ModusToolbox™ tools package release notes

ModusToolbox™ tools package version 3.8.0

A newer version of this document may be available on the web here.

About this document

Scope and purpose

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

ModusToolbox™ software is a set of tools that enable you to integrate our 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™ tools package user guide. See also What's included in this document.

This ModusToolbox™ 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™ tools package user guide, "Product versioning" section.

Reference documents

Refer to the following documents for more information as needed:

- ModusToolbox™ tools package user guide
- ModusToolbox™ software installation guide
- Training material on GitHub

Table of contents

1 Primary changes

The overall tools package release includes the following updates and features. See also the separate release notes for the Setup program and the Programmer tools package.

1.1 Support for ARM macOS

Apple has stopped support for Intel-based processors for macOS, and we too no longer support Intel-based processors for ModusToolbox™. We are now providing support for ARM-based processors.

1.2 Usability improvements for ARM MDK (Keil µVision)

We have made significant improvements for the export of applications to Keil µVision. The process generates a post-build file that reduces the manual configuration required to build and program.

1.3 Device Configurator improvements

- The stand-alone Infineon Device Configurator is now available for macOS and Linux. Previously, it had been for Windows only.
- We have added a warning pop-up message to tell users that the Device Configurator is about to regenerate code and over-write any manual edits that were made to the previously generated code.
- There is a new feature that saves all the settings of a resource, so if you disable that resource and later re-enable it, all the previous settings are restored.

1.4 Improved programmer integration with Eclipse

There is a new "launch" item in the Quick Panel called **Advanced KitProg3 Programming**, which is used to launch the ModusToolbox™ Programmer with the device and a HEX file already selected and configured for you. It also passes a number of parameters for configuring QSPI, sflash, and other features specific for different device families.

You must build your application before using this item.

Note: *In addition to the tools included with this package, there was an update to Eclipse for ModusToolbox available through the Setup program, plus updates to documentation in HTML available on the web here: https://documentation.infineon.com/modustoolbox/*

2 What's included

This release includes the following tools and versions:

Tool Name	Current Release	Previous Release
AIROC™ tools	5.0.1 (no change)	5.0.1
Bluetooth® Configurator	3.60.0	3.50.0
BSP Assistant	1.70.0	1.60.0
cymcuelftool	1.0 (no change)	1.0
Dashboard	3.8.0	3.7.0
Device Configurator	5.70.0	5.60.0
Device Firmware Update (DFU) Host Tool	2.120.0	2.100.0
EZ-PD™ Configurator	2.40.0	2.30.0
GNU make Build System (tools-make)	2.7.0	2.6.1
LCS Manager	1.60.0	1.50.0
Library Manager	2.70.0	2.60.0
LIN Configurator	1.90.0	1.80.0
memoryreport	1.1.0	1.0.0
modus-shell	1.5.0 (no change)	1.5.0
Core build infrastructure: • mtbarchive • mtbgetlibs • mtblaunch • mtbninja • mtbquery • mtbsearch	1.8.0	1.7.0
mtb-settings	1.30.0	1.20.0
mtbideexport	1.6.0	1.5.0
ninja	1.12.1 (no change)	1.12.1
Project Creator	2.70.0	2.60.0
Proxy Helper	1.100.0	1.90.0
QSPI Configurator	4.80.0	4.70.0
Segment LCD Configurator	1.110.0	1.100.0
SignCombineMkGen	1.3.0	1.2.0
Smart I/O Configurator	4.70.0	4.60.0
SRecord	1.64 (no change)	1.64
USB Configurator	3.20.0	3.10.0

What's included

2.1 Supported host operating systems

OS	Supported	Recommended (full testing)
Windows	11	**11**
macOS (Intel Macs)	not supported	**not supported**
macOS (ARM Macs)	Sonoma, Sequoia, Tahoe	**Tahoe**
Linux	22.04, 24.04 LTS, 25.10	**24.04 LTS**

2.2 Supported toolchains

The GCC Arm Embedded toolchain GCC 11.3 and GCC 14 are provided as separate packages to install via the Setup program in addition to the ModusToolbox™ tools package. These toolchains have no use restrictions and do not require license activation (it is distributed under the terms of the GNU Public License).

