

ModusToolbox™ 2.0 Release Notes

Production Release

Overview

ModusToolbox™ software is a set of tools that enable you to integrate Cypress devices into your existing development methodology. This release is an update to ModusToolbox 1.1. It does not replace the existing version; it installs alongside it. This document describes the features and known limitations for the ModusToolbox software included in this release.

Contents

Overview	1
Contents	1
What's Changed.....	2
What's Included.....	2
ModusToolbox IDE	3
Configurators and Tools	3
Utilities	4
Design Impact	4
Updating your Project from ModusToolbox 1.1	4
Supported Tool Chains	6
Supported Boards	7
Known Issues/Limitations.....	7
Installation.....	7
Programming/Debugging.....	8
ModusToolbox IDE	8
Documentation.....	10
Device Configurator	10
Library Manager.....	11
Open Source	11
Further Reading	11

What's Changed

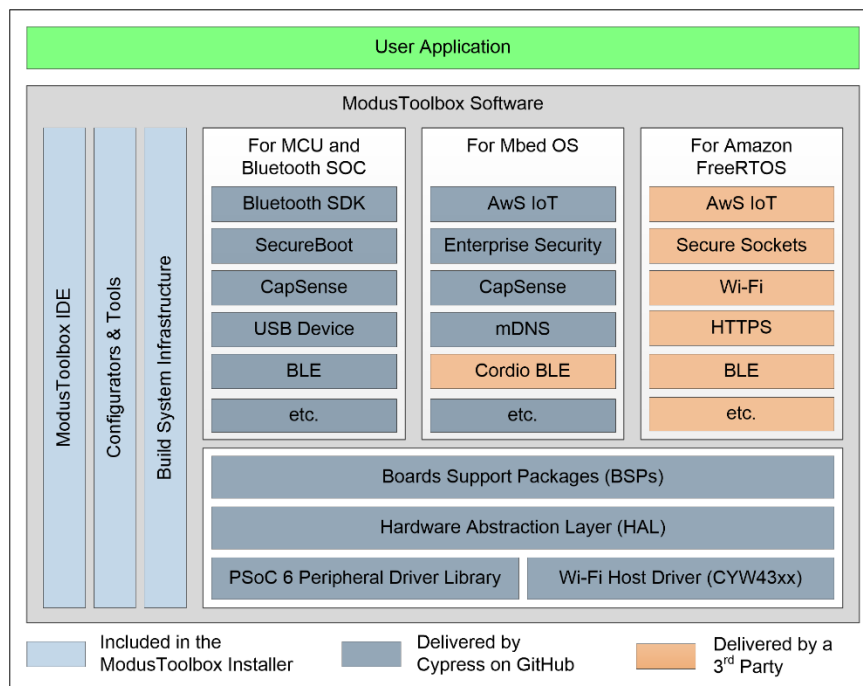
In conjunction with the ModusToolbox 1.1 release, Cypress began the process of moving away from an all-in-one application development solution. Many pieces of software and code examples were made available through Mbed OS and GitHub. With ModusToolbox 2.0, Cypress is expanding on that process by removing the software libraries installed on disk, as well as including Amazon FreeRTOS. That process includes the following high-level changes between ModusToolbox 1.1 to 2.0:

- New directory structure
- Updated Configurators
- Source code acquired from GitHub; not bundled with the IDE
- New / simpler project structure:
 - •Command-line builds and IDE builds are 1-to-1 identical
 - •Simpler, faster make-based build system

Note ModusToolbox 2.0 is not backward compatible with the 1.1 release. You must create a new application and add the necessary content from the old project. See [Updating your Project from ModusToolbox 1.1](#) for more details.

What's Included

The main pieces of the ModusToolbox software installation include the ModusToolbox IDE, Configurators and Tools, and the build system infrastructure. Various parts of the ModusToolbox software are available on GitHub and from 3rd party sources. The following diagram shows a high-level structure of the ModusToolbox software.



Note Some pieces of Cypress Programmer are included as part of command line tools with the ModusToolbox software. You can also install Cypress Programmer, including a GUI, separately.

ModusToolbox IDE

This is an Eclipse-based IDE that facilitates creating applications and adding additional Cypress firmware to them. There are several parts of the ModusToolbox IDE, including:

- **ModusToolbox Perspective** – This is a plugin that provides ease-of-use features in the IDE, including the Quick panel and debug mode.
- **New Application Wizard** – This is an Eclipse-specific wizard used for creating new applications for Cypress devices.
- **Code Editor** – This is the standard Eclipse code editing environment.
- **Context Menus:**
 - **design.modus:** File used to open the Device Configurator is now included with BSPs. The menu item (or Quick Panel item) will find and open the correct file – even when there are more than one BSP in your workspace.
 - **Library Manager** – This is dialog is used for transitioning your application to include/remove assorted kits, as well as adding/removing libraries.

