User's Guide
SLAU799A–January 2019–Revised April 2019

Programming the Bootloader of MSP430™ and SimpleLink™ MSP432™, CC13xx, and CC26xx MCUs Using UniFlash

Uniflash is a stand-alone tool that can program the on-chip flash memory on TI MCUs and on-board flash memory for Sitara processors. Uniflash has a GUI, command line, and scripting interface. For a full description of UniFlash, see the Uniflash Standalone Flash Tool page.

This user's guide describes how to use UniFlash with MSP430™ microcontrollers (MCUs) and select SimpleLink™ MCUs.

Contents

1 Introduction 2
2 Preparing the Image 2
3 Supported Devices 3
4 GUI Default Settings 3
5 Bootloader Programming of MSP430 MCUs 4
5.1 Programming the Firmware Image to the Target Bootloader 4
5.2 Reading the Memory of the Target Bootloader 7
6 Bootloader Programming for SimpleLink MSP432P4 MCUs 8
6.1 Programming the Firmware Image to the Target Bootloader 8
6.2 Reading the Memory of the Target Bootloader 10
7 Bootloader Programming for SimpleLink MSP432E4 MCUs 12
7.1 Programming the Firmware Image to the Target Bootloader 12
8 Bootloader Programming for SimpleLink CC13xx and CC26xx MCUs 14
8.1 Handling the CCFG Configuration Under Firmware Image 14
8.2 Programming the Firmware Image Into the Target Bootloader 15
8.3 Reading the Memory of the Target Bootloader 17
9 Related Documents 18

List of Figures

1 Device Selection 3
2 Select the Device MSP430FR2355 4
3 Enter the File of password.txt and the Firmware Image of Blink LED 4
4 Content of password.txt 4
5 Content of blink_2355.txt 5
6 Setting COM Port Number in the Settings & Utilities Tab 5
7 Wrong Password Execution 6
8 Correct Password Execution Followed by Successful Programming 6
9 Concatenation of the Bytes Into Password Format 7
10 Enter the Configuration on Read Section 7
11 Console View for Reading the Memory Successfully 7
12 Select the Device MSP432P4111 8
13 Loading a File Using the 256-Byte Default Password 8
14 Setting COM Port Number in the Settings & Utilities Tab 9
15 Sending Wrong Password Case 9

Copyright © 2019, Texas Instruments Incorporated

16 Console View for Programming the Data Block Successfully With the 256-Byte Default Password 10
17 Loaded Image on the Target Device 10
18 Concatenation of the Bytes Into Password Format 10
19 Enter the Configuration on Read Section 11
20 Console View for Reading the Memory Successfully 11
21 Select the Device MSP432E401Y 12
22 Enter the Firmware Image 12
23 Configure the COM Port and Use the Default Settings for Other Configurations 12
24 Console View of Downloading the UART Bootloader Flash-Based Application 13
25 Uncheck "Apply auto baud rate for speed initialization" 13
26 Select the Firmware Image of Blink Application 13
27 Console View of Downloading the Blink Application 14
28 Apply "Exclude from Build" for ccfg.c 14
29 Location Where CCFG is Programmed and Needs to be Deleted 15
30 Select Device CC2652R1F 15
31 Load the Image 16
32 Configure the COM Port and Use the Default Settings for Other Configuration 16
33 Console View of Downloading the pwmled Application 16
34 Image Valid Address Configuration 17
35 Enter the Configuration on Read Section 17
36 Console View for Reading the Memory Successfully 17

Trademarks

MSP430, SimpleLink, Code Composer Studio are trademarks of Texas Instruments.
IAR Embedded Workbench is a registered trademark of IAR Systems.
All other trademarks are the property of their respective owners.

1 Introduction

This document describes how to use UniFlash to program the bootloader of the following MCUs.

UniFlash 4.6.0 supports:

- MSP430 microcontrollers
- SimpleLink MSP432P4 microcontrollers
- SimpleLink MSP432E4 microcontrollers

UniFlash 5.0.0 adds support for:

- SimpleLink CC13xx microcontrollers
- SimpleLink CC26xx microcontrollers

2 Preparing the Image

Bootloader programming supports the following firmware image file formats:

- TI TXT file (.txt)
- Intel Hex file (.hex)
- Binary (.bin)

These file formats are generated by tools such as Code Composer Studio™ IDE and IAR Embedded Workbench® IDE.

Copyright © 2019, Texas Instruments Incorporated

3 Supported Devices

In UniFlash, devices with bootloader support are listed with the suffix (BOOTLOADER) in the device selection (see Figure 1). The "Serial" label means that the supported protocol is serial communication, such as UART, I^2C, or SPI. The "On-Chip" label means that the debug programming is already available in UniFlash.

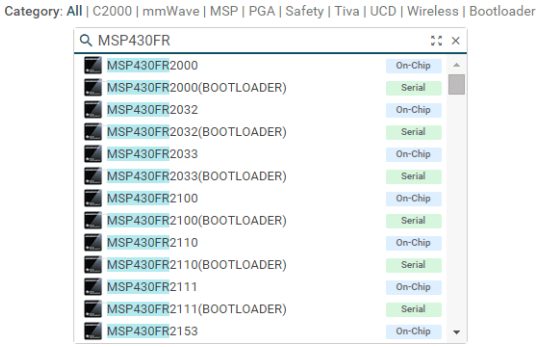

Figure 1. Device Selection

4 GUI Default Settings

The GUI has been configured with default values. Therefore, the only configurations required are selecting the firmware image to download and the COM port number that is connected to the target device.

Copyright © 2019, Texas Instruments Incorporated

5 Bootloader Programming of MSP430 MCUs

5.1 Programming the Firmware Image to the Target Bootloader

For the MSP430 MCUs, the following example downloads a blink LED application to an MSP430FR2355 MCU.

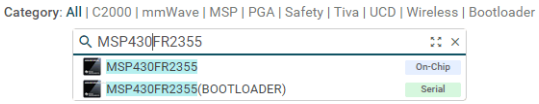

Figure 2. Select the Device MSP430FR2355

The Program tab displays three text fields for the firmware image. The Password field is to load the image of the bootloader password, to unlock the bootloader before the communication is established. The password is the first 32 bytes of data starting at memory address 0xFFE0h, where the interrupt vector is located. For more information about the password for the bootloader, see the MSP430™ Flash Device Bootloader (BSL) User's Guide or the MSP430™ FRAM Device Bootloader (BSL) User's Guide.

When the Password field is empty, the default password is used during execution. The default password is 0xFF for all 32 bytes.

This example uses the default password, which is stored in the file password.txt.

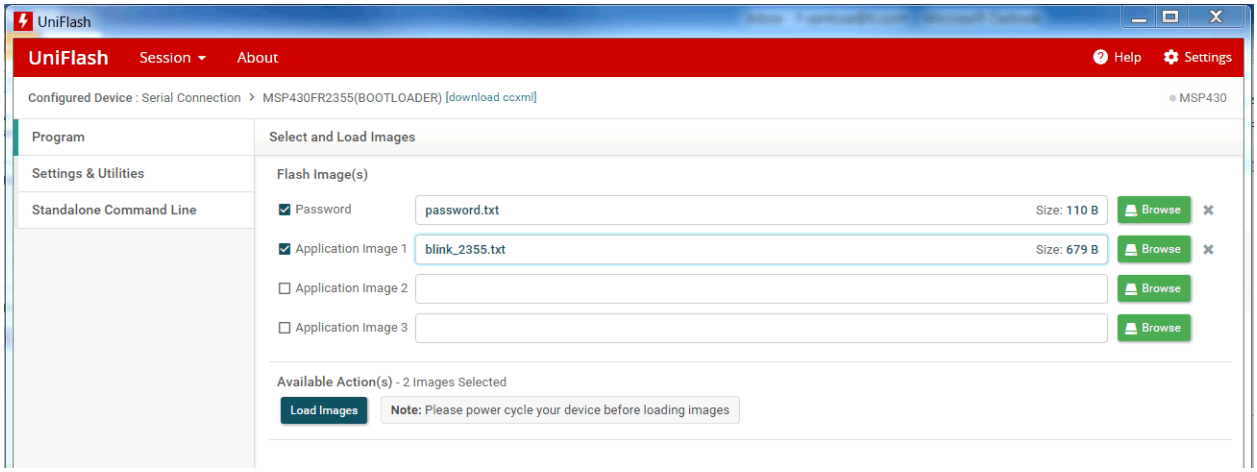

Figure 3. Enter the File of password.txt and the Firmware Image of Blink LED

For the following example, Figure 4 shows the content of password.txt, and Figure 5 shows the content of blink_2355.txt.

```
1  @ffe0
2  ff ff ff ff ff ff ff ff ff ff ff ff ff ff ff ff
3  ff ff ff ff ff ff ff ff ff ff ff ff ff ff ff ff
4  q
```