The ModusToolbox™ system also supports these toolchains for most applications and devices (see the application *README.md* file for applicable support) that you can install separately:

- Arm compiler v6 or newer (Windows and Linux hosts) (developer.arm.com)
- IAR Embedded Workbench v9 or newer (Windows only) (iar.com)
- LLVM 19.1.5 or newer (github.com/ARM-software/LLVM-embedded-toolchain-for-Arm/)

2.3 Supported boards

The 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/SOC/SIP	Connectivity
AIROC™ Bluetooth® BSPs		
AIROC™ Connectivity BSPs		
CYW943907AEVAL1F	CYW43907KWBG	<none>
CYW954907AEVAL1F	CYW54907KWBG	<none>
CYW955513EVK-01	CYW55513IUBG	CYW55513IUBG
CYW955913EVK-01	CYW55913IUBGT	CYW55913IUBGT
PMG BSPs		
PSOC™ 4 BSPs		
PSOC™ 6 BSPs		
PSOC™ Control BSPs		
KIT_PSC3M5_2GO	PSC3M5FDS2LGQ1	<none>
KIT_PSC3M5_CC1	PSC3M5FDS2AFQ1	<none>
KIT_PSC3M5_CC2	PSC3M5FDS2AFQ1	<none>
KIT_PSC3M5_EVK	PSC3M5FDS2AFQ1	<none>
PSOC™ Edge BSPs		
KIT_PSE84_AI	PSE846GPS2DBZC4A	LBEE5HY2FY (CYW55513IUBG)
KIT_PSE84_EVAL_EPC2	PSE846GPS2DBZC4A	CYW55513IUBG
KIT_PSE84_EVAL_EPC4	PSE846GPS4DBZC4A	CYW55513IUBG
Reference Design BSPs		
TRAVEO™ BSPs		
KIT_T2G_C-2D-6M_LITE	CYT4DNJBZS	<none>
KIT_T2G-B-E_LITE	CYT2BL5CAS	<none>
KIT_T2G-B-H_EVK	CYT4BFBCHE	<none>
KIT_T2G-B-H_LITE	CYT4BF8CDS	<none>
USB BSPs		
KIT_FX20_FMC_001	CYUSB4024-BZXI	<none>
KIT_FX2G3_104LGA	CYUSB2318-BF104AXI	<none>
Wireless Charging BSPs		
WLC1115-68LQXQ	WLC1115-68LQXQ	<none>
WLC1150-68LQXQ	WLC1150-68LQXQ	<none>
XMC™ BSPs		

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

2.4 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 our license agreement covering this software package. The applicable license agreements are available online:

https://www.infineon.com/cms/en/design-support/software/free-and-open-source-software-foss/modustoolbox-foss-packages/

3 Known issues, limitations, and workarounds

This section describes the known issues and limitations of this release, and provides workarounds for them:

3.1 ModusToolbox™ issues from previous releases

This document contains only recent issues pertinent to ModusToolbox™ version 3.x. All issues noted in previous ModusToolbox™ version 2.x releases have been made available online here: KBA236147.

3.2 Proxy

Problem	Workaround
When trying to use any of the ModusToolbox™ tools, you may see an error message similar to the following about not able to connect to the Internet: *Unable to open file at [URL]*	This can happen if you are behind a firewall and do not have your proxy settings configured. You must set your proxy server settings using the ModusToolbox™ Settings tool, available from any tool with a **Settings** menu. In previous versions of the ModusToolbox™ tools package, these types of errors only affected the Library Manager and Project Creator. In version 3.x, these errors apply to all tools.

