SiWG917 – TA Flash Memory Map Change Guide

Version 1.3

November 2023

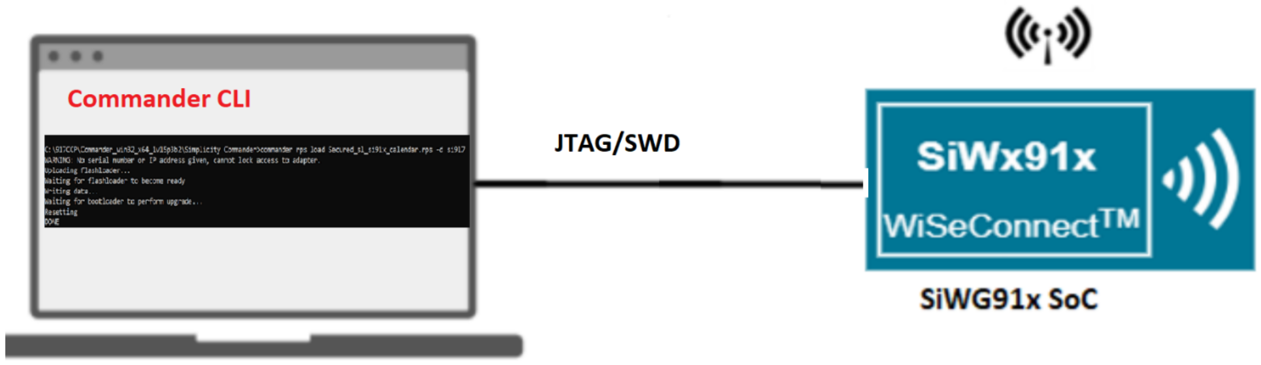
Table of Contents

1 Introduction

For SiWG917 IC (SiWG917M111MGTBA), Flash is shared between the Wireless processor (TA) and Cortex M4 processor(M4). So far, the Wireless image has been 1.6MB. All eval boards and IC shipments done before Oct 2023 are based on this configuration. The Master boot Record (MBR) and SW releases are based on this 1.6MB Wireless image. New Wireless enhancements/features adoption may increase the size of the Wireless image to 1.8MB. Users are recommended to change their existing devices to support Wireless Image with 1.8MB and make them future proof. For this change to affect, an MBR upgrade is needed.

This document is a user guide that helps in identifying the MBR version of the device and guides the user to update the MBR (if required) using the Commander CLI tool. The tool performs best with systems running on Windows, Linux, and MacOS.

Users must adapt the change based on the results from MBR version check. If 'MBR version check' returns 1F, then no change to MBR is needed. In this case, users can directly make project-related changes per section Configuring the M4. In case the MBR value is 1B, the user needs to use Simplicity Commander CLI and update the MBR per the following section starting from Simplicity Commander CLI.



```
C:\917CCP\Commander_win32_x64_1v15p3b2\Simplicity Commander>commander rps load Secured_sl_si91x_calendar.rps -d si917
WARNING: No serial number or IP address given, cannot lock access to adapter.
Uploading flashloader...
Waiting for flashloader to become ready
Writing data...
Waiting for bootloader to perform upgrade...
Resetting
DONE
```

Note: After the MBR is updated, it is mandatory to follow Configuring the M4 Application section and flash the application. If above step is skipped, the device will be corrupted and may be difficult to recover in certain scenarios.

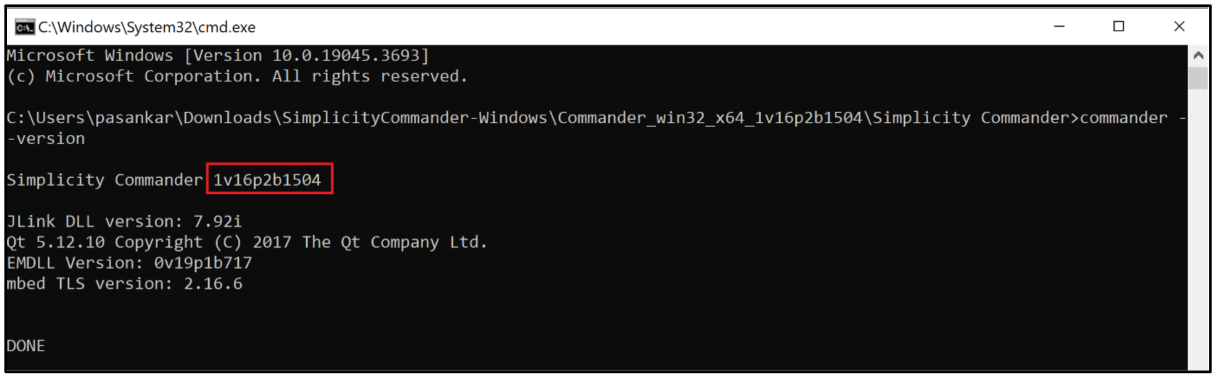
2 Prerequisites

2.1 Hardware

- BRD4338A with BRD4002A.
- USB Type C cable to connect to the PC.

