MAX78000 Secure Bootloader In-Application Programming with Python® User Guide

UGxxxx, Rev 4.2; 2/24

Abstract

This user guide details how to update the end-user software application in the MAX78000 through the in-application programming, plus how to program the host code into the MAX32630FTHR board. Details on the MAX78000 secure bootloader can be found in the **MAX78000 Secure Bootloader User Guide**.

Python is a registered trademark of the Python Software Foundation.

Table of Contents

List of Figures

List of Tables

Introduction

This application note provides the instructions to program example host code into the MAX32630FTHR development platform.

The document also gives details related to hardware setup and application programming by using the MAX78000 bootloader and example host code.

Note that the screenshots may differ according to the software versions, but the steps will be the same.

System Requirements

To program the MAX78000 bootloader code into the MAX78000FTHR, the minimum requirements are as follows:

- Windows® PC

    - Windows 10, Windows 7

    - OpenSSL

    - Maxim Toolchain Software (more information, including download and installation instructions, is in this document)

- MAX78000FTHR and micro-USB cable

- MAX32630FTHR and micro-USB cable

- MAX32625PICO evaluation kit (EV kit) and micro-USB cable

- Test wires to connect the MAX78000FTHR and MAX32630FTHR

Maxim Toolchain Installation

To install the Maxim Toolchain to your PC, use the following steps:

1. Download the Maxim SDK **here**.

2. After downloading is complete, double-click **MaximMicrosSDK.exe** and use the default settings and select **Next** until finished.

Environment Setup

Copy the release package content to your PC under the *C:\ MAX78000_MSBL* folder.

Programming the MAX32630FTHR

To program example host code into the MAX32630FTHR, use the following steps:

1. Connect the grey 10-pin connector to the MAX32630FTHR and the MAX32625PICO board.

2. Connect the micro-USB cable to the MAX32625PICO and the PC.

3. Connect the micro-USB cable to the MAX32630FTHR and the PC.

Figure 1. The MAX32630FTHR and MAX32625PICO board connection.

4. Wait a few minutes for the Windows driver to install, then verify that it is installed correctly.

    a. In the Windows 10 search box, type **Control Panel** (or for Windows 7, click **Control Panel** on the right side of the **Start Menu**). Either click **Hardware and Sound**, then **Device Manager**, or type **Device Manager** in the search box in the upper right.

    b. If the drivers have correctly installed, you should see one port listed as **mbed Serial Port** for the MAX32625PICO. Note the COM port number for the USB serial device.
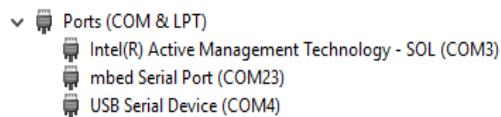

Figure 2. Serial port list.

    c. If you see the following, then you will need to install the correct Windows driver:
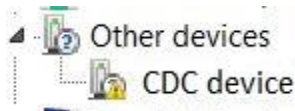
Figure 3. CDC device driver warning.

    d. Download the Arm Mbed® Windows serial port driver **here**.

e. For Windows 10, run *mbedWinSerial_16466.exe* by double-clicking it.

5. Drag and drop *max32630fthr-host-vx.x.x.bin* into the correct DAPLINK drive. This binary file resides in the MAX78000_MSBL directory.

6. Press the reset button on the MAX32630FTHR, as shown in **Figure 4**.

Figure 4. MAX32630FTHR host reset button.

7. Verify that the LED on the MAX32630FTHR is blinking, as shown in **Figure 5**.

Figure 5. MAX32630FTHR host blinking LED.

Hardware Setup

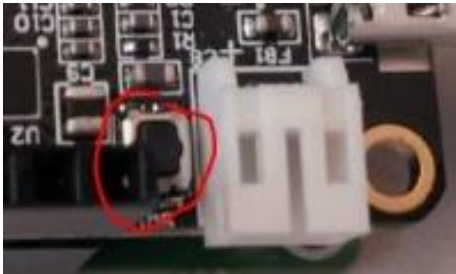Connect the MAX32630FTHR and MAX78000 with test wires according to **Table 1.** Pin connection between MAX32630FTHR and MAX78000FTHR is given in **Figure 6**.

Table 1. Pin Connection between the MAX32630FTHR and MAX78000FTHR

PIN FUNCTION	MAX78000FTHR	MAX32630FTHR
EBL GPIO	P0.5	P5.4
I2C1_SCL	P0.16	P3.5 + 4.7K pullup
I2C1_SDA	P0.17	P3.4 + 4.7K pullup
RST	RSTN	P5.6
GND	GND	GND

Note that R6 and R11 slots can be used for pull-up resistors that are not placed on default for the MAX32630FTHR board.