3.3 Device database error

Problem	Workaround

Various tools report an error with the device-db similar to this:
`fatal:'C:/test/ModusToolbox/packs/ModusToolbox-Early-Access-Pack/libs/device-db' does not appear to be a git repository`
`fatal: Could not read from remote repository.`
	Remove the .modustoolbox/global directory from user home and remove the mtb-shared directory from your application. Then, retry the failing operation.

3.4 Toolchain issues

Problem	Workaround
Any toolchain installed in a directory using Linux reserved characters such as < > | \ : ( ) & ; [for example Program Files (x86)], will be ignored by the ModusToolbox™ GNU make system. The GNU make system uses the format $(wilcard) to detect if a given path exists, and thus is not compatible.	Make sure you install desired toolchains in directories that do not use reserved characters or spaces. Use the appropriate toolchain DIR variable to specify the location of the toolchain (for example `CY_COMPILER_GCC_ARM_DIR=C:/arm-gnu-toolchain-12.2.rel1`) so that is it detected by the GNU make system.

3.5 Project Creator

Problem	Workaround
If you have Project Creator and ModusToolbox™ Programmer open at the same time, Project Creator might not detect a connected kit.	If you need to be able to detect a kit, open only one of the tools at a time.
Project Creator may fail to create a project with an error like: `"ERROR: <PATH>: Failed to open Git repository: config value 'safe.directory' was not found"` where <PATH> corresponds to the project to be created. This can happen if you choose to create your project on a non-NTFS USB drive or a shared folder whose owner differs from the current user running the Project Creator tool.	This error is an intentional behavior of Git that is being used by the Project Creator under the hood. This behavior is due to the known security problem CVE-2022-24765. If the USB drive is non-NTFS, such as exFAT/FAT, or if a shared folder owned by someone else, add the directory to the safe.directory setting in Git.

3.6 LCS Manager CLI

Problem	Workaround
The LCS Manager CLI tool does not use the ModusToolbox™ proxy settings that other tools use. Depending on your system environment, the LCS Manager CLI tool might not be able to access and download data from external repositories (such as GitHub).	Ensure that your system can access the desired external repositories by setting up proxy information before using the LCS Manager CLI tool. If you need assistance, contact your IT department.

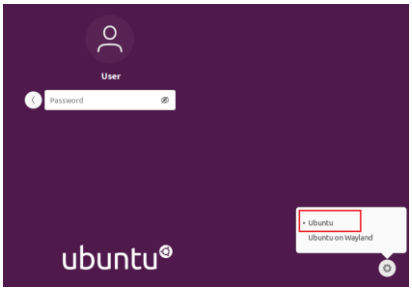
3.7 HTML/documentation on Ubuntu 22.04

Problem	Workaround
On Ubuntu 22.04, sometimes HTML documentation links do not work anywhere. This issue can occur as documentation not opening at all or opening in the wrong program (e.g., Notepad, Office Writer, etc.)	Reset the default browser from the Ubuntu 22 Settings UI (**Settings > Default Applications > Web**). Changing this setting via the BROWSER environment variable, or the xdg-settings/xdg-mime commands does not work.
On Ubuntu, cannot open documentation in a web browser with file not found error. This applies to using the Dashboard or Eclipse "Help" menu, even though the documentation files are present and accessible.	This issue is related to the security settings for snap packages installed on the system. We recommend replacing the snap-based browser with a deb package version.

3.8 Eclipse IDE

Eclipse issues are included in the Eclipse release notes included in that installation.