2.2 Software

- Simplicity Commander CLI (1v16p1 and above versions)
  Note : To check commander version open the cli mode (To open commander cli mode follow Section 3.1 from step 1 to step 3) and give the following command :
  commander –version



C:\Windows\System32\cmd.exe

```
Microsoft Windows [Version 10.0.19045.3693]
(c) Microsoft Corporation. All rights reserved.

C:\Users\pasankar\Downloads\SimplicityCommander-Windows\Commander_win32_x64_1v16p2b1504\Simplicity Commander>commander -
-version

Simplicity Commander 1v16p2b1504

JLink DLL version: 7.92i
Qt 5.12.10 Copyright (C) 2017 The Qt Company Ltd.
EMDLL Version: 0v19p1b717
mbed TLS version: 2.16.6


DONE
```

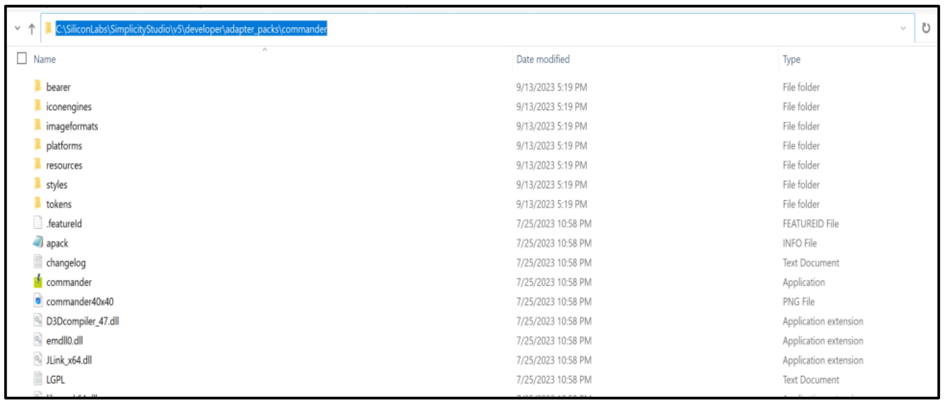
3 MBR Version check

3.1 Using Simplicity Commander CLI

This section will guide the user to read the address location "**0x4000194**". Depending on the output the user will know what MBR version is loaded in the board/IC.

Steps to follow:

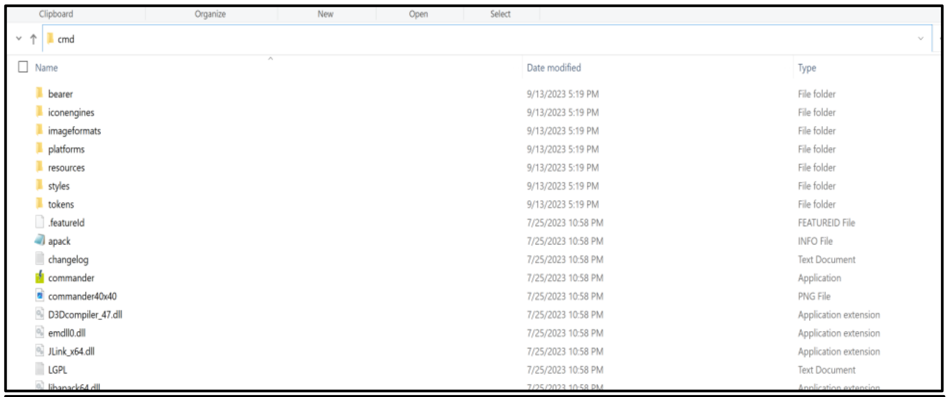
1. Connect the device to the PC using USB cable (type C).
2. Traverse to the path where Simplicity Studio is installed
   For our case the path **(default path)** is : **C:\SiliconLabs\SimplicityStudio\v5\developer\adapter_packs\commander**