On the other hand, see the document "Maxim_Bootloader_Tools_UG .pdf" to use MAX32625PICO2 or a custom board such as MAX32665 EvKit as the host device instead of MAX32630FTHR board
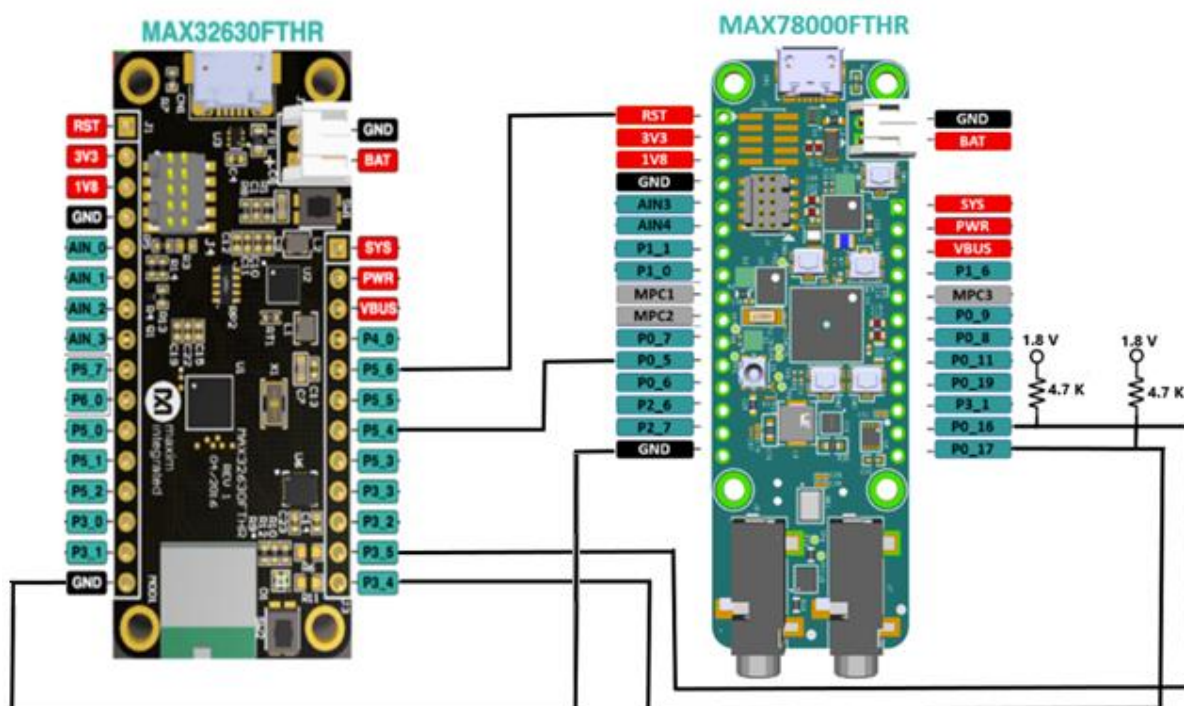
Figure 6. Pin Connection between the MAX32630FTHR and MAX78000FTHR

In-Application Programming

Installing Microsoft Visual C++ Runtime

To download and install Microsoft Visual C++ Runtime, use the following steps:

1. Visit the download link **here** and click Download button.
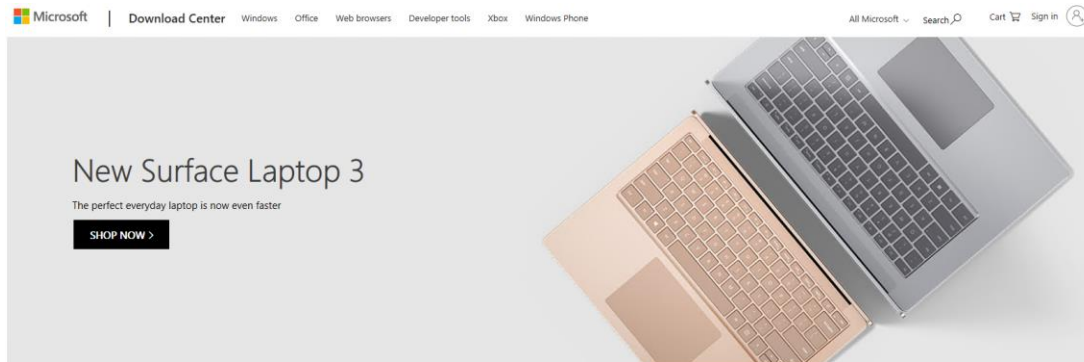
2. Select vcredist_x86.exe and continue by selecting Next.
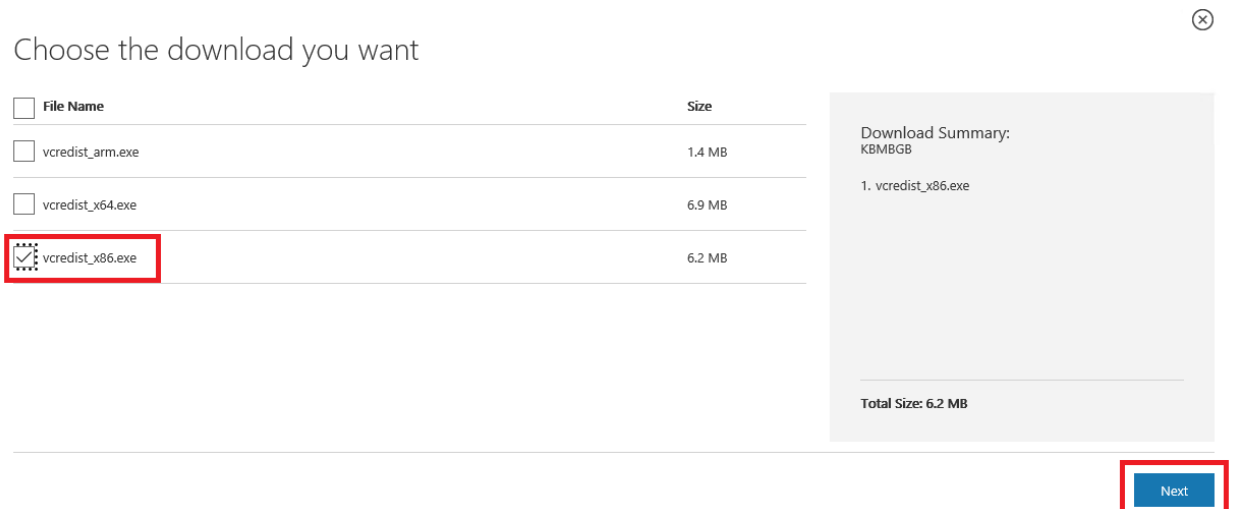
3. Download and run the setup file.

4. Read and click if you agree with the terms. Then select Install.
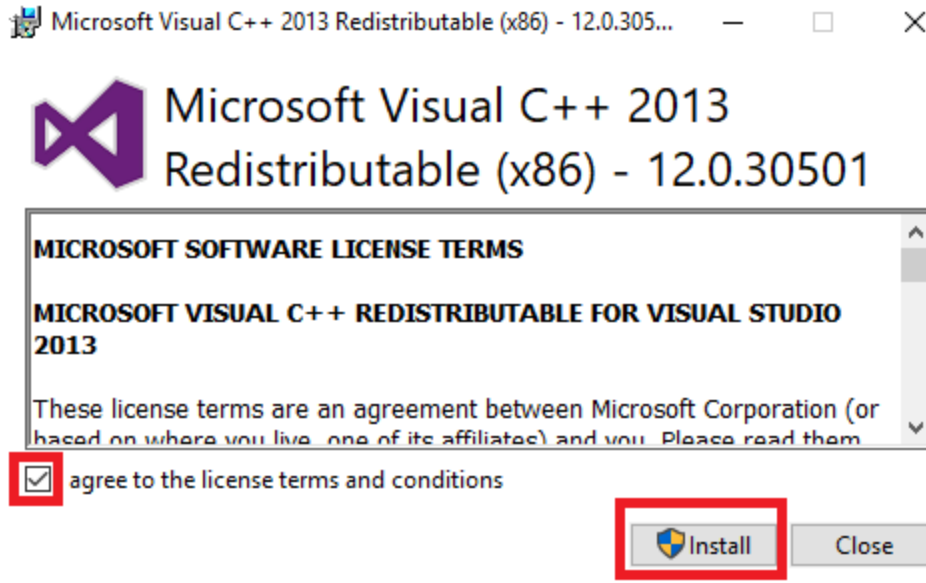
5. Close the installation application.

Installing OpenSSL Library

To download and install OpenSSL Library, use the following steps:

1. Visit OpenSSL downloads page **here**.

2. Select **Win32 OpenSSL v1.1.10L Light** package.

Download Win32/Win64 OpenSSL today using the links below!
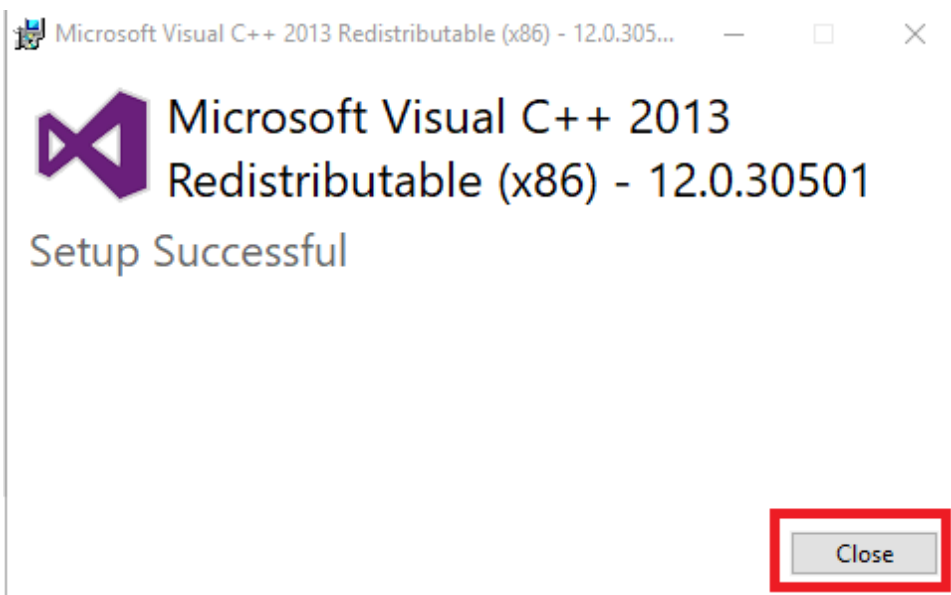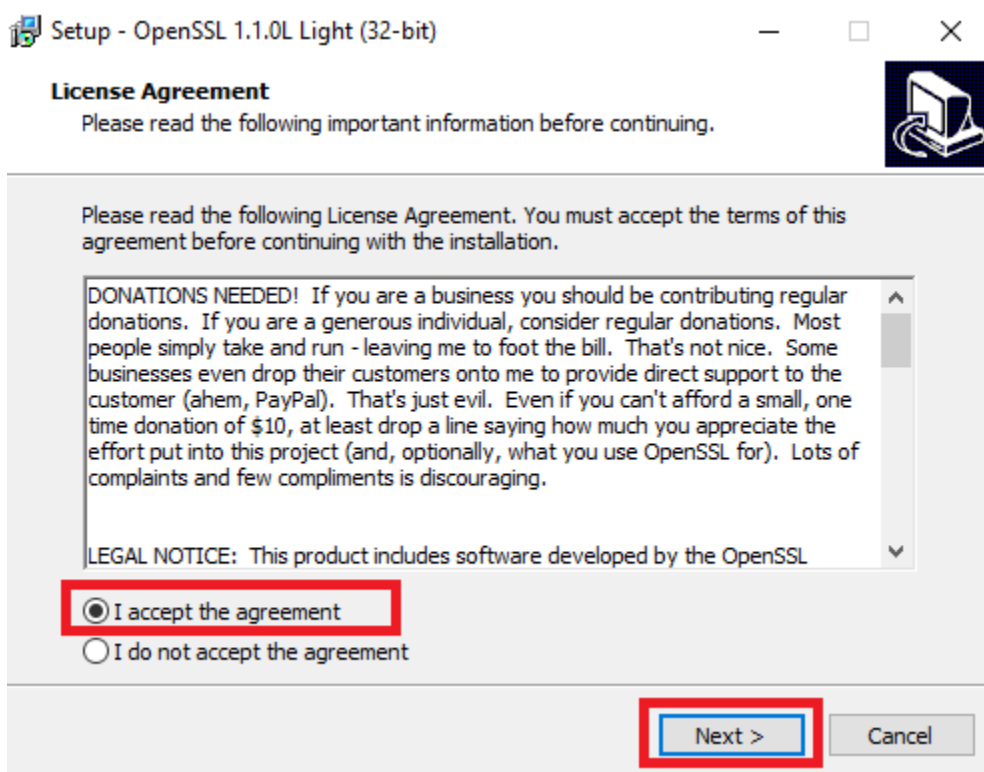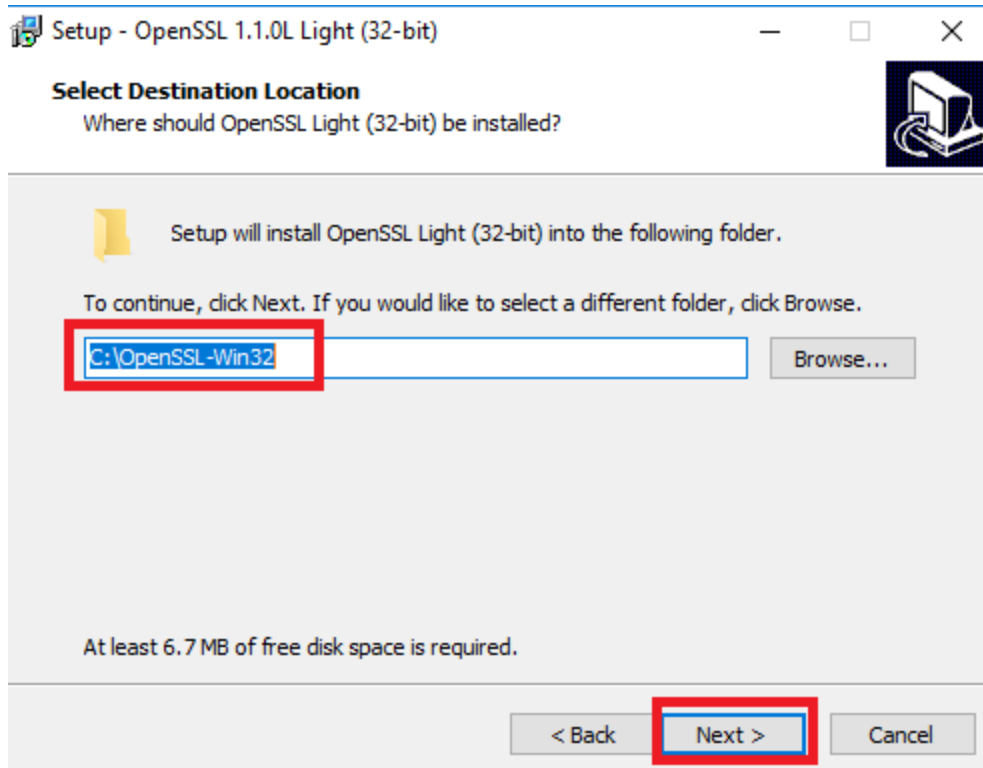
File	Type	Description
Win64 OpenSSL v1.1.1f Light EXE | MSI (experimental)	3MB Installer	Installs the most commonly used build of OpenSSL and is subject t
Win64 OpenSSL v1.1.1f EXE | MSI (experimental)	63MB Installer	Installs Win64 OpenSSL v1.1.1f (subject to local and state laws. M
Win32 OpenSSL v1.1.1f Light EXE | MSI (experimental)	3MB Installer	Installs the most commonly used state laws. More information can
Win32 OpenSSL v1.1.1f EXE | MSI (experimental)	54MB Installer	Installs Win32 OpenSSL v1.1.1f (found in the legal agreement of th
Win64 OpenSSL v1.1.0L Light	3MB Installer	Installs the most commonly used build of OpenSSL and is subject t
Win64 OpenSSL v1.1.0L	37MB Installer	Installs Win64 OpenSSL v1.1.0L subject to local and state laws. M
Win32 OpenSSL v1.1.0L Light	3MB Installer	Installs the most commonly used state laws. More information can
Win32 OpenSSL v1.1.0L	30MB Installer	Installs Win32 OpenSSL v1.1.0L laws. More information can be fou
Win64 OpenSSL v1.0.2u Light	3MB Installer	Installs the most commonly used build of OpenSSL and is subject t
Win64 OpenSSL v1.0.2u	23MB Installer	Installs Win64 OpenSSL v1.0.2u subject to local and state laws. M
Win32 OpenSSL v1.0.2u Light	2MB Installer	Installs the most commonly used state laws. More information can
Win32 OpenSSL v1.0.2u	20MB Installer	Installs Win32 OpenSSL v1.0.2u laws. More information can be fou