Figure 4. Content of password.txt

Copyright © 2019, Texas Instruments Incorporated

```
@8000
31 80 06 00 3E 40 00 00 3E F0 3F 00 81 4E 00 00
3F 40 01 00 1F F3 81 4F 02 00 3D 40 01 00 1D F3
81 4D 04 00 5E 06 5F 02 0F DE 1F D1 04 00 3F D0
00 A5 82 4F 60 01 31 50 06 00 10 01 21 83 B2 40
80 5A CC 01 92 C3 30 01 D2 D3 04 02 D2 E3 02 02
B1 40 10 27 00 00 91 83 00 00 81 93 00 00 F6 27
FA 3F 03 43 03 43 FF 3F 03 43 1C 43 10 01 31 40
00 30 B0 13 00 80 B0 13 6A 80 0C 43 B0 13 3C 80
1C 43 B0 13 64 80 32 D0 10 00 FD 3F 03 43
@ff80
FF FF FF FF FF FF FF FF FF FF FF FF
@ffa0
FF FF
@ffce
86 80 86 80 86 80 86 80 86 80 86 80 86 80 86 80
86 80 86 80 86 80 86 80 86 80 86 80 86 80 86 80
86 80 86 80 86 80 86 80 86 80 86 80 86 80 86 80
6E 80
q
```

Figure 5. Content of blink_2355.txt

The minimum requirement in the Settings & Utilities tab is the COM port and correct protocol.

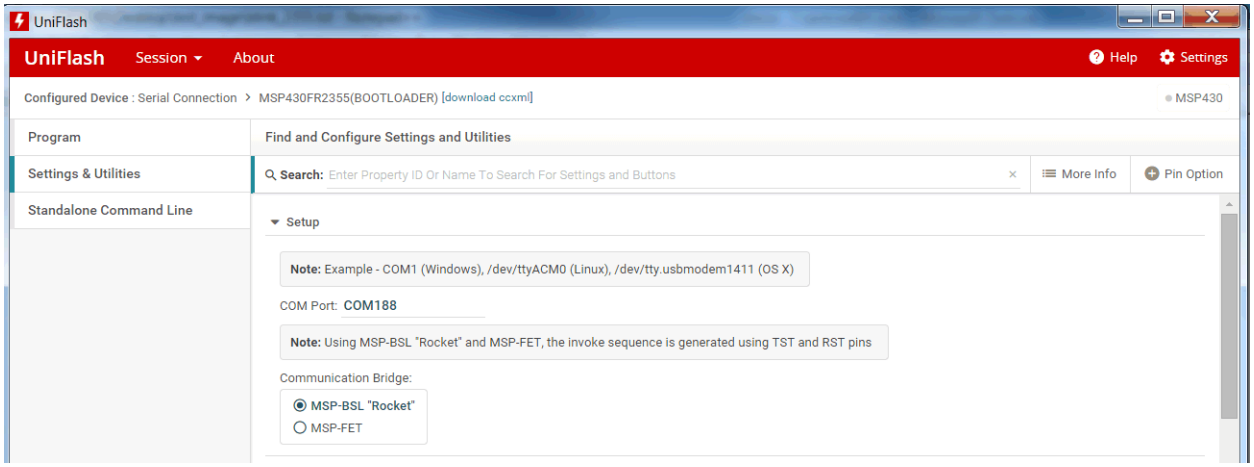

Figure 6. Setting COM Port Number in the Settings & Utilities Tab

Copyright © 2019, Texas Instruments Incorporated

If the device was programmed and the interrupt vector sector is not empty, an error will be returned with status "BSL Password is incorrect!" (see Figure 7). Sending a wrong password triggers a mass erase on the device.

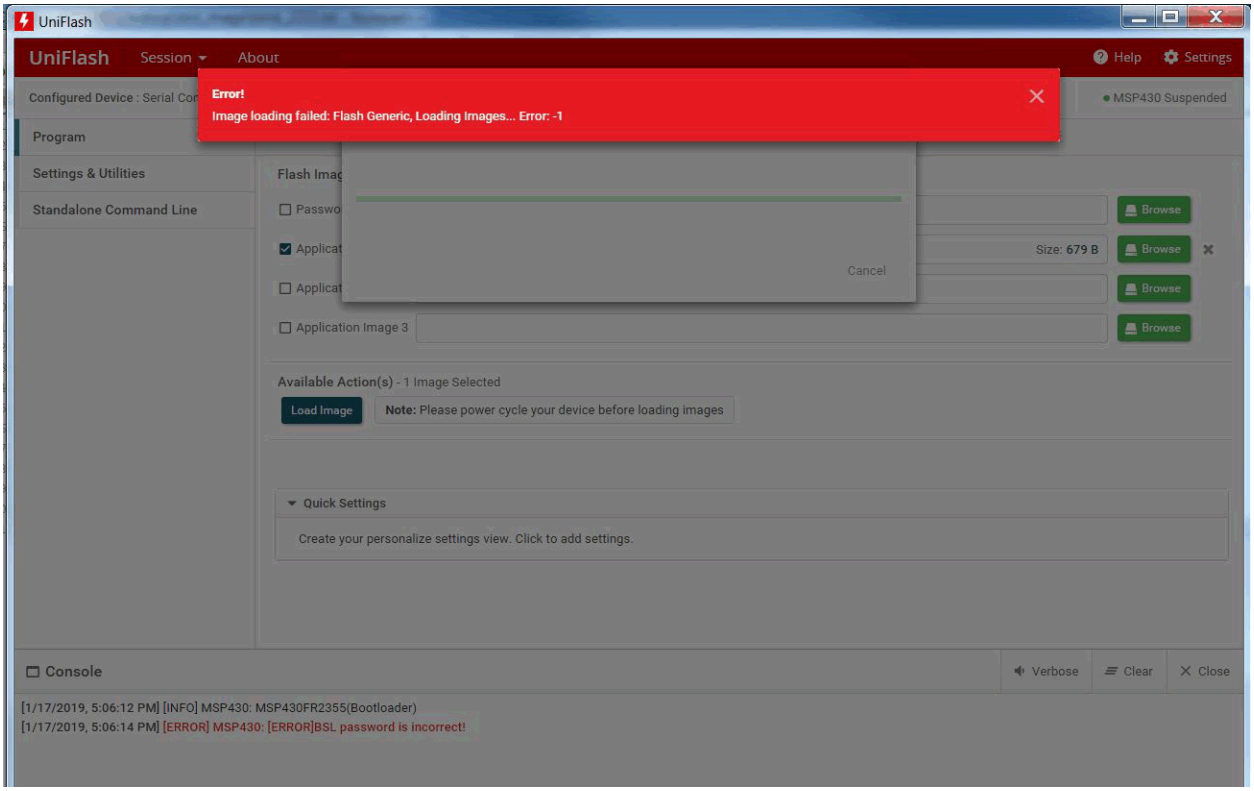

Figure 7. Wrong Password Execution

If the device is empty, the programming is successfully executed and the console shows the log result (see Figure 8).

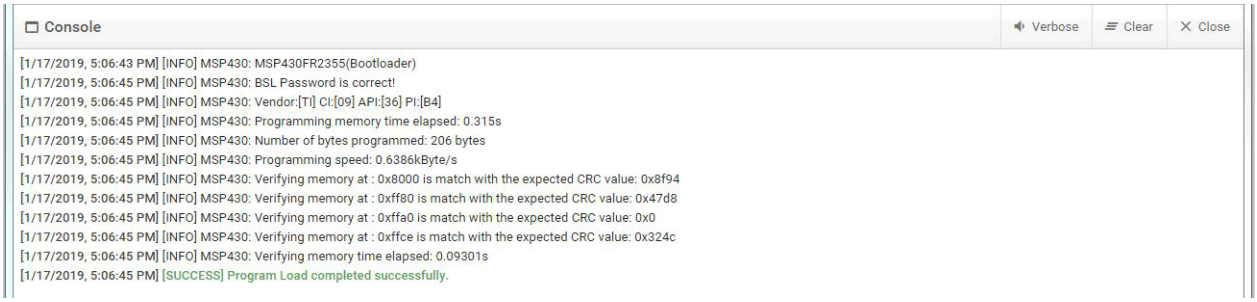

Figure 8. Correct Password Execution Followed by Successful Programming

Copyright © 2019, Texas Instruments Incorporated

5.2 Reading the Memory of the Target Bootloader

To be able to read the content of the memory of target bootloader, the bootloader must be unlocked using the password. The password is 32 bytes that are entered manually in a 32-bit word format (see Figure 9). Concatenate the bytes into 32-bit format using the LSB format with separation of 32 bytes each. The password translates into the hex format in Figure 9.

```
1 0x86808680 ,0x86808680 ,0x86808680 ,0x86808680,0x86808680 ,0x86808680 ,0x86808680 ,0x86806E80
```

Figure 9. Concatenation of the Bytes Into Password Format

Copy the password to the Read section in the Settings & Utilities tab. The Read Image File field specifies the file where UniFlash writes the binary. Enter the start address and number of bytes to read in the following fields.