C:\SiliconLabs\SimplicityStudio\v5\developer\adapter_packs\commander

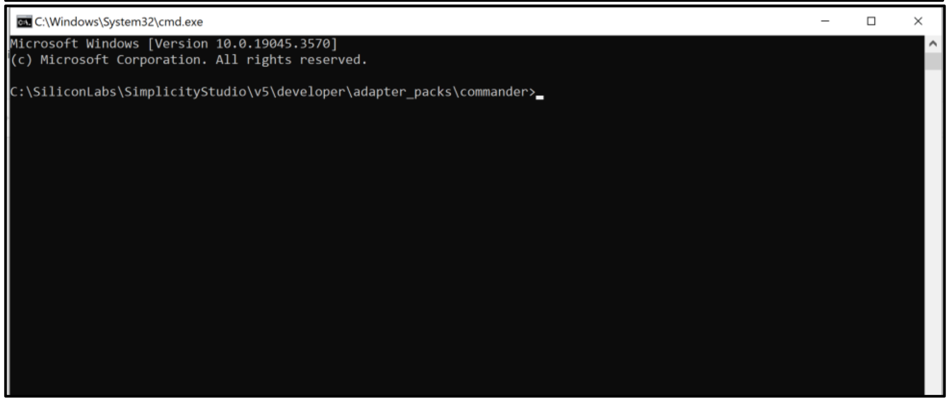
Name	Date modified	Type
bearer	9/13/2023 5:19 PM	File folder
iconengines	9/13/2023 5:19 PM	File folder
imageformats	9/13/2023 5:19 PM	File folder
platforms	9/13/2023 5:19 PM	File folder
resources	9/13/2023 5:19 PM	File folder
styles	9/13/2023 5:19 PM	File folder
tokens	9/13/2023 5:19 PM	File folder
.featureId	7/25/2023 10:58 PM	FEATUREID File
apack	7/25/2023 10:58 PM	INFO File
changelog	7/25/2023 10:58 PM	Text Document
commander	7/25/2023 10:58 PM	Application
commander40x40	7/25/2023 10:58 PM	PNG File
D3Dcompiler_47.dll	7/25/2023 10:58 PM	Application extension
emdll0.dll	7/25/2023 10:58 PM	Application extension
JLink_x64.dll	7/25/2023 10:58 PM	Application extension
LGPL	7/25/2023 10:58 PM	Text Document

3. Type "**cmd**" in highlighted part section in the above picture and hit enter. It will open the command prompt CLI in that path.



cmd

Name	Date modified	Type
bearer	9/13/2023 5:19 PM	File folder
iconengines	9/13/2023 5:19 PM	File folder
imageformats	9/13/2023 5:19 PM	File folder
platforms	9/13/2023 5:19 PM	File folder
resources	9/13/2023 5:19 PM	File folder
styles	9/13/2023 5:19 PM	File folder
tokens	9/13/2023 5:19 PM	File folder
.featureId	7/25/2023 10:58 PM	FEATUREID File
apack	7/25/2023 10:58 PM	INFO File
changelog	7/25/2023 10:58 PM	Text Document
commander	7/25/2023 10:58 PM	Application
commander40x40	7/25/2023 10:58 PM	PNG File
D3Dcompiler_47.dll	7/25/2023 10:58 PM	Application extension
emdll0.dll	7/25/2023 10:58 PM	Application extension
JLink_x64.dll	7/25/2023 10:58 PM	Application extension
LGPL	7/25/2023 10:58 PM	Text Document



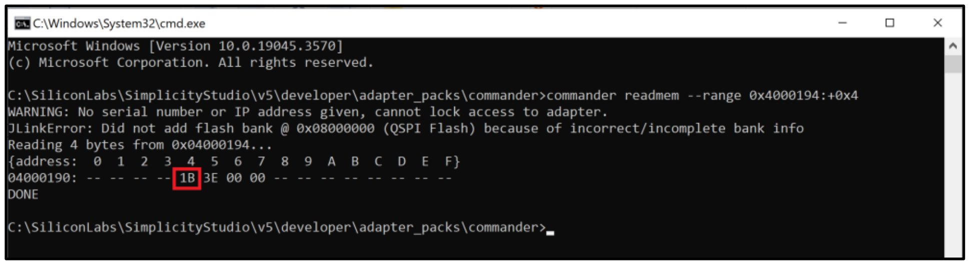
C:\Windows\System32\cmd.exe

```
Microsoft Windows [Version 10.0.19045.3570]
(c) Microsoft Corporation. All rights reserved.

C:\SiliconLabs\SimplicityStudio\v5\developer\adapter_packs\commander>
```

4. Enter the following command in the command prompt CLI to read the MBR from memory location address "**0x4000194**". Command: **commander readmem --range 0x4000194:+0x4**

5. User would get output similar to below after executing the above command.



C:\Windows\System32\cmd.exe

```
Microsoft Windows [Version 10.0.19045.3570]
(c) Microsoft Corporation. All rights reserved.

C:\SiliconLabs\SimplicityStudio\v5\developer\adapter_packs\commander>commander readmem --range 0x4000194:+0x4
WARNING: No serial number or IP address given, cannot lock access to adapter.
JLinkError: Did not add flash bank @ 0x08000000 (QSPI Flash) because of incorrect/incomplete bank info
Reading 4 bytes from 0x04000194...
{address:  0  1  2  3  4  5  6  7  8  9  A  B  C  D  E  F}
04000190: -- -- -- -- 1B 3E 00 00 -- -- -- -- -- -- -- --
DONE

C:\SiliconLabs\SimplicityStudio\v5\developer\adapter_packs\commander>_
```

