8-Bit MCU SDK 4.1.7.0 GA

19Q3 8051 SDK

September, 2019

The 8051 SDK provides infrastructure support for applications developed on 8-bit devices, and it provides interfaces with the underlying hardware. It is composed of the following modules:

- 8-Bit Device Header Files
- 8-Bit Peripheral Driver Libraries
- Sample Applications/Examples for 8-Bit Development Kits

This document covers the following SDK version:

8051 SDK 4.1.7.0 released September, 2019

KEY FEATURES

- VCPXpress 2.0 release
- Removed application note code from SDK
- Replaced compiler_defs.h with si_toolchain.h

Compatibility and Use Notices

If you are new to the Silicon Labs 8-bit SDK, see Using This Release.

Compatible Compilers:

- Keil v9.53
- Keil v9.56

Contents

1 New Items

1.1 New Device Headers

None

1.2 New Peripheral Driver Libraries

None

1.3 New Sample Applications

None

2 Improvements

2.1 Updated Device Headers

Replaced all references to compiler_defs.h with si_toolchain.h

2.2 Updated Peripheral Driver Libraries

Replaced all references to compiler_defs.h with si_toolchain.h

Released VCPXpress version 2.0

- Added a ZLP after writes of multiple of 64-bytes
- The callback right before the ZLP will not trigger a user callback
- Updated Block_Write() to output the correct number of bytes transmitted
- Added VCP_Set_Suspend() API command to set the new user-configured suspend enable in run-time
- Added VCP_Get_Suspend() API command to get the new user-configured suspend enable status in run-time
- Added VCP_Suspend() which was previously called USB_Suspend() which did not work and did not manually suspend the device
- Removed the option to store variables in USB FIFO
- Allocated VCP variable without hard-coding the locations
- Updated VCPXcore_CP210x_Get_Eventstate() to return 2 bytes instead of 1
- Updated VCPXcore_CP210x_Embed_Events() to return error indication if wValue is not in the range 0x00 - 0xFF
- Updated VCPXcore_CP210x_Set_Char() to not fail for certain indexes
- Updated VCPXcore_CP210x_Set_Char() to fail when passed an invalid index
- Updated VCPXpress.h and docs
- Removed the useFifo variable from the initialization struct. This may require removing the useFifo variable in the user's application
- Replaced callback function pointer with a user-defined VCP_Callback(). This may require adding VCP_Callback in the user's application

2.3 Updated Sample Applications

Replaced all references to compiler_defs.h with si_toolchain.h

Updated VCPXpress_Echo (in all VCPXpress-enabled devices)

- Removed callback function pointer
- Removed option to store vcpx variables in USB FIFO

Updated VCPXpress_UsbToUart (in all VCPXpress-enabled devices)

- Removed callback function pointer
- Removed option to store vcpx variables in USB FIFO

3 Fixed Issues

The table below lists issues resolved in the latest release.

ID #	Description
363329	Fixed incorrect version numbers in cslib.h
	Fixed a data overflow issue in EFM8UB1's VCPXpress_USBToUart example project
422385	Watch Dog Timer resets in certain Space Invaders demo projects

4 Known Issues in the Current Release

The table below lists known issues in the latest release. Items shown in blue are links to additional information.

ID #	Description	Workaround
408543	Missing or corrupted .hwconf files in SB1 capsense examples	The project source code is included in the example and can be modified without the use of the hardware configurator.
355966	Dropped characters on multiple calls to UART1_WriteBuffer()	Insert a short delay between any two bufferWrite calls
354781	Missing autopaging in the efm8_memory_lcd library	Insert SFRPAGE save and restore
344029	Missing autopaging in UART1_writeBuffer()	Insert SFRPAGE save and restore

5 Deprecated Items

None

6 Removed Items

The deprecated **VCPXpress_API_Callback** macro definition in the VCPXpress library has been removed.

The directory, **/an,** has been removed from the 8-bit SDK as it contained duplicate copies of all 8-bit application note examples. These code examples are still available through Simplicity Studio.

7 Using This Release

7.1 Installation and Use

The 8-Bit SDK can be installed through Simplicity Studio. Installation instructions can be found in AN1211.

Use the 8-bit SDK with the Simplicity Studio V4 development platform. Simplicity Studio ensures that most software and tool compatibilities are managed correctly. Install software and board firmware updates promptly when you are notified.

Documentation specific to the SDK version is installed with the SDK. API references and other information about this and earlier releases is available on http://devtools.silabs.com/studio/doc/EFM8/software/.

7.2 Support

Development Kit customers are eligible for training and technical support. You can use https://www.silabs.com/products/mcu/8-bit to obtain information about all Silicon Labs 8-bit products and services, and to sign up for product support.

You can contact Silicon Laboratories support at http://www.silabs.com/support.

8 Legal

8.1 Disclaimer

Silicon Labs intends to provide customers with the latest, accurate, and in-depth documentation of all peripherals and modules available for system and software implementers using or intending to use the Silicon Labs products. Characterization data, available modules and peripherals, memory sizes and memory addresses refer to each specific device, and "Typical" parameters provided can and do vary in different applications.

Application examples described herein are for illustrative purposes only.

Silicon Labs reserves the right to make changes without further notice and limitation to product information, specifications, and descriptions herein, and does not give warranties as to the accuracy or completeness of the included information. Silicon Labs shall have no liability for the consequences of use of the information supplied herein. This document does not imply or express copyright licenses granted hereunder to design or fabricate any integrated circuits. The products are not designed or authorized to be used within any Life Support System. A "Life Support System" is any product or system intended to support or sustain life and/or health, which, if it fails, can be reasonably expected to result in significant personal injury or death. Silicon Labs products are not designed or authorized for military applications. Silicon Labs products shall under no circumstances be used in weapons of mass destruction including (but not limited to) nuclear, biological or chemical weapons, or missiles capable of delivering such weapons.

8.2 Trademark Information

Silicon Laboratories Inc.® , Silicon Laboratories®, Silicon Labs®, SiLabs® and the Silicon Labs logo®, Bluegiga®, Bluegiga Logo®, Clockbuilder®, CMEMS®, DSPLL®, EFM®, EFM32®, EFR, Ember®, Energy Micro, Energy Micro logo and combinations thereof, "the world’s most energy friendly microcontrollers", Ember®, EZLink®, EZRadio®, EZRadioPRO®, Gecko®, ISOmodem®, Micrium, Precision32®, ProSLIC®, Simplicity Studio®, SiPHY®, Telegesis, the Telegesis Logo®, USBXpress®, Zentri, Z-Wave and others are trademarks or registered trademarks of Silicon Labs.

ARM, CORTEX, Cortex-M3 and THUMB are trademarks or registered trademarks of ARM Holdings.

Keil is a registered trademark of ARM Limited. All other products or brand names mentioned herein are trademarks of their respective holders.