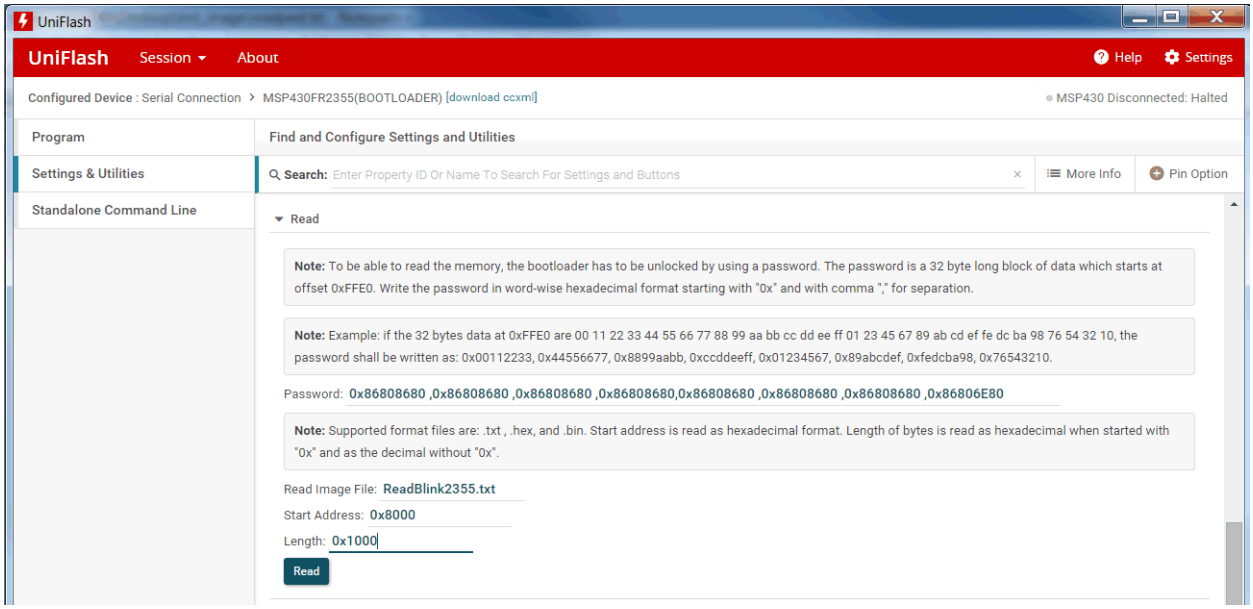

Figure 10. Enter the Configuration on Read Section

Successful reading memory execution show the log in Figure 11.

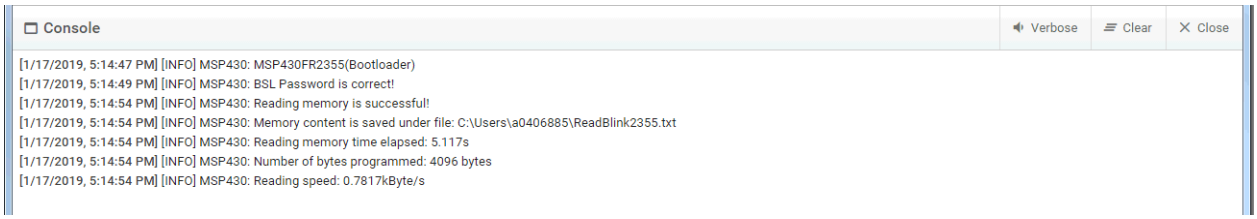

Figure 11. Console View for Reading the Memory Successfully

Copyright © 2019, Texas Instruments Incorporated

6 Bootloader Programming for SimpleLink MSP432P4 MCUs

6.1 Programming the Firmware Image to the Target Bootloader

For the MSP432P4 family, the following example downloads a data block to the memory and reads back the programmed area.

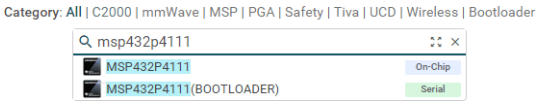

Figure 12. Select the Device MSP432P4111

The Program tab has three text fields for the firmware image. The Password field specifies a file with the bootloader password, to unlock the bootloader before the communication is established. The password is the first 256 bytes of data starting at memory address 0x0h. For more information about the password for the bootloader, see MSP432P4xx SimpleLink™ Microcontrollers Bootloader (BSL) User's Guide.

If the Password field is empty, the default password is used during execution. The default password is 0xFF for all 256 bytes.

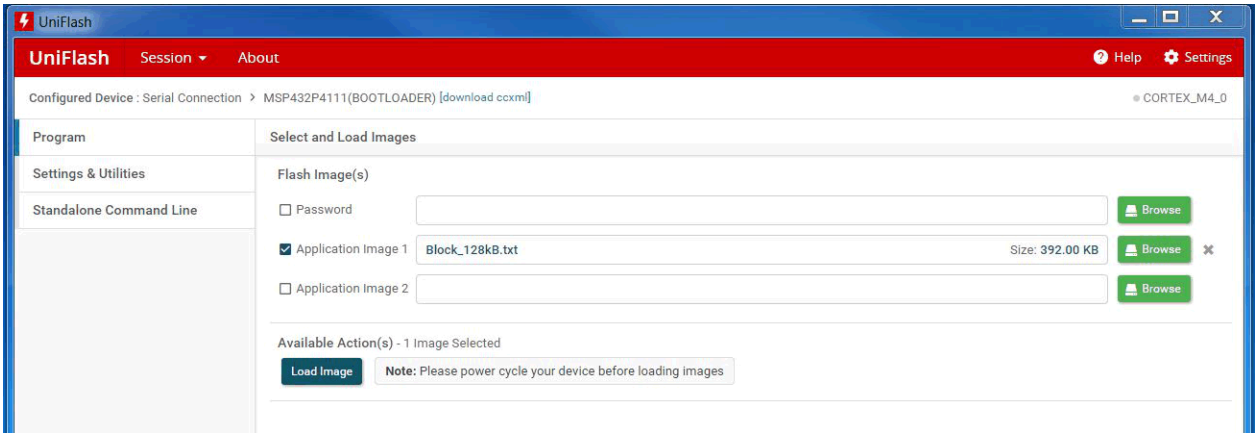

Figure 13. Loading a File Using the 256-Byte Default Password

Copyright © 2019, Texas Instruments Incorporated

The minimum configuration to enter in the Setting & Utilities tab is the COM port (see Figure 14).

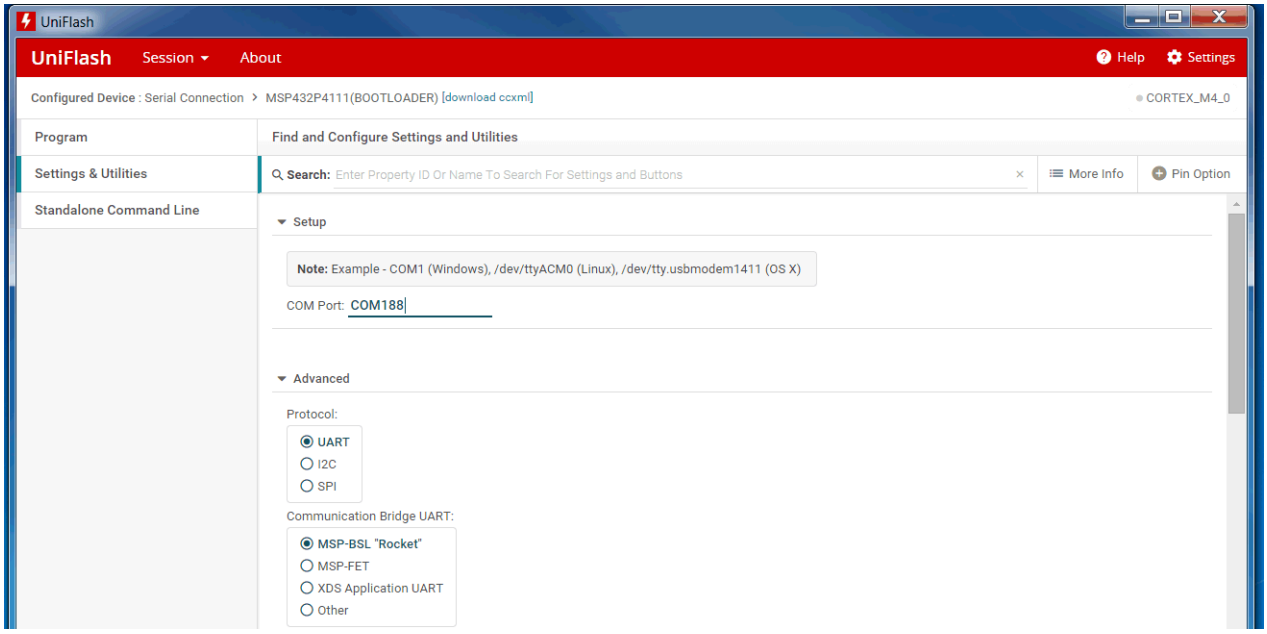

Figure 14. Setting COM Port Number in the Settings & Utilities Tab

If the device is not empty, sending the default password is the same as sending a wrong password. The console shows the BSL password is incorrect, and the program stops execution. When the bootloader receives the wrong password, a mass erase is executed for all of the main memory area.