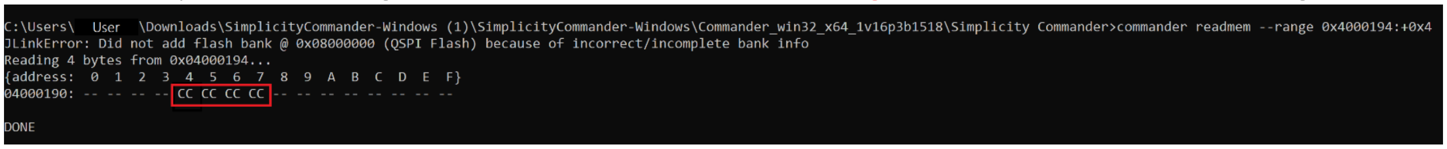
6. If the output contains "1B"(as marked in red box in the image above) then the device has 1.6 MB MBR flashed in it

   Go to section "Programming the MBR"

   If the output contains "1F" (same place where red box in the image above) then the device has 1.8 MB MBR flashed in it.

   User can use the GA release to proceed or Go to section "Configuring the M4 Application" if using release earlier than GA release to find out linker changes.

Note : If the output (after executing the command: **commander readmem --range 0x4000194:+0x4**) is like the below figure :



```
C:\Users\ User \Downloads\SimplicityCommander-Windows (1)\SimplicityCommander-Windows\Commander_win32_x64_1v16p3b1518\Simplicity Commander>commander readmem --range 0x4000194:+0x4
JLinkError: Did not add flash bank @ 0x08000000 (QSPI Flash) because of incorrect/incomplete bank info
Reading 4 bytes from 0x04000194...
{address:  0  1  2  3  4  5  6  7  8  9  A  B  C  D  E  F}
04000190: -- -- -- -- CC CC CC CC -- -- -- -- -- -- -- --

DONE
```

Having all "CC"s(marked inside red box) , then the board is corrupted. Please go to Section 6 to reflash 1.8v MBR in it.

4 Programming the MBR

The following steps are for Programming the MBR in the Common Flash Device(s).

4.1 Backup the original TA, M4, and efuse contents

This step is recommended before doing any update.

TA MBR: `commander manufacturing read tambr --out <filename.bin>`
Example: commander manufacturing read **tambr** --out **tambr.bin**

M4 MBR: `commander manufacturing read m4mbrcf --out <filename.bin>`
Example: commander manufacturing read **m4mbrcf** --out **m4mbr.bin**

eFusecopy: `commander manufacturing read efusecopy --out <filename.bin>`
Example: commander manufacturing read efusecopy --out efusecopy.bin

4.2 MBR File(s)

The file that shall be used to update the MBR to support the 1.8MB Wireless Image.

Board Number	MBR File Link
BRD4338A	ta_mbr_SiWG917M1xxMGTBA.bin

Note: PSRAM support is disabled in the above mentioned MBR file, if PSRAM support is needed, please contact Silicon Labs support.

Download this file and copy to commander folder. E.g. In this case default path
C:\SiliconLabs\SimplicityStudio\v5\developer\adapter_packs\commander

4.3 Flashing Procedure

Following is the sequence to program the device(s).

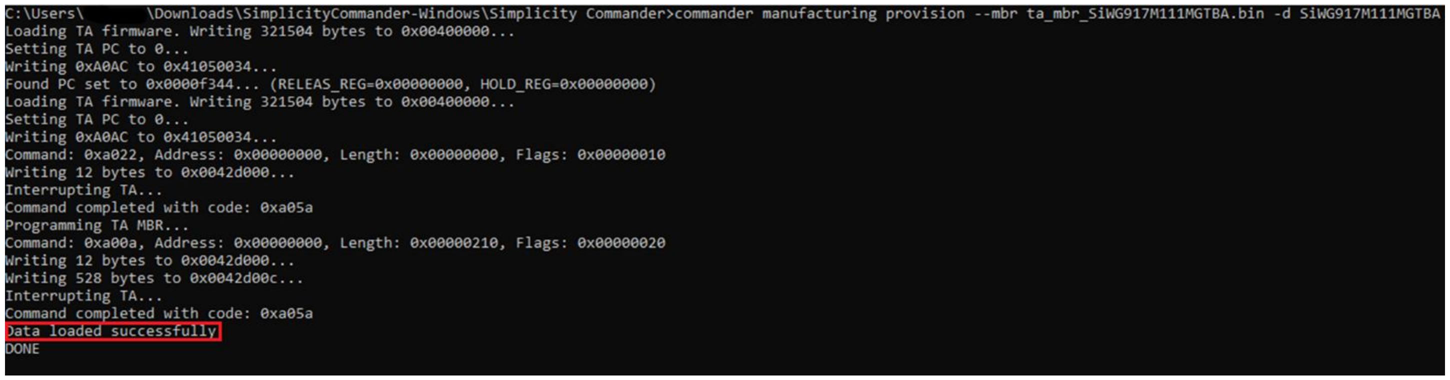
1. Write TA MBR
2. Write M4 MBR
3. Write the calibration data to the M4 Flash

4.3.1 Write TA MBR

Use the below command to update the TA MBR.