Configurators and Tools

Configurators are GUI-based tools provided as part of the installation. Each can also be run using the command line. In general, you interact with these tools from the ModusToolbox IDE. However, these tools can also be used separately, and they can be run from the command line. Each of these tools provides a guide, available from the tool's **Help** menu.

- **Device Configurator:** Set up the system (platform) functions, as well as the basic peripherals (e.g., UART and Timer, etc.).
- **CapSense Configurator and Tuner:** Configure CapSense, test it, and generate the required firmware.
- **USB Configurator:** Configure USB settings and generate the required firmware.
- **QSPI Configurator:** Configure external memory and generate the required firmware.
- **Bluetooth Configurator:** Configure Bluetooth settings and generate the required firmware.
- **Smart I/O™ Configurator:** Configure Smart I/O settings and generate the required firmware.
- **SegLCD Configurator:** Configure a generic LCD Direct Segment Drive controller for a variety of LCD glass.
- **Project Creator:** A stand-alone new project wizard to create projects when not using the ModusToolbox IDE.
- **MCU Elf Tool:** Merges core executable images into a single file for programming / debugging.
- **Power Estimator Tool:** Provides an estimate of power consumed by a target device (CYW920819EVB_02 kit only).
- **Device Firmware Update Host Tool:** Tool to update a PSoC® 6 MCU firmware that has previously been programmed with an application that supports device firmware update capability.

Utilities

The installation also includes the following utilities:

- OpenOCD: Open On-Chip Debugger provides debugging, in-system programming, and boundary-scan testing for embedded target devices.
- Cypress KitProg Firmware Loader: Used to upgrade firmware on Cypress kits. It allows you to switch the KitProg firmware from KitProg2 to KitProg3, and back.
- JRE: Java Runtime Environment. Required for Eclipse plugin and Configurators.
- GCC: This is the default compiler for the firmware application.
- Make: Includes scripts for the build system.
- Modus-Shell: This includes a version of Cygwin (for Windows only) that provides basic Unix style commands. This includes make, bash, diff, and many others.
- Configurator Backend Support: Command-line tools used by the Configurators in the background.

Design Impact

This section includes issues that might impact current designs.

Updating your Project from ModusToolbox 1.1

As mentioned under [What's Changed](#), the application file structure has been updated in ModusToolbox 2.0. That means that you cannot directly open a ModusToolbox 1.1 application in ModusToolbox 2.0. However, you can create a new application and copy/paste the relevant source code from the old application to the new application. There are separate instructions for PSoC 6 MCU applications and WICED Bluetooth Applications.

Note Both versions of the ModusToolbox IDE use the same default folder for the local user workspace, which is 'mtw' in the user home directory. When launching the ModusToolbox 2.0 IDE, choose an alternate workspace location for 2.0 projects. Do not share the same default workspace with ModusToolbox 1.1 projects.

PSoC 6 MCU Application

The following is an example to update a CapSense application for the CY8CKIT-062-WIFI-BT kit:

1. Create a new Hello World application using the ModusToolbox 2.0 IDE, and then build it. Refer to the *Quick Start Guide* (**Help > ModusToolbox IDE Documentation > Quick Start Guide**) for more details.
2. Copy the application source code (*main.c*) from your ModusToolbox 1.1 application, and replace the code in the ModusToolbox 2.0 application.
3. Save and exit the ModusToolbox 2.0 IDE.
4. Copy the *design.modus* file from the ModusToolbox 1.1 application, and paste it over the *design.modus* file in the ModusToolbox 2.0 application, located as follows:

```
<wkspc_name>/<app_name>/libs/TARGET_CY8CKIT-062-WIFI-BT\COMPONENT_BSP_DESIGN_MODUS/design.modus
```
5. Reopen the new application in the ModusToolbox 2.0 IDE, and then open the Device Configurator. The Notice List should include several "Fixes" that need to be addressed.

- a. Click **File > Update All Personalities** to address all the messages about older versions on personalities.
 - b. For the remaining messages, click each **Fix** cell to address the specific issues (for example, drive strength, clocks, etc.).
6. Save the Device Configurator to generate updated code; there should be no errors.
7. Launch the CapSense Configurator, and make sure all the pins are assigned appropriately.
8. Save the CapSense configuration and exit the CapSense Configurator.
9. Exit the Device Configurator.
10. In the IDE, select **Project > Clean** and allow the tool to build automatically. It should build without any errors.