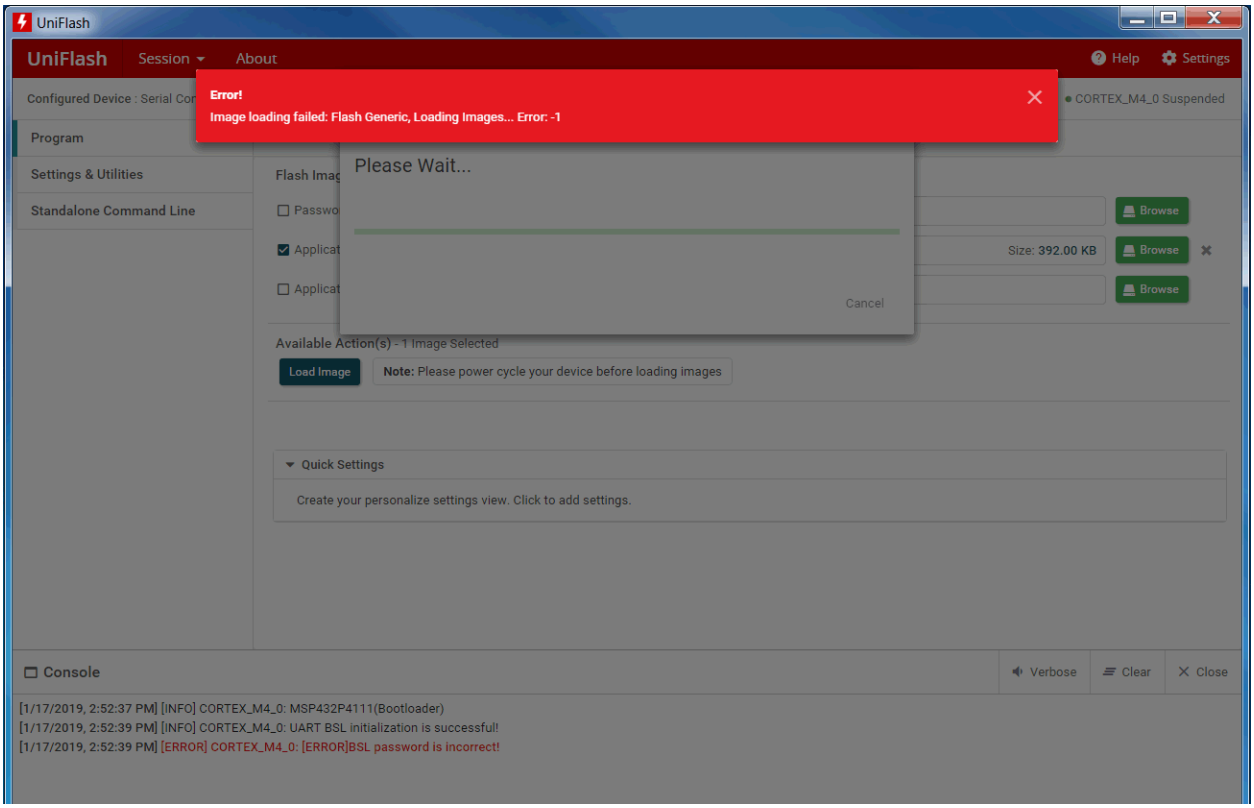

Figure 15. Sending Wrong Password Case

Copyright © 2019, Texas Instruments Incorporated

If you repeat the process with the same configuration, the process executes successfully, because the memory is now empty.

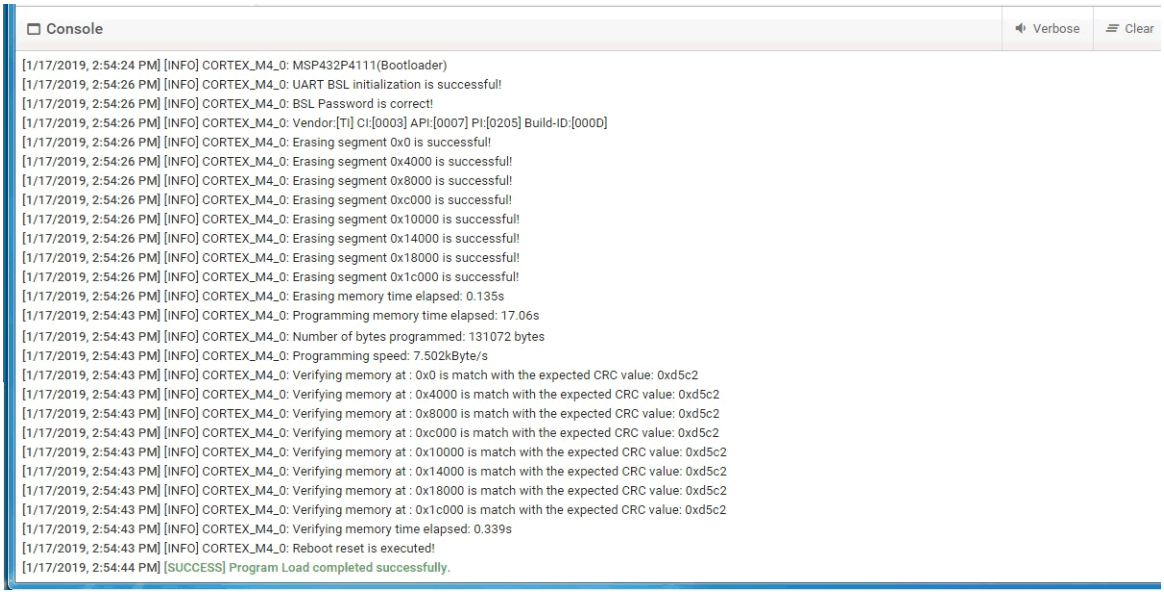

Figure 16. Console View for Programming the Data Block Successfully With the 256-Byte Default Password

6.2 Reading the Memory of the Target Bootloader

To read the contents of the target bootloader from memory, the bootloader must be unlocked using the password. The password is 256 bytes that are entered manually in a 32-bit word format.

The image loaded in the MCU starts at address 0x0h and is 128KB long. The password is the first 256 bytes (see Figure 17).

```
@0
00 11 22 33 44 55 66 77 88 99 aa bb cc dd ee ff
01 11 22 33 44 55 66 77 88 99 aa bb cc dd ee fe
02 11 22 33 44 55 66 77 88 99 aa bb cc dd ee fd
03 11 22 33 44 55 66 77 88 99 aa bb cc dd ee fc
04 11 22 33 44 55 66 77 88 99 aa bb cc dd ee fb
05 11 22 33 44 55 66 77 88 99 aa bb cc dd ee fa
06 11 22 33 44 55 66 77 88 99 aa bb cc dd ee f9
07 11 22 33 44 55 66 77 88 99 aa bb cc dd ee f8
08 11 22 33 44 55 66 77 88 99 aa bb cc dd ee f7
09 11 22 33 44 55 66 77 88 99 aa bb cc dd ee f6
0a 11 22 33 44 55 66 77 88 99 aa bb cc dd ee f5
0b 11 22 33 44 55 66 77 88 99 aa bb cc dd ee f4
0c 11 22 33 44 55 66 77 88 99 aa bb cc dd ee f3
0d 11 22 33 44 55 66 77 88 99 aa bb cc dd ee f2
0e 11 22 33 44 55 66 77 88 99 aa bb cc dd ee f1
0f 11 22 33 44 55 66 77 88 99 aa bb cc dd ee f0
```

Figure 17. Loaded Image on the Target Device

Concatenate the bytes into a 32-bit format using the LSB format with separation of 32 bytes each. The password in Figure 17 is translated into hex format as shown in Figure 18.

```
0x00112233 ,0x44556677 ,0x8899aabb ,0xccddeeff,01112233 ,0x44556677 ,0x8899aabb ,0xccddeefe
0x02112233 ,0x44556677 ,0x8899aabb ,0xccddeefd,03112233 ,0x44556677 ,0x8899aabb ,0xccddeefc
0x04112233 ,0x44556677 ,0x8899aabb ,0xccddeefb,05112233 ,0x44556677 ,0x8899aabb ,0xccddeefa
0x06112233 ,0x44556677 ,0x8899aabb ,0xccddeef9,07112233 ,0x44556677 ,0x8899aabb ,0xccddeef8
0x08112233 ,0x44556677 ,0x8899aabb ,0xccddeef7,09112233 ,0x44556677 ,0x8899aabb ,0xccddeef6
0x0a112233 ,0x44556677 ,0x8899aabb ,0xccddeef5,0b112233 ,0x44556677 ,0x8899aabb ,0xccddeef4
0x0c112233 ,0x44556677 ,0x8899aabb ,0xccddeef3,0d112233 ,0x44556677 ,0x8899aabb ,0xccddeef2
0x0e112233 ,0x44556677 ,0x8899aabb ,0xccddeef1,0f112233 ,0x44556677 ,0x8899aabb ,0xccddeef0
```

Figure 18. Concatenation of the Bytes Into Password Format

Copyright © 2019, Texas Instruments Incorporated

Copy the password to the Read section in the Settings & Utilities tab. The filename specifies a file where UniFlash can write the binary and save in the users folder automatically. Type the start address and how many bytes to read in the following text fields.