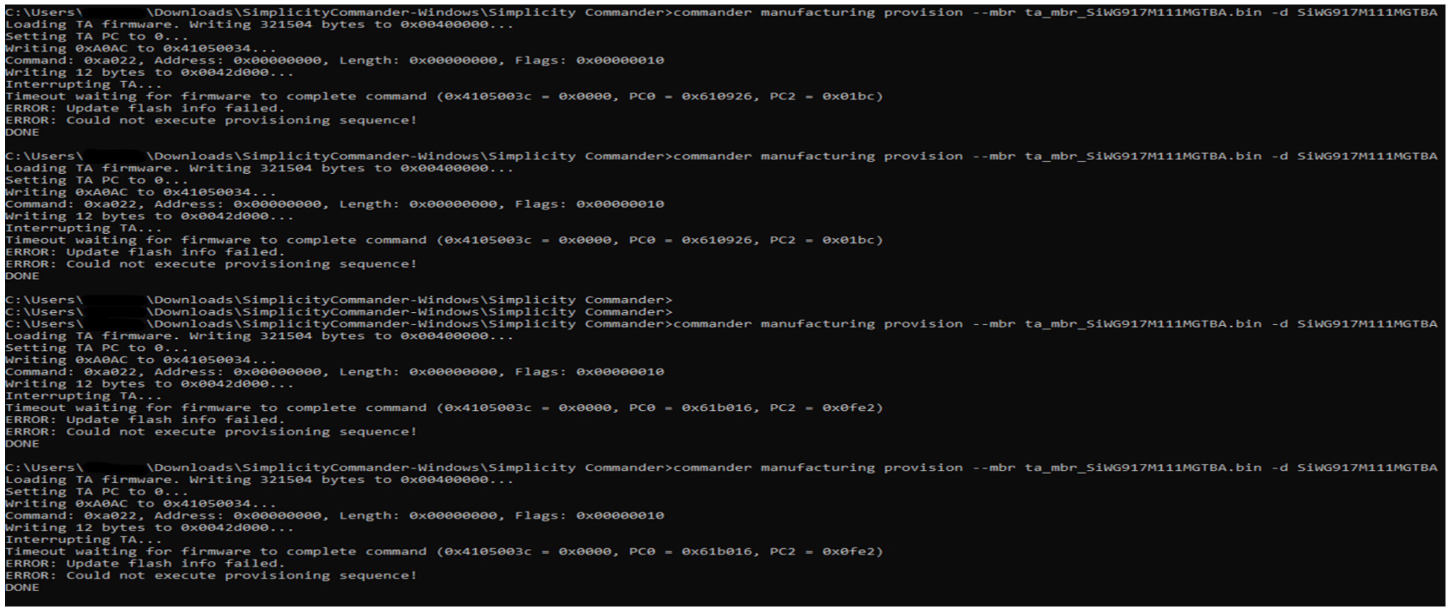
Command: `commander manufacturing provision --mbr <filename.bin> -d <full opn>`

Example: commander manufacturing provision --mbr **ta_mbr_SiWG917M1xxMGTBA.bin** -d **SiWG917M111MGTBA**



```
C:\Users\[redacted]\Downloads\SimplicityCommander-Windows\Simplicity Commander>commander manufacturing provision --mbr ta_mbr_SiWG917M111MGTBA.bin -d SiWG917M111MGTBA
Loading TA firmware. Writing 321504 bytes to 0x00400000...
Setting TA PC to 0...
Writing 0xA0AC to 0x41050034...
Found PC set to 0x0000f344... (RELEAS_REG=0x00000000, HOLD_REG=0x00000000)
Loading TA firmware. Writing 321504 bytes to 0x00400000...
Setting TA PC to 0...
Writing 0xA0AC to 0x41050034...
Command: 0xa022, Address: 0x00000000, Length: 0x00000000, Flags: 0x00000010
Writing 12 bytes to 0x0042d000...
Interrupting TA...
Command completed with code: 0xa05a
Programming TA MBR...
Command: 0xa00a, Address: 0x00000000, Length: 0x00000210, Flags: 0x00000020
Writing 12 bytes to 0x0042d000...
Writing 528 bytes to 0x0042d00c...
Interrupting TA...
Command completed with code: 0xa05a
Data loaded successfully
DONE
```