WICED Bluetooth Application

The following is an example to update a WICED Bluetooth application for the CYW920819EVB-02 kit:

1. Create a new 'wiced_btSDK' library project (pre-requisite to any WICED application project) using the **New Application** wizard in the Quick Panel. Refer to the *Quick Start Guide (Help -> ModusToolbox IDE Documentation -> Quick Start Guide)* for more details. Note this step takes several minutes to complete while it downloads all necessary BT SDK collateral.
2. Run the **New Application** wizard again, select the CYW920819EVB-02 kit, and the 'Empty-20819EVB02' application project to create the new 2.0 project.
3. Copy the main application source code (*empty_wiced_bt.c* for example, or any other customized application source files, including Bluetooth configuration from *GeneratedSource/cycfg_gatt_db.c/h files*) from your ModusToolbox 1.1 application, and replace or merge the code in the ModusToolbox 2.0 application.
4. Merge any project settings that were customized in the 1.1 application, for example CY_APP_DEFINES. Only settings that were changed from defaults in 1.1 need to be merged to 2.0 settings.

Most custom settings are contained in the 1.1 *modus.mk* file. However, if any settings were manually changed in the IDE **C/C++ Build -> Settings** UI (from right click project -> **Properties**), they should be added to the appropriate makefile variable in the 2.0 application makefile, for example: INCLUDES, DEFINES, CFLAGS, LDFLAGS, etc.

All 2.0 application settings are contained in the application *makefile* only. Edit the makefile as needed to transfer the settings from 1.1 to 2.0. In general, if a setting from a 1.1 *modus.mk* file does not exist in the 2.0 *makefile* it is not needed. However, some settings have changed names from 1.1 to 2.0 as follows:

1.1 Setting	2.0 Setting
CY_EXAMPLE_NAME	APPNAME
PLATFORM	TARGET
CY_VALID_PLATFORMS	SUPPORTED_TARGETS
APP_XIP	XIP
CY_MAINAPP_SWCOMP_USED	COMPONENTS (see note)

The CY_MAINAPP_SWCOMP_USED is used in 1.1 with full path names for middleware libraries. In 2.0, the COMPONENTS variable needs only the name of the library. For example, if the 1.1 *modus.mk* file contained:

```
CY_MAINAPP_SWCOMP_USED=bt_20819A1-1.0/components/BT-
SDK/common/libraries/anc_lib
```

Then the 2.0 *makefile* should use:

```
COMPONENTS+=anc_lib
```

If the 1.1 *modus.mk* file contained a `FEATURE_VALUES` setting, this is a list of `VARIABLE/VALUE` pairs that should be entered directly as make variables with values into the 2.0 *makefile*. For example, if the 1.1 *modus.mk* file contained:

```
FEATURE_VALUES=BT_DEVICE_ADDRESS,default UART,AUTO
```

Then the 2.0 *makefile* should use:

```
BT_DEVICE_ADDRESS=default
```

```
UART=AUTO
```

The 1.1 setting for `TOOLCHAIN` used value 'GCC', but all 2.0 projects should use default setting of 'GCC_ARM'.

5. Save and exit the ModusToolbox 2.0 IDE.
6. Copy the *design.modus* file from the ModusToolbox 1.1 application, and paste it over the *design.modus* file in the ModusToolbox 2.0 'wiced_btsdk' project, located as follows:

```
<wkspc_name>/wiced_btsdk/dev-kit/bsp/TARGET_CYW920819EVB-02\COMPONENT_bsp_design_modus/design.modus
```
7. Reopen the new application in the ModusToolbox 2.0 IDE, select the application in Project Explorer, and then open the Device Configurator via the ModusToolbox Quick Panel under Tools. The Notice List may include several "Fixes" that need to be addressed.
 - a. Click **File > Update All Personalities** to address all the messages about older versions on personalities.
 - b. For the remaining messages, click each **Fix** cell to address the specific issues (for example, drive strength, clocks, etc.).
8. Save the Device Configurator to generate updated code; there should be no errors.
9. Exit the Device Configurator.
10. In the IDE, select the project, and in the ModusToolbox Quick Panel, click the 'Clean' and then 'Build' entries. It should build without any errors.