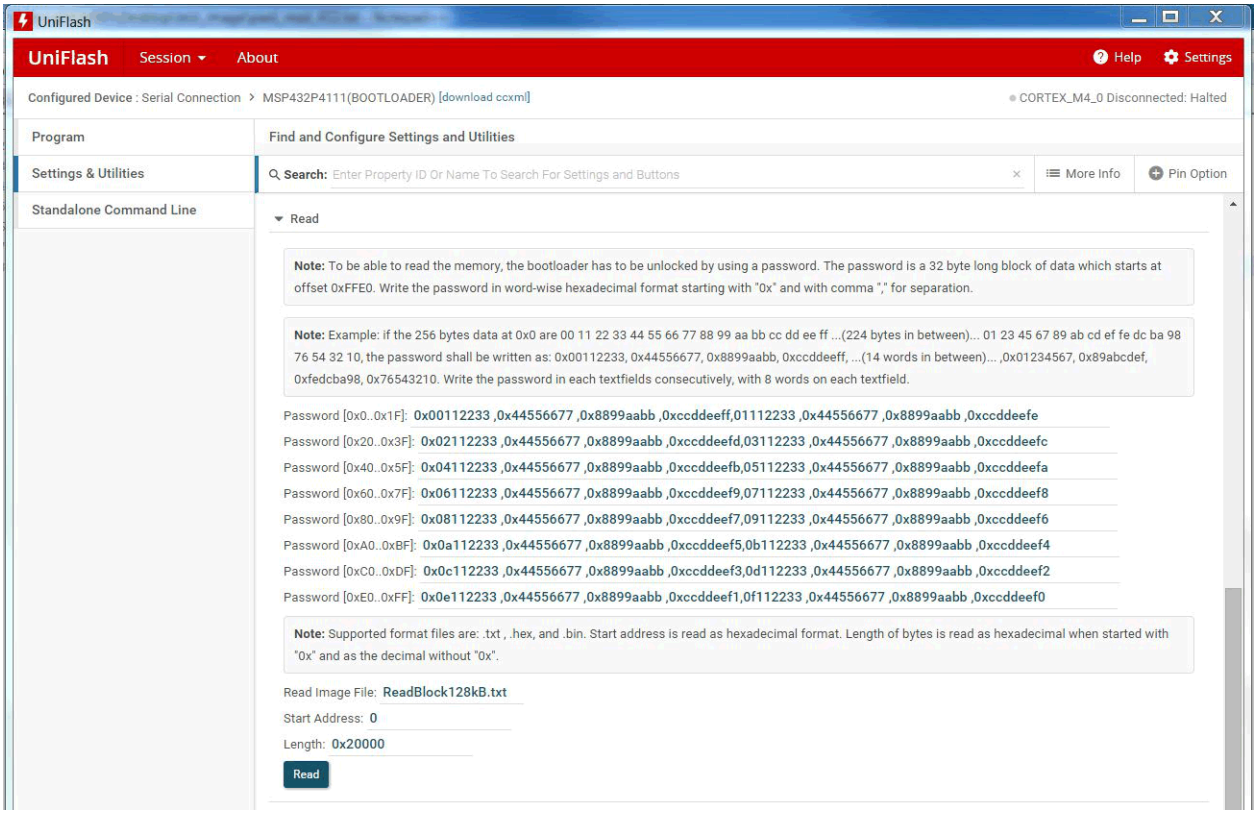

Figure 19. Enter the Configuration on Read Section

Figure 20 shows the console view after successful reading of the memory.

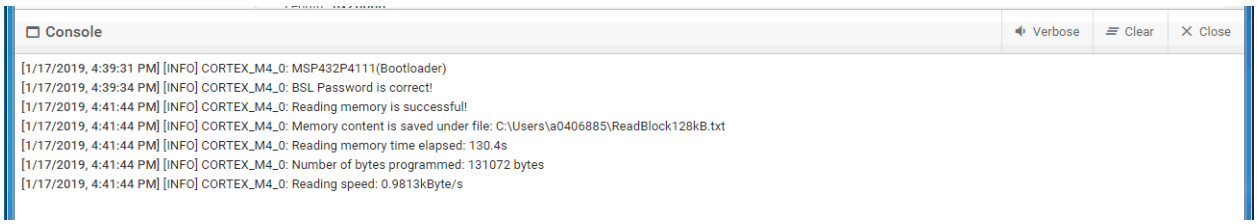

Figure 20. Console View for Reading the Memory Successfully

Copyright © 2019, Texas Instruments Incorporated

7 Bootloader Programming for SimpleLink MSP432E4 MCUs

7.1 Programming the Firmware Image to the Target Bootloader

For the MSP432E4 MCU, the following example downloads the UART bootloader flash-based application from the SDK. The device must be fully erased to run this example.