3.9 Visual Studio Code

Problem	Workaround
In in VS Code, when debugging multi-threaded applications, such as those using FreeRTOS, clicking the Restart icon causes the new main thread to be assigned the next available thread ID instead of starting from ID 1.	No workaround is available because this behavior is specific to the GDB client.
Various ModusToolbox™ GUI tools (Device Configurator, Library Manager, etc.) fail to start on Ubuntu when executed from the Terminal window in VS Code, with the following error: Failed to load client buffer integration: "wayland-egl" Available client buffer integrations: () No shell integration named "xdg-shell" found No shell integration named "xdg-shell-v6" found No shell integration named "wl-shell" found No shell integration named "ivi-shell" found Loading shell integration failed. Attempted to load the following shells ("xdg-shell", "xdg-shell-v6", "wl-shell", "ivi-shell")	**Solution 1**: Open an Ubuntu Terminal, navigate to the application directory, and use the applicable "make" command to open the GUI (e.g., `make library-manager`). **Solution 2**: Log out from the current Ubuntu session, and then log back in and make sure Ubuntu on Wayland is **not** selected on the login screen.  On newer versions of Ubuntu, the login screen will show "Ubuntu on Xorg".

3.10 Building/programming/debugging

Problem	Workaround
During application programming (at the verification step) into the NVM using the JTAG interface, the following error might occur: "Error: Failed to read memory at <mem_address>."	Try lowering the probe JTAG clock by adding MTB_PROBE_FREQUENCY_KHZ=<clock_val_in_khz> to the bsp.mk file. For example: MTB_PROBE_FREQUENCY_KHZ=2000

In VS Code for an XMC7200 application, the **Restart** button does not work when using OpenOCD versions 5.6 – 5.7.	Open the *scripts\target\infineon\cat1\func_mxs40.cfg* script file and replace the following:
	`"} elseif [string match "cat1c" $target ] {"`
	
	With this:
	`"} elseif [string match "cat1c*" $target ] {"`

Restarting a XMC7000 debug session using MiniProg4 in JTAG mode sometimes might lead to the situation when VS Code assumes target is running and not halting at `main()`. When you pause, the IDE refreshes its state and shows the target stopped at the beginning of the `Reset_Handler()`. Subsequent resuming should pause at `main()`. Very rarely, the debug session might be unresponsive when trying to pause the running target.	To work around this issue, retry restarting again. We recommend using SWD instead of JTAG, which will reduce the number of issues you encounter.

Known issues, limitations, and workarounds

Problem	Workaround

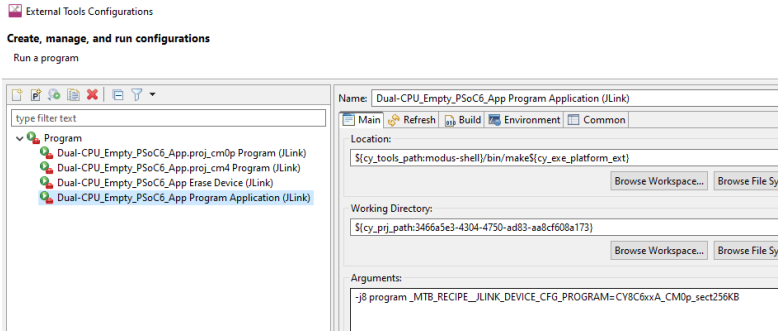
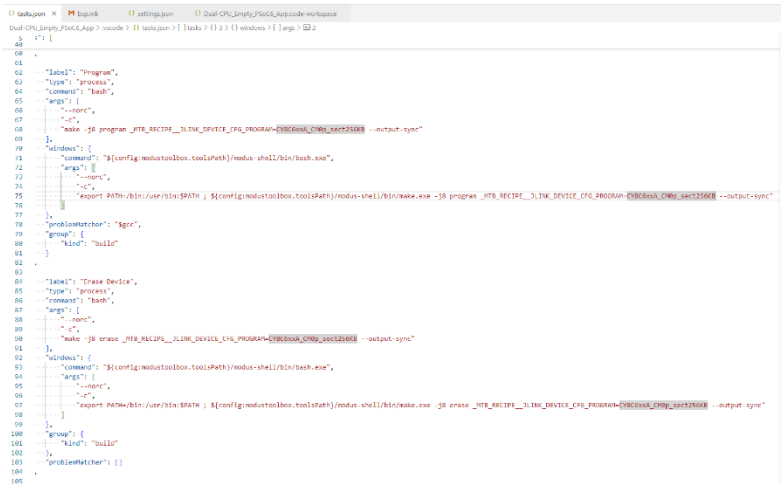
When debugging in Eclipse IDE or VS Code reaches a code with empty loops, the Step operation might cause the debugger to wait indefinitely.
This problem happens because GDB executes single steps and waits until the execution reaches an instruction related to another line in the source code. Since empty loops have no terminating condition, GDB will wait forever, leading to the hang of the debugging process.
	To avoid this problem, consider the following recommendations:
	• Avoid using Step operations in empty loop debugging. Instead of using the Step operation, use the Continue operation to bypass the empty loop.
	• Alternatively, you can place an operation that produces a CPU instruction within the loop body. For instance, adding a `__NOP()` instruction as shown below can resolve the issue.
	`for ( ; ; )`
	`{`
	`  __NOP();`
	`}`