Note: User might see failure multiple times while updating the TA MBR(refer below image). "Reset" the board and try again.



```
C:\Users\[redacted]\Downloads\SimplicityCommander-Windows\Simplicity Commander>commander manufacturing provision --mbr ta_mbr_SiWG917M111MGTBA.bin -d SiWG917M111MGTBA
Loading TA firmware. Writing 321504 bytes to 0x00400000...
Setting TA PC to 0...
Writing 0xA0AC to 0x41050034...
Command: 0xa022, Address: 0x00000000, Length: 0x00000000, Flags: 0x00000010
Writing 12 bytes to 0x0042d000...
Interrupting TA...
Timeout waiting for firmware to complete command (0x4105003c = 0x0000, PC0 = 0x610926, PC2 = 0x01bc)
ERROR: Update flash info failed.
ERROR: Could not execute provisioning sequence!
DONE

C:\Users\[redacted]\Downloads\SimplicityCommander-Windows\Simplicity Commander>commander manufacturing provision --mbr ta_mbr_SiWG917M111MGTBA.bin -d SiWG917M111MGTBA
Loading TA firmware. Writing 321504 bytes to 0x00400000...
Setting TA PC to 0...
Writing 0xA0AC to 0x41050034...
Command: 0xa022, Address: 0x00000000, Length: 0x00000000, Flags: 0x00000010
Writing 12 bytes to 0x0042d000...
Interrupting TA...
Timeout waiting for firmware to complete command (0x4105003c = 0x0000, PC0 = 0x610926, PC2 = 0x01bc)
ERROR: Update flash info failed.
ERROR: Could not execute provisioning sequence!
DONE

C:\Users\[redacted]\Downloads\SimplicityCommander-Windows\Simplicity Commander>
C:\Users\[redacted]\Downloads\SimplicityCommander-Windows\Simplicity Commander>
C:\Users\[redacted]\Downloads\SimplicityCommander-Windows\Simplicity Commander>commander manufacturing provision --mbr ta_mbr_SiWG917M111MGTBA.bin -d SiWG917M111MGTBA
Loading TA firmware. Writing 321504 bytes to 0x00400000...
Setting TA PC to 0...
Writing 0xA0AC to 0x41050034...
Command: 0xa022, Address: 0x00000000, Length: 0x00000000, Flags: 0x00000010
Writing 12 bytes to 0x0042d000...
Interrupting TA...
Timeout waiting for firmware to complete command (0x4105003c = 0x0000, PC0 = 0x61b016, PC2 = 0x0fe2)
ERROR: Update flash info failed.
ERROR: Could not execute provisioning sequence!
DONE

C:\Users\[redacted]\Downloads\SimplicityCommander-Windows\Simplicity Commander>commander manufacturing provision --mbr ta_mbr_SiWG917M111MGTBA.bin -d SiWG917M111MGTBA
Loading TA firmware. Writing 321504 bytes to 0x00400000...
Setting TA PC to 0...
Writing 0xA0AC to 0x41050034...
Command: 0xa022, Address: 0x00000000, Length: 0x00000000, Flags: 0x00000010
Writing 12 bytes to 0x0042d000...
Interrupting TA...
Timeout waiting for firmware to complete command (0x4105003c = 0x0000, PC0 = 0x61b016, PC2 = 0x0fe2)
ERROR: Update flash info failed.
ERROR: Could not execute provisioning sequence!
DONE
```

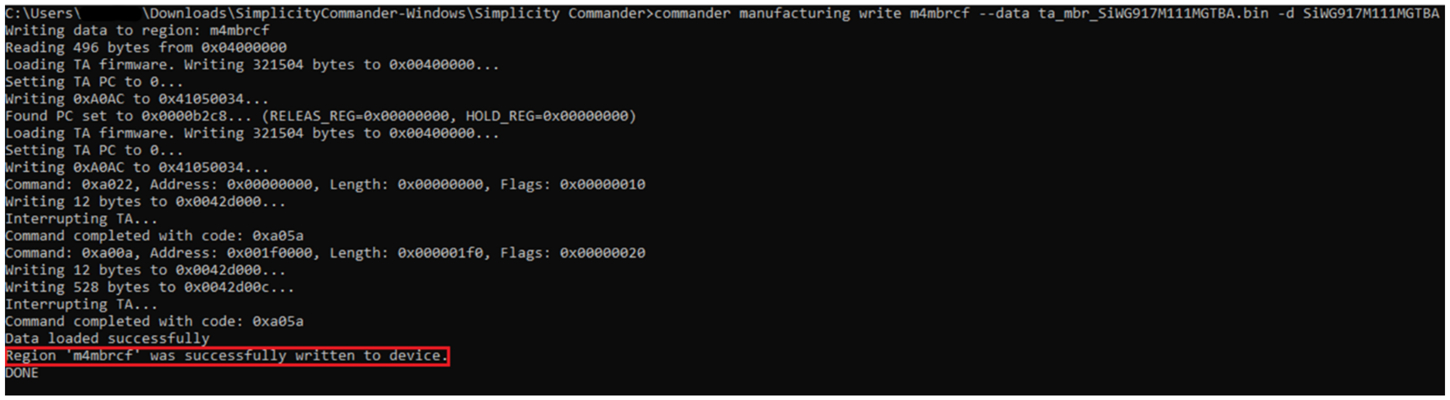
4.3.2 Write M4 MBR

TA and M4 regions have the same data in both their MBR regions. Pick the relevant binary files from section "5.2 MBR File(s)" and use the below command to update the M4 MBR