3. Download and run the installer.

4. Read and click if you agree with the terms. Then select Next.

Setup - OpenSSL 1.1.0L Light (32-bit)

License Agreement
Please read the following important information before continuing.

Please read the following License Agreement. You must accept the terms of this agreement before continuing with the installation.

DONATIONS NEEDED! If you are a business you should be contributing regular donations. If you are a generous individual, consider regular donations. Most people simply take and run - leaving me to foot the bill. That's not nice. Some businesses even drop their customers onto me to provide direct support to the customer (ahem, PayPal). That's just evil. Even if you can't afford a small, one time donation of $10, at least drop a line saying how much you appreciate the effort put into this project (and, optionally, what you use OpenSSL for). Lots of complaints and few compliments is discouraging.

LEGAL NOTICE: This product includes software developed by the OpenSSL

◉ I accept the agreement
○ I do not accept the agreement

Next >   Cancel

5. Leave the destination location default and click Next.

6. Leave "**The Windows system directory**" selected and click Next.
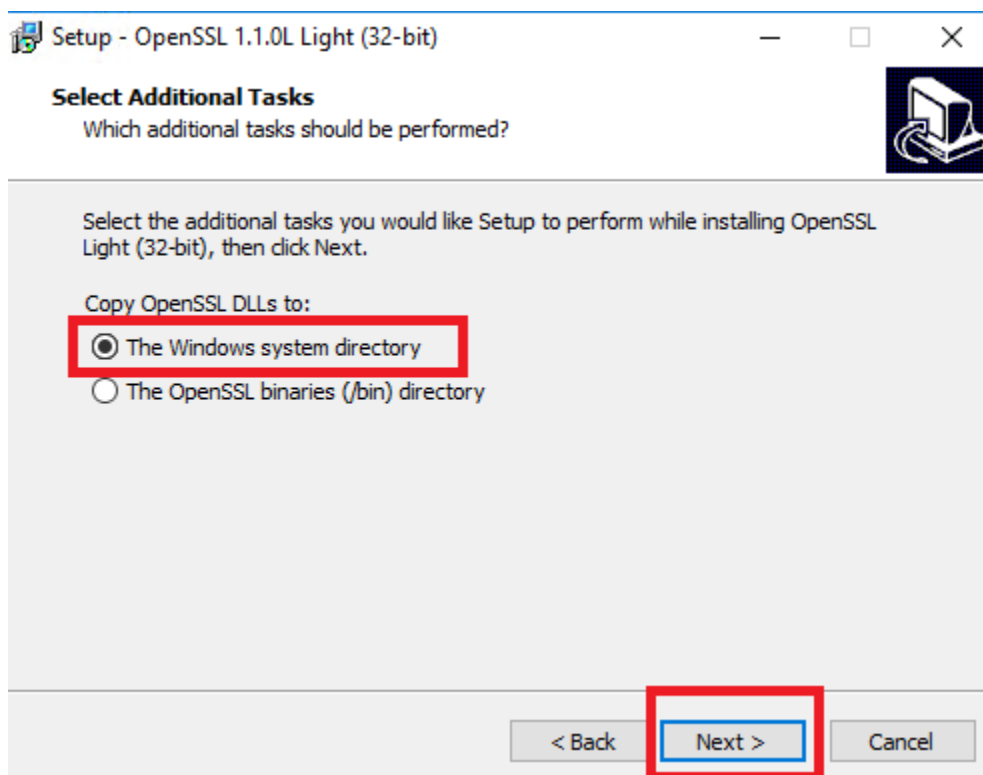
7. Click install.

8. When the installation is completed, click finish.

Compiling the Hello World Example in the Release Package with the Make Command

To compile the Hello World with the make command, use the following steps:

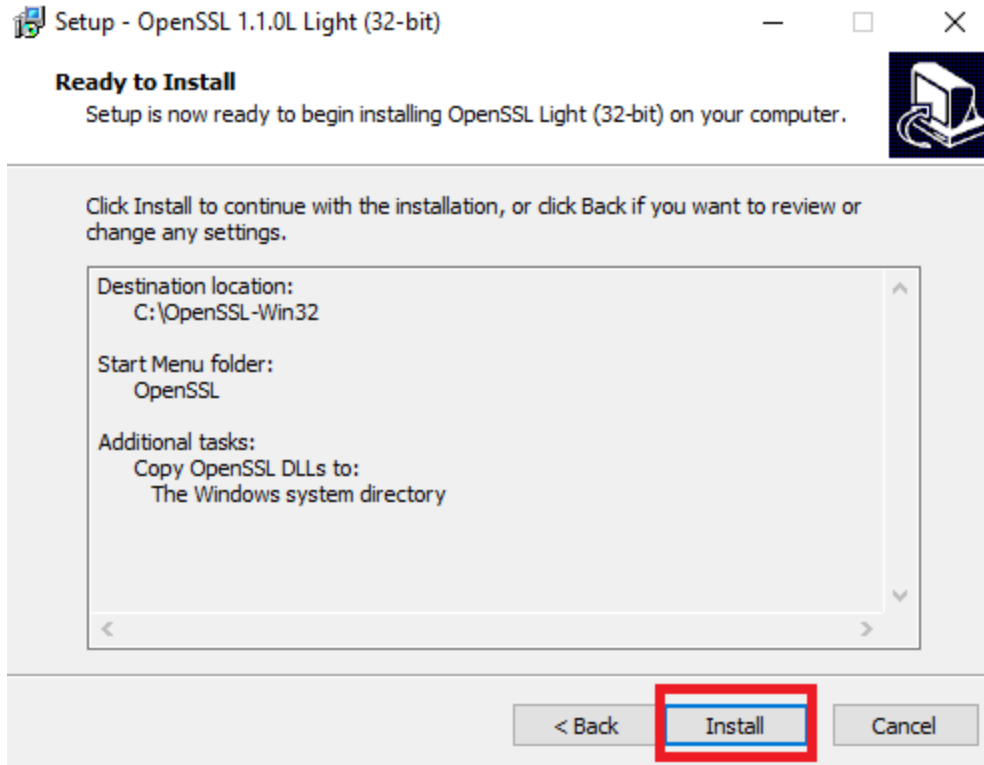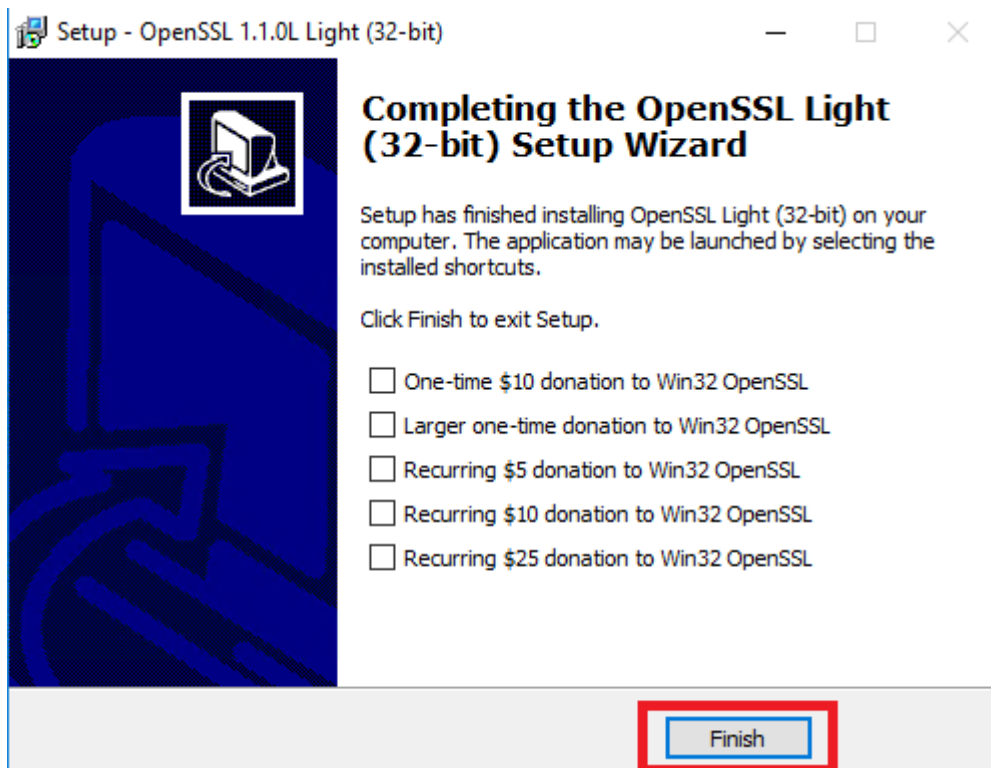1. Navigate to the correct directory consisting of the "Hello World" Example in the **MinGW** window with the following command:

```
cd " C:\MAX78000_MSBL \MAX78000_Hello_World"
```