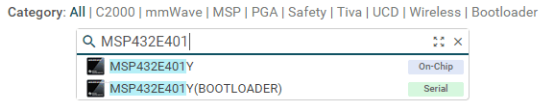

Figure 21. Select the Device MSP432E401Y

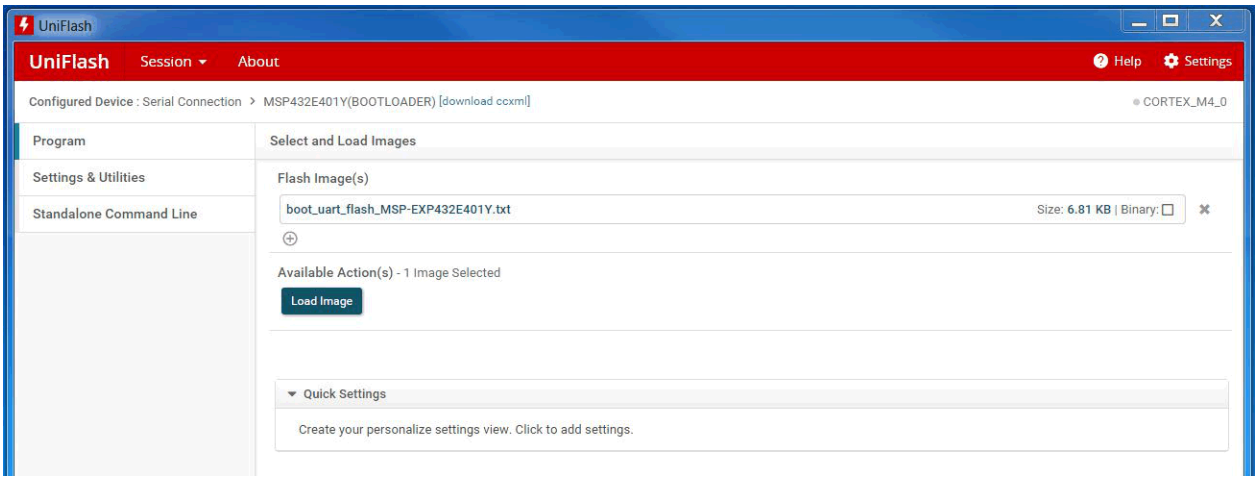

Figure 22. Enter the Firmware Image

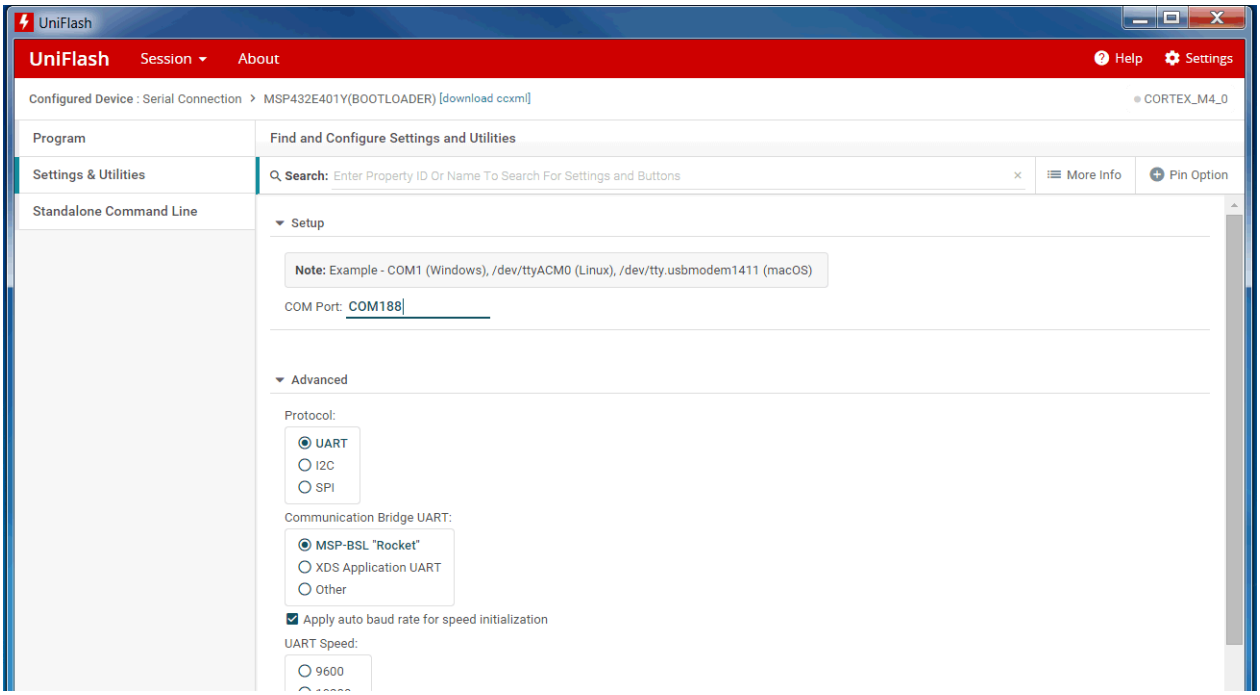

Figure 23. Configure the COM Port and Use the Default Settings for Other Configurations

Copyright © 2019, Texas Instruments Incorporated

Select Load Image to start the programming process. The console shows the log of each operation (see Figure 24).

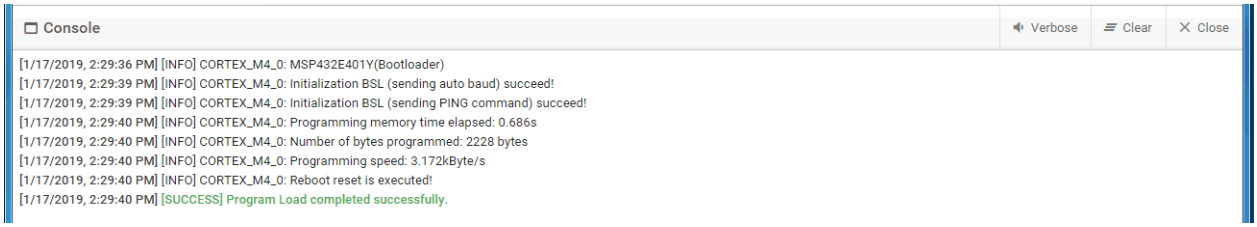

Figure 24. Console View of Downloading the UART Bootloader Flash-Based Application

After the first programming successfully executes, the UART bootloader flash-based application runs each time the device is reset. The next step is to program a blink LED application that is provided in the SDK examples with no auto-baud rate configuration.

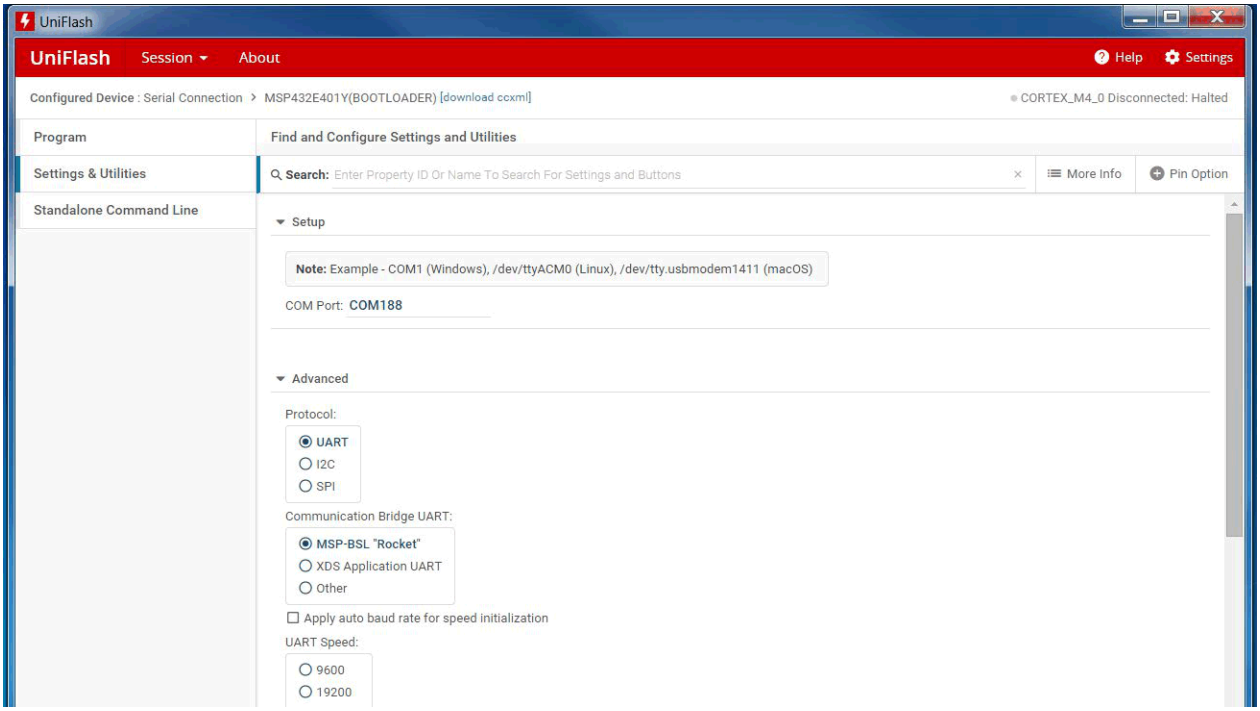

Figure 25. Uncheck "Apply auto baud rate for speed initialization"

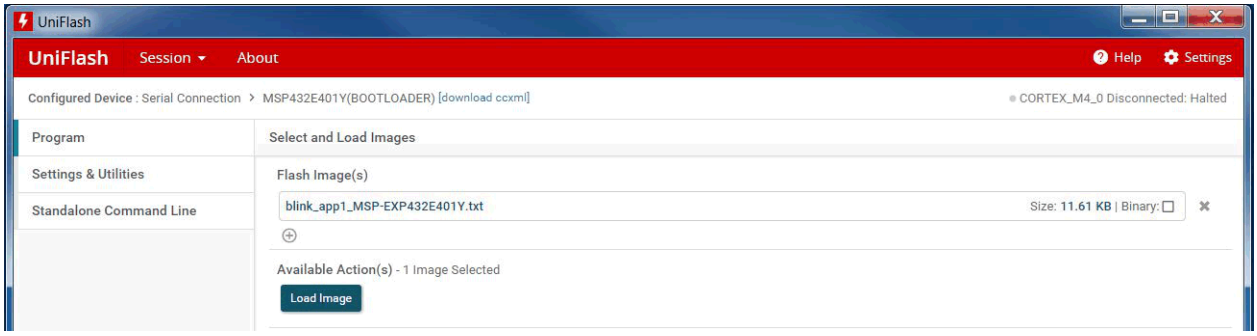

Figure 26. Select the Firmware Image of Blink Application

Copyright © 2019, Texas Instruments Incorporated

The console output after a successful operation shows that the initialization only executes the PING command without auto baud rate (see Figure 27).

Figure 27. Console View of Downloading the Blink Application

8 Bootloader Programming for SimpleLink CC13xx and CC26xx MCUs

8.1 Handling the CCFG Configuration Under Firmware Image

By default, creating a firmware image based on examples in the SDK create the CCFG configuration. CCFG configuration is programmed at these addresses:

- 0x57FA8h for CC13x2 and CC26x2 devices
- 0x1FFA8h for CC13x0 and CC26x0 devices

UniFlash supports CCFG configuration during bootloader programming. Therefore, any CCFG configuration from the firmware image is discarded. To not build the CCFG section during compilation, set the "Exclude from Build" option on the ccfg.c file that is automatically imported to the project when using the example from the SDK.