The MTB_PROBE_SERIAL variable is ignored for AIROC™ CYW55913 devices.	Eclipse 1. For KitProg3 - follow section **5.1.11.2 Selecting by serial number** 2. For J-Link - follow section **5.1.12 Select Specific J-Link Device** VS Code 1. For KitProg3 add "*adapter serial xxxxxxxxxxxxxxx*" command to the *openOCDLaunchCommands* property of Attach configuration, where xxxxxxxxxxxxxxx - serial number of the probe 2. For JLink add new property "serialNumber": "50129901", to Attach configuration, where 50129901- serial number of the JLink probe

```
// When using 'attach', make sure your program is running on the board and that your executable matches
// the image in the chip exactly, or else strange things can happen with breakpoint, variables, etc.
{
    "name": "Attach Debug (KitProg3_MiniProg4)",
    "type": "cortex-debug",
    "request": "attach",
    "cwd": "${workspaceFolder}",
    "executable": "./build/APP_CYW955513EVK-01/Debug/mtb-example-threadx-empty-app.elf",
    "servertype": "openocd",
    "searchDir": [
        "${workspaceFolder}",
        "${config:modustoolbox.toolsPath}/openocd/scripts/"
    ],
    "configFiles": [
        "interface/kitprog3.cfg",
        "target/cyw55500.cfg"
    ],
    "openOCDLaunchCommands": [
        "adapter serial 1117108B0237400",
        "proc before_examine_proc { } {cyw55500.dap apreg 0x10000 0xD04 0xe000edf0; cyw55500.dap apreg 0
    ],
```

```
// When using 'attach', make sure your program is running on the board and that your
// executable matches the image in the chip exactly, or else strange things can happen
// with breakpoints, variables, etc.
{
    "name": "Attach Debug (JLink)",
    "type": "cortex-debug",
    "request": "attach",
    "cwd": "${workspaceFolder}",
    "executable": "./build/APP_CYW955513EVK-01/Debug/mtb-example-threadx-empty-app.elf",
    "servertype": "jlink",
    "device": "Cortex-M33",
    "interface": "jtag",
    "serialNumber": "50129901",
    "overrideAttachCommands": [
        "set *0xE000EDFC=(*0xE000EDFC|0x10000)" // Set DEMCR.MON_EN
    ],
    "overrideRestartCommands": [
    ],
    "showDevDebugOutput": "none"
},
```