Note: User need to use same MBR for both TA and M4 thus in this case use **ta_mbr_SiWG917M1xxMGTBA.bin**

Command: `commander manufacturing write m4mbrcf --data <filename.bin> -d <full opn>`

Example: commander manufacturing write **m4mbrcf** --data **ta_mbr_SiWG917M1xxMGTBA.bin** -d **SiWG917M111MGTBA**



```
C:\Users\[redacted]\Downloads\SimplicityCommander-Windows\Simplicity Commander>commander manufacturing write m4mbrcf --data ta_mbr_SiWG917M111MGTBA.bin -d SiWG917M111MGTBA
Writing data to region: m4mbrcf
Reading 496 bytes from 0x04000000
Loading TA firmware. Writing 321504 bytes to 0x00400000...
Setting TA PC to 0...
Writing 0xA0AC to 0x41050034...
Found PC set to 0x0000b2c8... (RELEAS_REG=0x00000000, HOLD_REG=0x00000000)
Loading TA firmware. Writing 321504 bytes to 0x00400000...
Setting TA PC to 0...
Writing 0xA0AC to 0x41050034...
Command: 0xa022, Address: 0x00000000, Length: 0x00000000, Flags: 0x00000010
Writing 12 bytes to 0x0042d000...
Interrupting TA...
Command completed with code: 0xa05a
Command: 0xa00a, Address: 0x001f0000, Length: 0x000001f0, Flags: 0x00000020
Writing 12 bytes to 0x0042d000...
Writing 528 bytes to 0x0042d00c...
Interrupting TA...
Command completed with code: 0xa05a
Data loaded successfully
Region 'm4mbrcf' was successfully written to device.
DONE
```

4.3.3 Write the calibration data to the M4 Flash

Step 1: Copy the calibration data from TA to a bin file.
Step 2: Write the copied data to M4 Flash (the same bin file will be given as input).

Note: In case the above procedure fails please reset the board a few times and retry the steps.

4.3.3.1 Copy the Calibration data from TA to a bin file

Command: `commander manufacturing read taipmu --out <filename.bin>`

Example: commander manufacturing read **taipmu** --out **ipmu.bin**

4.3.3.2 Write the copied data to M4 Flash

Command: `commander manufacturing write m4ipmucf --data <file.bin> -d <full opn>`

Example: commander manufacturing write **m4ipmucf** --data **ipmu.bin** -d **SiWG917M111MGTBA**

Note:

- After the flashing is done, read the location 0x4000194 . It should return **1F.** Refer to section MBR Version check for steps to read the MBR version

- Update the TA firmware. Refer to Upgrade SiWx91x Connectivity Firmware. Else use the TA firmware image that is included in the SDK that you are using.

Note: In case the above procedure fails please reset the board a few times and retry the steps.

5 Configuring the M4 Application

Note: This is a mandatory section to follow. If not followed, the device will be corrupted and is irrecoverable.

Wi-Fi SDK 3.1.0 is the latest release that is out when this document is made. Users using 3.1.0 or older release(s) need to make some configuration changes to their projects. Unless this is done, the applications will not work with the updated MBR. **The next release is GA 3.1.1 will have this addressed by default.**

Here are the configuration changes that shall be made in the Application projects.

1. In **rsi_ipmu.h** file, update the following Macros with the corresponding addresses given below[i].

```
#define PACKAGE_TYPE_VALUES_OFFSET_COMMON_FLASH 0x81F0292

#define SILICON_REV_VALUES_OFFSET_COMMON_FLASH  0x81F0293

#define COMMON_FLASH_IPMU_VALUES_OFFSET 0x81F0258
```

Path to rsi_ipmu.h file: **wiseconnect3_sdk_3.1.0 > siwx917_soc > drivers > systemlevel > inc > rsi_ipmu.h**

2. Change the origin address of **rom** to **0x8202000** in the **linker_SoC.ld** file of the project.
   The linker file is available under **"autogen"** folder.

```
MEMORY
{
   rom    (rx)   : ORIGIN = 0x8202000, LENGTH = 0x6e000
   ram    (rwx)  : ORIGIN = 0xc, LENGTH = 0x30000
   psram  (rwx)  : ORIGIN = 0xa000000, LENGTH = 0x800000
}
```

   Note : For POWER SAVE related applications following address need to be changed in the preprocessor defines -

   IVT_OFFSET_ADDR = **136323072**

3. Clean and build the project and flash it in the device. The application should work as mentioned in its readme file.

Note:
Older addresses:

```
#define PACKAGE_TYPE_VALUES_OFFSET_COMMON_FLASH 0x81B0292
#define SILICON_REV_VALUES_OFFSET_COMMON_FLASH  0x81B0293
#define COMMON_FLASH_IPMU_VALUES_OFFSET         0x81B0258

rom   (rx)  : ORIGIN = 0x81c2000, LENGTH = 0x6e000
```

6 SiWx917 Board recovery

Due to 2 variants of boot loaders shipped to our customers as part of the alpha program, there is a possibility of MBR corruption that results in the board failure. The following two instances would result to MBR corruption on SiWx917 Board.