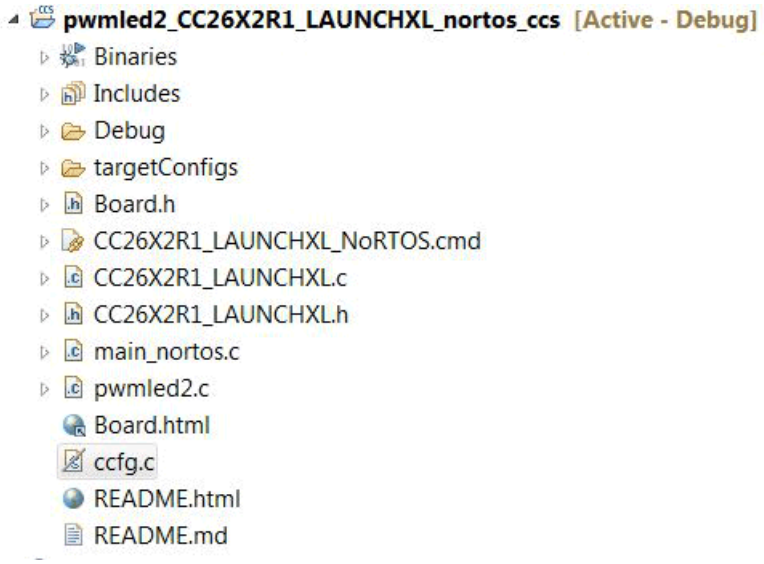

Figure 28. Apply "Exclude from Build" for ccfg.c

Another option to is to manually delete the CCFG section from the generated firmware image. The CCFG section is last in the file (see Figure 29).

Copyright © 2019, Texas Instruments Incorporated

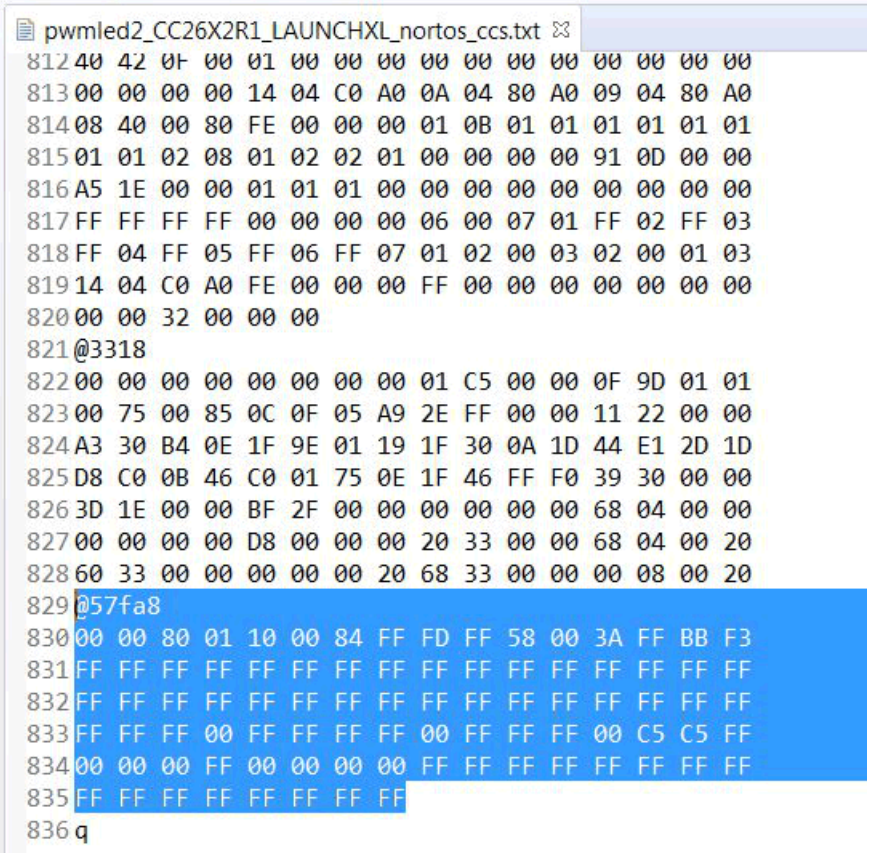

Figure 29. Location Where CCFG is Programmed and Needs to be Deleted

8.2 Programming the Firmware Image Into the Target Bootloader

To select one of the CC13xx and CC26xx wireless MCUs, type the device name and then select the entry that ends with "(Bootloader)" and that has a label of "Serial" next to it (see Figure 30). The "On-Chip" label is a programming feature that uses a supported debugger. On this example, the LaunchXL-CC26x2R1 with CC2652R1F is used.

Figure 30. Select Device CC2652R1F

Up to three separated images can be programmed on the device.

Copyright © 2019, Texas Instruments Incorporated

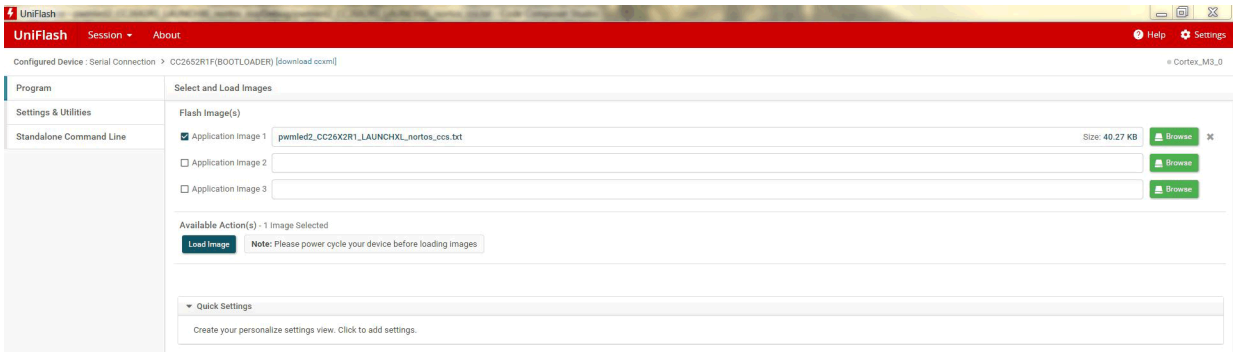

Figure 31. Load the Image

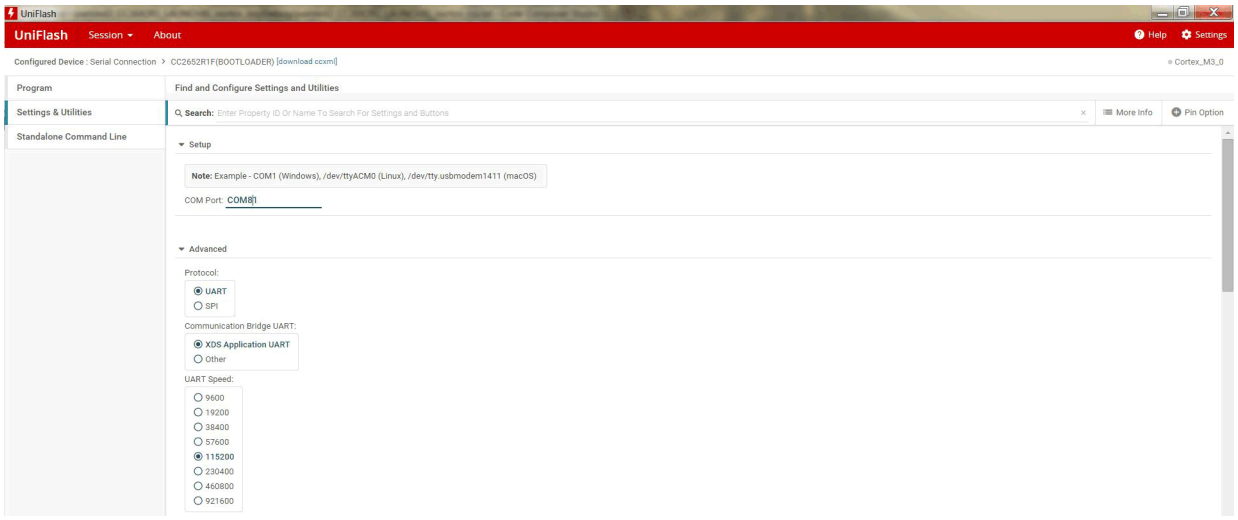

Figure 32. Configure the COM Port and Use the Default Settings for Other Configuration

After you click "Load Image", UniFlash starts the programming process and the console shows the log of each operation.