Supported Tool Chains

The GCC Arm Embedded toolchain GCC 7.2.1 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 (Windows and Linux hosts)
- IAR Embedded Workbench v8.2 (Windows only)

Supported Boards

At the time of this installer release, it supports the following boards:

Board	MCU	Connectivity
CY8CKIT-062-BLE	CY8C6347BZI-BLD53	On-chip Bluetooth
CY8CKIT-062-WIFI-BT	CY8C6247BZI-D54	CYW4343W Wi-Fi + Bluetooth
CY8CPROTO-062-4343W	CY8C624ABZI-D44	CYW4343W Wi-Fi + Bluetooth
CY8CKIT-062S2-43012		CYW43012 Wi-Fi + Bluetooth
CYW9P62S1-43012EVB-01		CYW43012 Wi-Fi + Bluetooth
CY8CPROTO-063-BLE		On-chip Bluetooth
CYBT-213043-EVAL	CYW20819	On-chip Bluetooth
CYBT-213043-MESH	CYW20819	On-chip Bluetooth
CYW920719B2Q40EVB-01	CYW20719	On-chip Bluetooth
CYW920721B2EVB-01	CYW20721	On-chip Bluetooth
CYW920721B2EVB-03	CYW20721	On-chip Bluetooth
CYW920819EVB-02	CYW20819	On-chip Bluetooth
CYW920820EVB-02	CYW20820	On-chip Bluetooth
CYW989820EVB-01	CYW20820	On-chip Bluetooth

Note After the installer has been released, additional boards will be supported on an ongoing basis.

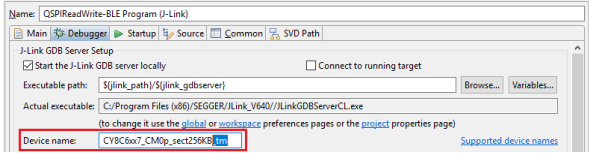
Known Issues/Limitations

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

Installation

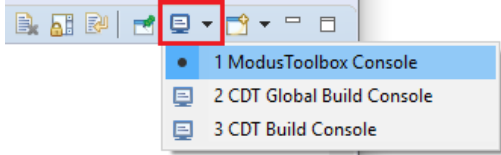
Problem	Workaround
On common Linux distributions, the serial UART ports (usually /dev/ttySx or /dev/ttyUSBx devices) belongs to the root user and to the dialout and plugdev groups. Standard users are not allowed to access these devices.	<p>An easy way to allow the current user access to the Linux machine's 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>Note For this command to take effect, the user must log out and then log back in.</p>
When trying to create a new project, you may see the following error message: <i>Unable to open file at</i> http://github.com/cypresssemiconductorco/mtb-super-manifest/raw/v2.X/mtb-super-manifest.xml . Some boards and apps may be missing. Check the logfile for a detailed error message.	<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 and you must set Preferences > General > Network Connections > Active Provider to "Manual" and add HTTP and HTTPS to the "Proxy entries" on that same preference screen.</p>

Programming/Debugging

Problem	Workaround
<p>OpenOCD (KitProg3) launch configurations for secure CY8CPROTO-064-SB applications sometimes may not halt at the 'main' function during debugging. This limitation applies to:</p> <ul style="list-style-type: none"> command line 'make debug' IDE 	<p>To resolve this:</p> <ul style="list-style-type: none"> command line: Remove or comment out 'mon gdb_sync' and 'stepi' commands in the <i>gdbinit_secure</i> file located at <code><project_path>/libs/psoc6make/make/scripts</code>. IDE: Remove 'mon gdb_sync' and 'stepi' under Run/Restart Commands in the Startup tab of the appropriate launch configuration (Debug or Attach).
<p>On some occasions, the 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 the Cypress platforms that connect via FTDI on MacOS Catalina. The boards include:</p> <ul style="list-style-type: none"> CYW920819EVB-02 CYW920820EVB-02 CYW920719B2Q40EVB-01 CYW920721B2EVK-01 CYW920721B2EVK-03 CYW989820EVB-01 	<p>This only happens in MacOS Catalina, and it is due to a serial port detection error. This happens because the MacOS Catalina does not include the AppleUSBFTDI.kext driver.</p> <p>There is no known workaround for this problem for the ModusToolbox 2.0 release. A Knowledge Base Article will be published as soon as a solution to this problem is found or resolved in MacOS Catalina.</p>
<p>User must manually reset after programming PSoC 6 kits when using GDB SEGGER + Jlink + JTAG interface.</p>	<p>Update each of the following Launch Configurations under the Debugger tab. In the Device Name field, delete the "_tm" suffix.</p>  <ul style="list-style-type: none"> "<app-name> Erase (JLink)" "<app-name> Program (JLink)"