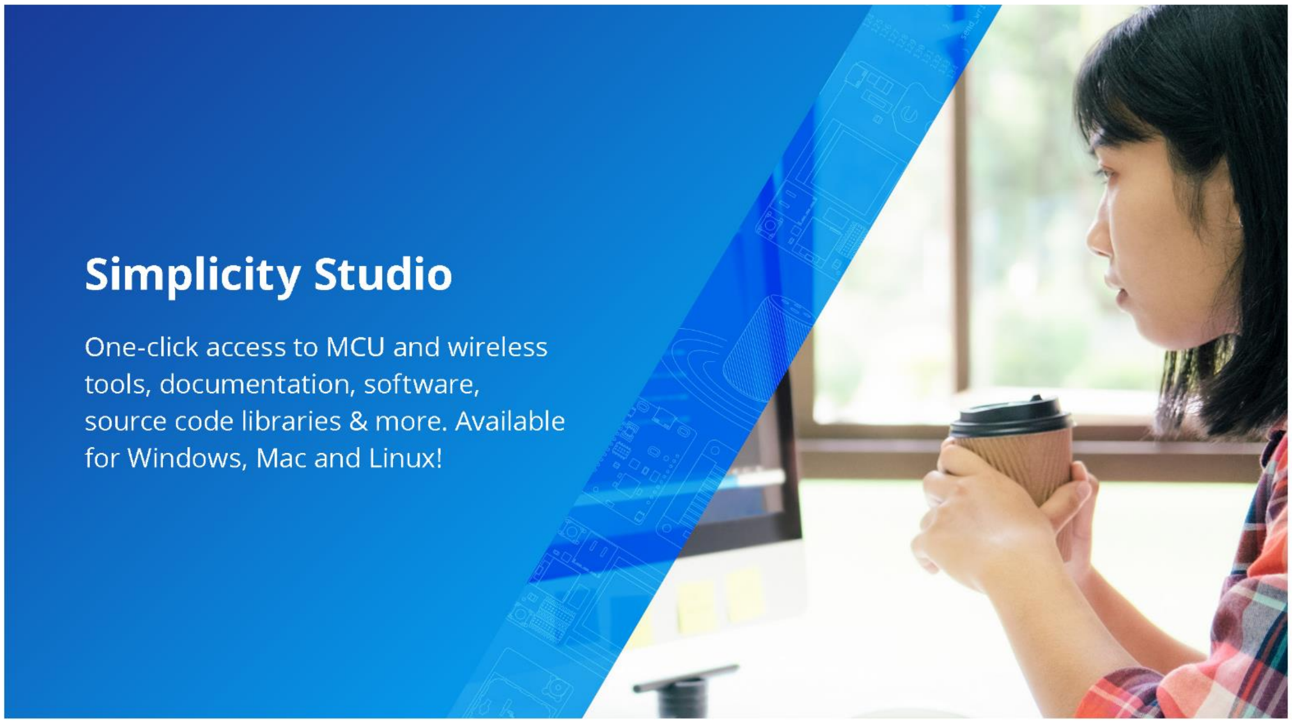
1. The BRD4338A board with 1.8v MBR, if the user tries to flash any matter application from https://docs.silabs.com/matter/2.1.1/matter-wifi-getting-started/ or any example from matter extension release version(2.1.1) and SMG release(2.2.0-1.2)
2. The BRD4338A board with 1.6v MBR, if the user tries to update/flash the firmware from version 2.9.0.0.30 or later.

Steps to follow to reflash the 1.8v MBR(recovery steps):

Download the MBR file : ta_mbr_SiWG917M1xxMGTBA.bin

Execute the below commands sequentially to reflash the MBR.

1. Commander manufacturing provision –mbr ta_mbr_SiWG917M1xxMGTBA.bin -d SiWG917M111MGTBA
   Example: *commander manufacturing provision --mbr ta_mbr_SiWG917M111MGTBA.bin -d SiWG917M111MGTBA*
2. commander manufacturing write m4mbrcf --data <filename.bin> -d SiWG917M111MGTBA
   Example: *commander manufacturing write m4mbrcf --data ta_mbr_SiWG917M111MGTBA.bin -d SiWG917M111MGTBA*
3. *commander manufacturing read taipmu --out filename.bin*
4. *commander manufacturing write m4ipmucf --data filename.bin*


Simplicity Studio
One-click access to MCU and wireless tools, documentation, software, source code libraries & more. Available for Windows, Mac and Linux!

IoT Portfolio
www.silabs.com/IoT

SW/HW
www.silabs.com/simplicity

Quality
www.silabs.com/quality

Support & Community
www.silabs.com/community

Silicon Laboratories Inc.
400 West Cesar Chavez
Austin, TX 78701
USA

www.silabs.com