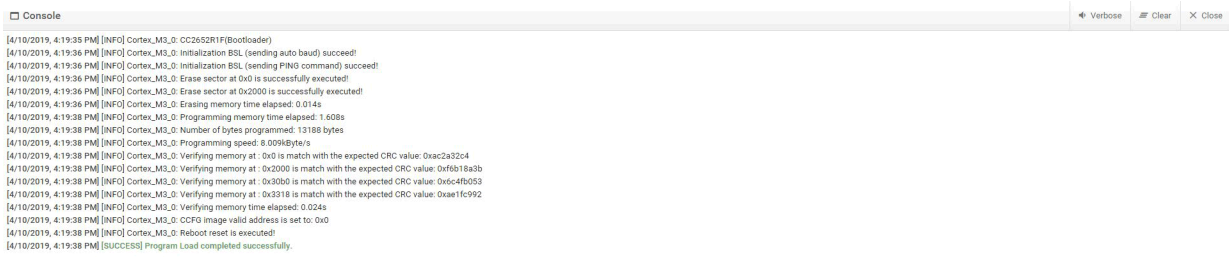

Figure 33. Console View of Downloading the pwmled Application

In the default setting of CCFG, the Image Valid Configuration in the CCFG section is required. Set the Image Valid Configuration to 0x0000:0000 to enable the boot sequence to transfer control to the user application image. Any other value forces the boot sequence to call the bootloader instead.

Copyright © 2019, Texas Instruments Incorporated

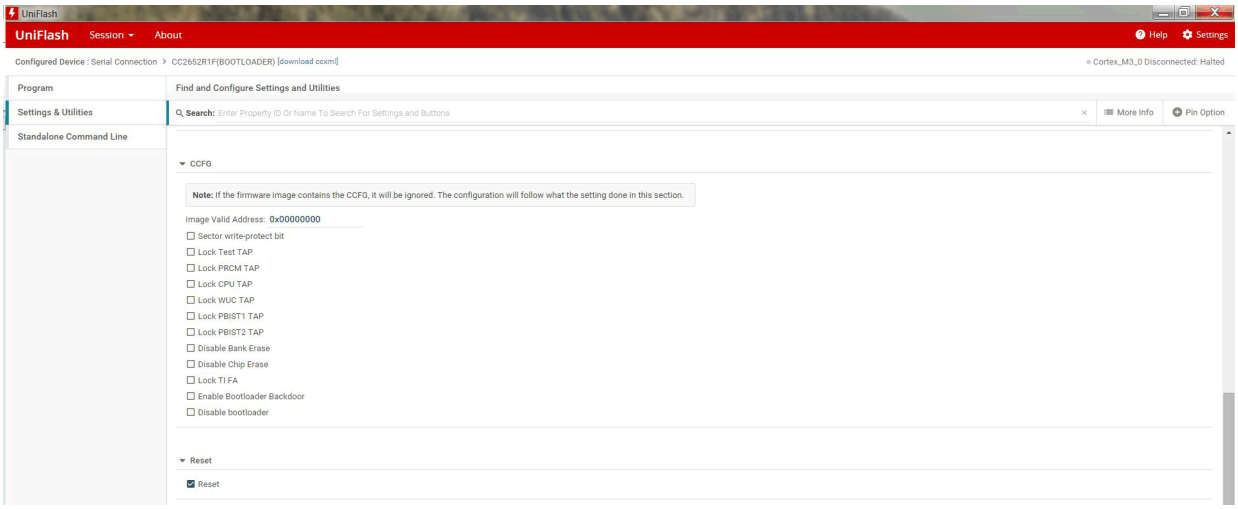

Figure 34. Image Valid Address Configuration

8.3 Reading the Memory of the Target Bootloader

Reading the programmed memory is possible by using the bootloader backdoor invocation. To enable bootloader backdoor invocation, the CCFG configuration must be done during the initial programming.

Parameters needed to do the reading are:

- File name that ends with .txt, .hex, or .bin according with the expected file format
- Start address of the memory to read
- Number of bytes to read

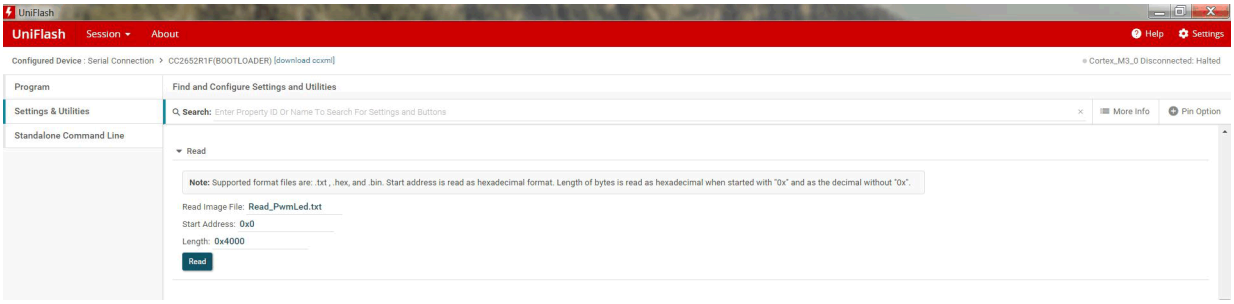

Figure 35. Enter the Configuration on Read Section

Successful reading memory execution will show the log in the console as shown below.

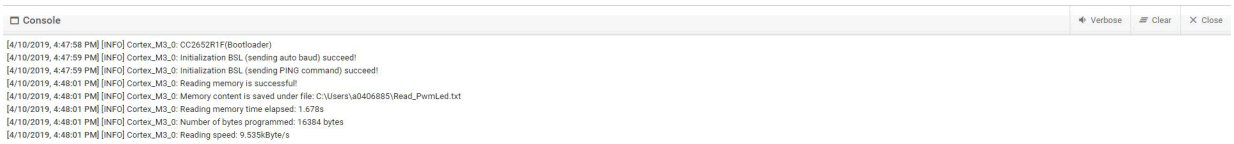

Figure 36. Console View for Reading the Memory Successfully

Copyright © 2019, Texas Instruments Incorporated

9 Related Documents

1. MSP430™ Flash Device Bootloader (BSL) User's Guide
2. MSP430™ FRAM Device Bootloader (BSL) User's Guide
3. MSP432P4xx SimpleLink™ Microcontrollers Bootloader (BSL) User's Guide
4. MSP432E4 SimpleLink™ Microcontrollers Bootloader (BSL) User's Guide
5. CC13x2, CC26x2 SimpleLink™ Wireless MCU Technical Reference Manual
6. CC13x0, CC26x0 SimpleLink™ Wireless MCU Technical Reference Manual
7. CC2538/CC26x0/CC26x2 Serial Bootloader Interface

Copyright © 2019, Texas Instruments Incorporated

Revision History

NOTE: Page numbers for previous revisions may differ from page numbers in the current version.

Changes from January 31, 2019 to April 15, 2019 **Page**

- Updated the document title to add CC13xx and CC26xx MCUs 1
- Added devices supported in UniFlash 5.0.0 2
- Added Section 8, *Bootloader Programming of SimpleLink CC13xx and CC26xx MCUs* 14
- Added Section 9, *Related Documents* 18

Copyright © 2019, Texas Instruments Incorporated

IMPORTANT NOTICE AND DISCLAIMER

TI PROVIDES TECHNICAL AND RELIABILITY DATA (INCLUDING DATASHEETS), DESIGN RESOURCES (INCLUDING REFERENCE DESIGNS), APPLICATION OR OTHER DESIGN ADVICE, WEB TOOLS, SAFETY INFORMATION, AND OTHER RESOURCES “AS IS” AND WITH ALL FAULTS, AND DISCLAIMS ALL WARRANTIES, EXPRESS AND IMPLIED, INCLUDING WITHOUT LIMITATION ANY IMPLIED WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE OR NON-INFRINGEMENT OF THIRD PARTY INTELLECTUAL PROPERTY RIGHTS.

These resources are intended for skilled developers designing with TI products. You are solely responsible for (1) selecting the appropriate TI products for your application, (2) designing, validating and testing your application, and (3) ensuring your application meets applicable standards, and any other safety, security, or other requirements. These resources are subject to change without notice. TI grants you permission to use these resources only for development of an application that uses the TI products described in the resource. Other reproduction and display of these resources is prohibited. No license is granted to any other TI intellectual property right or to any third party intellectual property right. TI disclaims responsibility for, and you will fully indemnify TI and its representatives against, any claims, damages, costs, losses, and liabilities arising out of your use of these resources.

TI’s products are provided subject to TI’s Terms of Sale (www.ti.com/legal/termsofsale.html) or other applicable terms available either on ti.com or provided in conjunction with such TI products. TI’s provision of these resources does not expand or otherwise alter TI’s applicable warranties or warranty disclaimers for TI products.

Mailing Address: Texas Instruments, Post Office Box 655303, Dallas, Texas 75265
Copyright © 2019, Texas Instruments Incorporated