ModusToolbox IDE

Problem	Workaround
<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:</p> <p>"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>After creating a new application in the ModusToolbox IDE, you may notice that the Quick Panel "Device Configurator" link also displays the text "(new configurator)".</p>	<p>There is no workaround. Although that text is inaccurate, the Device Configurator opens with the correct configuration file when you click the link. This link text will be addressed in a future release.</p>
<p>Creating a new application takes a long time, and the IDE does not seem to be doing anything.</p>	<p>The new application process is performing a <code>git clone</code> operation, and it can take a few minutes to complete. Make sure the ModusToolbox Console is selected before creating a new application to view the messages displayed while the process is running.</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 to remove a particular directory.</p>	<p>Before running the Library Manager from the ModusToolbox IDE, right-click on the project and select Team > Disconnect. When you are done with the Library Manager, go to Team > Share Project and select the correct project to reconnect.</p> <p>Note Various projects may be set up differently, and the process to use the Team options will vary as well.</p>
<p>The IDE Build Configuration 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 ModusToolbox IDE will occasionally unselect your application project in the Project Explorer. However, it will leave the Launch 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 Launch link again.</p>

Problem	Workaround
<p>When trying to create a project from a private git repo, the git clone process will hang with a message similar to:</p> <p><i>Cloning into 'Hello_World'.... "</i></p> <p>The process hangs because git is waiting for a username and password to be entered. There is no way to enter this information in the new project creation process.</p>	<p>git has a mechanism for storing authentication credentials. See the following websites for more information:</p> <ul style="list-style-type: none"> • https://git-scm.com/docs/gitcredentials.html • https://git-scm.com/book/en/v2/Git-Tools-Credential-Storage
<p>When trying to create a new project you may see the following error message: Unable to open file at http://github.com/cypresssemiconductorco/mtb-super-manifest/raw/v2.X/mtb-super-manifest.xml. Some boards and apps may be missing. Check the logfile for a detailed error message.</p>	<p>See the instructions in the Installation problem section above concerning proxy settings.</p>

Documentation

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

Device Configurator

Problem	Workaround
<p>In the Device Configurator, a port cannot be connected to more than one mux at the same time (for example, P10[0] can be connected to SAR Ch0 Vplus or SAR Ch0 Vminus, but not both).</p>	<p>There is no workaround at this time. This will be addressed in a future release.</p>
<p>When you open the Device Configurator using a ModusToolbox 1.1 application, a message displays indicating the device currently being used is not supported.</p> <p>This is expected because the following devices were made obsolete and do not work for ModusToolbox 2.0:</p> <ul style="list-style-type: none"> • CY8C624ABZI-D04 • CY8C624ABZI-D14 • CY8C624AAZI-D14 • CY8C6248AZI-D14 • CY8C6248BZI-D44 • CY8C6248AZI-D44 • CY8C6248FNI-D43 	<p>These devices are not supported in ModusToolbox 2.0; therefore, you cannot use version 2.0 of the Device Configurator with them.</p> <p>Click OK to close the message and then click Cancel on the subsequent dialog.</p> <p>After that, close the version 2.0 Device Configurator, and go back to your ModusToolbox 1.1 application to select a device that is supported in ModusToolbox 2.0. Once that is complete, you will be able to open the version 2.0 Device Configurator for that design.</p>

Library Manager

Problem	Workaround
Multiple instances of the Library Manager open at the same time could cause confusion and indeterminate states in applications.	Do not open multiple instances of the Library Manager. This issue will be addressed in a later release.
Sometimes when you click Apply in the Library Manager, the tool will begin to process your request and then appear to freeze.	Terminate the Library Manager and relaunch it. This issue will be addressed in a future release.
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>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"> • A command-line prompt is in that directory. In this case, "cd" to a different directory. • 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. • The Eclipse git "egit" plugin has a lock on the folder. In this case, exit and restart the ModusToolbox Eclipse IDE. <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>

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
- Cypress Programmer Release Notes
- PSoC 6 Software API Reference Guides
- Bluetooth API Documentation

- ModusToolbox IDE Quick Start Guide
- ModusToolbox IDE User Guide
- ModusToolbox Software Overview
- ModusToolbox Configurator Guides (for each Configurator)

Other documentation includes (but is not limited to):

- Device Datasheets
- Application Notes
- Training

Contact your Cypress representative, as needed.

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

© Cypress Semiconductor Corporation, 2017-2019. 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 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. Other names and brands may be claimed as property of their respective owners.

This software is based, in part, on the FreeType2 font engine from the FreeType Project.