2. Enter the following command in the **MinGW** window and wait several minutes for the command to complete:

```
make
```

3. After successful compiling, the Hello world binary image *max78000.bin* will be in the `"C:\MAX78000_MSBL \MAX78000_Hello_World\build"` directory.

Be sure that the correct linker file is used for generating the .bin file. A sample linker file, max78000_with_bl.ld, can be found under the Hello_World example folder.

4. If you want to rebuild, then enter these commands respectively:

```
make clean
make
```

Generating AES Keys

Generated AES keys can be used for encrypting bin files during msbl file generation and loading to the bootloader. Customer can use their own method for generating keys, but the file format should be similar to the sample max78000_sample_key.txt file.

1. Navigate to the correct directory consisting of the key generation script in the **MinGW** window with the following command:

```
cd "C:\MAX78000_MSBL"
```

2. Enter the following command in the **MinGW** window to generate a key:

```
./generate_key_vX.sh
```

Where "X" corresponds to the version the file.

3. Rename generated key.txt as *max78000_key.txt* file for using msbl file generation or bootloader key programming.

Generating Bootloader Compatible Binary Using Maxim SDK Project

Following steps should be applied for generating bootloader compatible binary using the Maxim SDK project:

1. Update your linker file:
    a. Copy max78000_with_bl.ld file which is located under *MAX78000_MSBL \MAX78000_Hello_World* to your Maxim SDK project.

2. Update the makefile
    a. Set the linker file variable (LINKERFILE) in makefile as below:

    Add following lines to project.mk:
    ```
    # Point this variable to a linker file to override the default file
    LINKERFILE=$(TARGET_LC)_with_bl.ld
    ```

    b. Add following lines to makefile under all targets for generating the necessary bin file:
    ```
    all:

        arm-none-eabi-objcopy $(BUILD_DIR)/$(PROJECT).elf -O binary
    $(BUILD_DIR)/$(PROJECT).bin
    ```

    Another way of generating bin files (using Eclipse):