Known issues, limitations, and workarounds

Problem	Workaround
You must manually reset after programming PSOC™ 6 kits when using GDB SEGGER + Jlink + JTAG interface.	Eclipse: Update Erase/Program launch configs to redefine internal make variable '*MTB_RECIPE__*JLINK_DEVICE_CFG_PROGRAM' with the target JLink alias without _tm suffix (e.g., CY8C6xxA_CM0p_sect256KB):  VS Code: Update Program/Erase tasks with the same information. 
In VS Code, "Failed to update peripheral TCPWM0: Error: peripheral-viewer: readMemory failed" error might appear if you open the "XPERIPHERALS" window when the MCU is in run state and the plugin cannot read peripheral registers on runtime.	To avoid this error, open the "XPERIPHERALS' window when you on a on breakpoint or some instruction.
Exception has occurred" error might appear when you execute Attach/Debug launch configuration. Such errors started to be visible in VS Code - 1.53.2.	You can safely ignore this error.
Existing application opened in ModusToolbox™ version 3.1 or later reports "error: unknown type name 'uint8_t'" or similar compiler errors. This error is caused by the update to GCC 11.	Add '#include <stdint.h>' in all affected source files. This issue only applies to applications migrated to 3.1 or later. It will not occur for new applications created with 3.1 or later tools.

Known issues, limitations, and workarounds

Problem	Workaround
When calling `make program` with `-j` flag on the command line in multi-core applications, the programming processes get out of sync. This results in the qprogram starting before the build finishes. There is no error message reported. This issue does not exist when programming from Eclipse IDE or VSCode.	Since this issue only exists in a scenario in which the user calls `make program` with `-j` flag at the application level in multi-core applications, the current workaround is to run `make build -j` and then run `make qprogram`. This defect is being addressed in the next release.
In the Eclipse IDE, programming a device is skipped when the device was previously programmed.	This is expected behavior and there is no workaround. This was a change made to the launch configurations in Eclipse to support multi-core debugging.
If you change the MCU/SOC/SIP for the BSP using the BSP Assistant, the Register View may not available while debugging using the Eclipse IDE or VS Code.	This is because the svd file path is not present in the launch configurations. To fix this, navigate to the application folder and run make getlibs. The, run make eclipse or make vscode, as applicable.
The **Attach** launch config does not work for the secure lifecycle of AIROC™ CYW20829 devices.	Do not use the **Attach** launch config for secure lifecycle.
The CySecuretool does not work with PyOCD as the debugger on macOS M1 CPUs.	No workaround; we do not plan to support PyOCD.
When exporting ModusToolbox™ applications created for the TRAVEO™ T2G Body High MC devices to be used with the µVision and IAR IDEs, there are some limitations using the J-Link probe (including ETM traces, program, and debug via the CM7 core).	Use native Arm and IAR probes to work with TRAVEO™ T2G Body High MC devices.
There is an issue with reset for TRAVEO™ T2G Body High MC and T2G Instrument Cluster devices in the SEGGER tool. The **Debug** launch config does not work properly for CM7_0 and CM7_1 cores via the J-Link probe in Eclipse and VS Code IDEs.	To debug the code, use the **Attach** launch config.
While using the Eclipse IDE on Windows for various program/debug operations, there's a plugin issue that prevents the debug port from shutting down. This could result in abnormal power consumption, the watchdog timer being blocked, or the inability to connect in JTAG mode after a successful connection in SWD mode.	Reset the device. For example, on the CY8CKIT-062S2-40312 kit, press the **SW1/XRES** button.

3.11 Library Manager/make getlibs

Problem	Workaround
The Library Manager does not list the retarget-io library for the CY8CKIT-040T.	The retarget-io library is not available currently for this device. The retarget-io library depends on mtb-hal-cat2, which is not supported by the CY8CKIT-040T.
A .mtb file added as a direct dependency to the *deps* directory is overridden and removed during the `make getlibs` operation.	This problem occurs when you add a direct dependency using the 'latest-x.y' tag instead of a specific 'release-x.y.z' tag when the BSP or another library requires the same library as an indirect dependency. To ensure your direct dependency is kept, use the 'release-x.y.z' tag in your .mtb file.

