depend entirely on the process that is holding the lock. Common scenarios include:</p> <ul style="list-style-type: none"> <li>• A command-line prompt is in that directory. In this case, “cd” to a different directory.</li> <li>• A text editor has a file from that directory open. In this case, close the file in the text editor. Depending on the text editor, you may have to exit the entire text editor.</li> <li>• The Eclipse git “egit” plugin has a lock on the folder. In this case, exit and restart the ModusToolbox Eclipse IDE.</li> </ul> <p>In all cases, once the lock is removed you must remove the associated directory from the libraries directory before the Library Manager will be able to work with that particular library again.</p>

### Device Configurator

Problem	Workaround
<p>The analog router may route across pins that are not intended to be routed (either because they are digital inputs, or because the project routes them in software). This can cause signal corruption or other connection problems.</p>	<p>Use the Analog Route Editor to manually route around such pins, if necessary.</p>

### CapSense Tuner

Problem	Workaround
<p>If using the UART communication interface, a low packet-transferring rate may cause the CapSense Tuner to disconnect due to a data-reading timeout error.</p>	<p>You can resolve this using either of these solutions:</p> <ul style="list-style-type: none"> <li>• Increase the UART communication baud rate.</li> <li>• Increase the value of the <code>uartSingleReadTimeout</code> parameter available in the tuner INI file.</li> </ul> <p>You can find the tuner INI file as follows:</p> <ul style="list-style-type: none"> <li>• Windows: <code>&lt;user_home&gt;/AppData/Roaming/Cypress Semiconductor Corporation/CapSense Tuner.ini</code></li> <li>• Linux: <code>/home/&lt;user_home&gt;/config/Cypress Semiconductor Corporation/CapSense Tuner.ini</code></li> <li>• macOS: <code>/Users/&lt;user_home&gt;/config/Cypress Semiconductor Corporation/CapSense Tuner.ini</code></li> </ul>

## Documentation

Problem	Workaround
Various documents included with the release may contain incomplete information, or may not contain up-to-date screen captures or information.	New versions of documents, including this release notes document, may be available online at: <a href="https://www.cypress.com/products/modustoolbox-software-environment">https://www.cypress.com/products/modustoolbox-software-environment</a>

## Open Source

Portions of this software package are licensed under free and/or open source licenses such as the GNU General Public License. Such free and/or open source software is subject to the applicable license agreement and not the Cypress license agreement covering this software package. The applicable license agreements are available online:

<http://www.cypress.com/documentation/software-and-drivers/free-and-open-source-software-download-page>

## Further Reading

There are several related documents provided with ModusToolbox software. These documents include (but are not limited to):

- [ModusToolbox Installation Guide](#)
- [ModusToolbox User Guide](#)
- [Cypress Programmer Release Notes](#)

Other documentation includes (but is not limited to):

- Device Datasheets
- Application Notes
- Training

Contact your Cypress representative, as needed.





Cypress Semiconductor  
An Infineon Technologies Company  
198 Champion Ct.  
San Jose, CA 95134-1709 USA  
[www.cypress.com](http://www.cypress.com)  
[www.infineon.com](http://www.infineon.com)

© Cypress Semiconductor Corporation (an Infineon company), 2017-2021. This document is the property of Cypress Semiconductor Corporation and its subsidiaries, including Spansion LLC ("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. 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. Cypress products are not designed, intended, or authorized for use as critical components in systems designed or intended for the operation of weapons, weapons systems, nuclear installations, life-support devices or systems, other medical devices or systems (including resuscitation equipment and surgical implants), pollution control or hazardous substances management, or other uses where the failure of the device or system could cause personal injury, death, or property damage ("Unintended Uses"). A critical component is any component of a device or system whose failure to perform can be reasonably expected to cause the failure of the device or system, 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 or related to all Unintended Uses of Cypress products. You shall indemnify and hold Cypress harmless from and against all claims, costs, damages, and other liabilities, including claims for personal injury or death, arising from or related to any Unintended Uses of Cypress products.

Cypress, the Cypress logo, Spansion, the Spansion logo, and combinations thereof, ModusToolbox, 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](http://cypress.com). Other names and brands may be claimed as property of their respective owners.