3.12 BSP Assistant

Problem	Workaround
When BSP assistant is run from the command line to create your BSP, it does not create the BSP_COMPONENTS variable in the Makefile.	Use the BSP assistant GUI to create your BSP.
When creating a BSP by selecting MPNs, you may see the message "Errors exist in the project's configuration" above the message that states the BSP was created successfully.	If this happens, run the Device Configurator from inside the BSP Assistant and fix all issues in the Notice List before using the BSP.
If you edit an older BSP and change the UDB SDIO port using the updated BSP Assistant 1.10, you need to manually edit the config/cyreservedresources.list file. The older BSP Assistant tool warned of this change. The updated BSP Assistant tool makes this change for you, but only for BSPs created with the newer version of the tool. Not editing the config/cyreservedresources.list file will lead to indeterminant results.	Manually edit the config/cyreservedresources.list file as described, or use only the BSP Assistant version 1.0 tool to make changes to older BSPs.

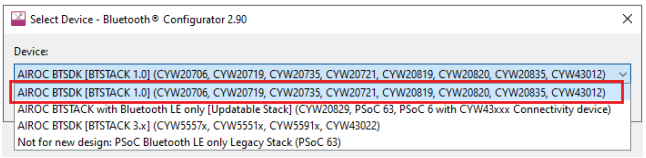
3.13 Device Configurator

Problem	Workaround
The freeRTOS and abstraction-rtos libraries do not support tickless sleep for the PSOC™ Control C3 device. Therefore, deep-sleep cannot be used.	None. This feature will be supported in the next release. Regular sleep is supported and set as the default.
For ModusToolbox™ 2.x applications using PSOC™ 4 devices, if you open the application using ModusToolbox™ version 3.x, the Device Configurator (version 4.0) may show various errors with various types of connections, such as SAR ADC SOC Input, TCPWM Start/Stop/Capture/Count/Reload Signal, DMA Channel Trigger Input or connections between pins over Digital Input and Digital Output.	Select new signals for the invalid connections to resolve the issues.

Known issues, limitations, and workarounds

Problem	Workaround
For ModusToolbox™ 2.x applications using PSOC™ 4 or PSOC™ 6 devices, if you open the application using ModusToolbox™ version 3.x, the Device Configurator (version 4.0) may show "Resource overused" errors in some cases where the pin's Digital Input, Digital Output, and Analog parameters are set at the same time.	Select new signals for the invalid connections to resolve the issues.
If one of the WICED Radio Interface personalities is instantiated in the Universal Digital Block (UDB), it will not allow picking connections for the DMA signals.	The WICED Radio Interface personalities should not be used in practice. Instead, to enable communication with an external radio device from a PSOC™ 6S1 device, the BSP/Application should use the udb-sdio-whd library and set the appropriate component for the desired port. See the README.md file contained in the udb-sdio-whd library for more details. Additionally, to reserve the appropriate resources so they do not get overused by the configurator, the board should include a cyreservedresources.list file next to the design.modus file. Refer to one of the existing PSOC™ 6S1 boards (eg: CY8CKIT-062-WIFI-BT, CYW9P62S1-43012EVB-01, CYW9P62S1-43438EVB-01, …) for what this file should contain.

3.14 Bluetooth® Configurator

Problem	Workaround
The Bluetooth® Configurator does not automatically detect the correct device for the application when creating a new configuration. This issue is applicable to AIROC™ BTSDK [BTSTACK 1.0] devices; for example, CYW20706.	Manually select the first option "AIROC BTSDK [BTSTACK 1.0] (devices)" in the Bluetooth® Configurator Select Device dialog. 

3.15 DFU Host tool

Problem	Workaround
While trying to program the PSOC™ Edge device, increasing the packet size may be unstable and cause failures.	Try to increase timeouts or reduce the packet length.

3.16 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: ModusToolbox™ Software website

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

Edition 2026-03-23

Published by

Infineon Technologies AG

81726 Munich, Germany

Do you have a question about this document?

Email: erratum@infineon.com

Document reference

002-22557 Rev. AL

Important notice

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

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

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

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

Warnings

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

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