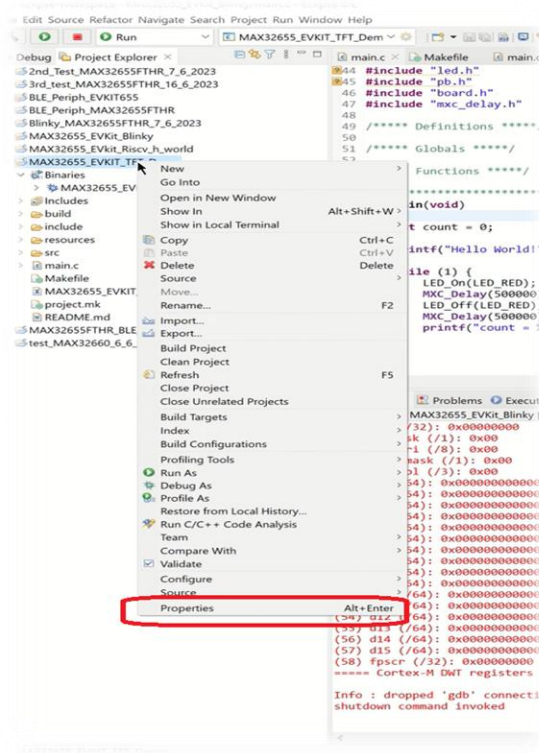   i. Go to the project explorer, right-click and select "Properties"

Figure 7 : Project properties of an Eclipse project

ii. From the left pane, choose the "C/C++ Build"
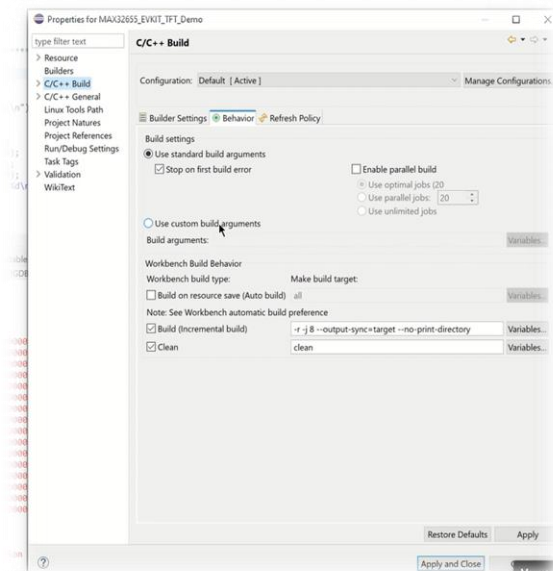iii. Select the "Behavior" tab.

Figure 8 : Behavior tab of C/C++ Build configurations

iv. Choose the "Use Custom Build Arguments"

v. Into the Build arguments field, type "release" (without the quotes and all lowercase.).
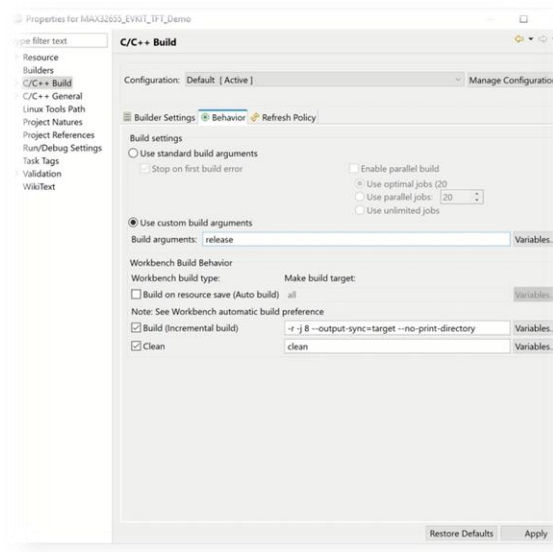
vi. Click "Apply", then "Apply and Close".

Figure 9 : Custom build argument configuration

3. Build the project.
4. .bin file is generated under build folder.

Converting the .bin File to the .msbl File Format

The .msbl file is generated automatically by using a .msbl generator.

1. Navigate to the directory consisting of msbl generator and the .bin file to be converted in the **MinGW** window.

2. Enter the following command in the **MinGW** window to convert the .bin (max78000.bin as an example) application to a .msbl file:

./msblGenWin32.exe max78000.bin MAX78000 8192 max78000_key.txt

3. "*max78000.msbl" file will be generated*

Note that the msblGenWin32.exe, the binary file, and the key text file might be in different directories. So, don't forget to give the path of them individually using the command in step 2.

Programming Keys to the Bootloader

To program the keys to the MAX78000 by using the MAX78000 bootloader, use the following steps:

At the command prompt or in the PowerShell window, enter the following commands, replacing COMxx with the correct USB serial device COM port found in section Programming the MAX32630FTHR at step 4.b. The "download_fw_over_host.exe" executable file resides in the MAX78000_MSBL directory.

./download_fw_over_host.exe -k max78000_key.txt -p COMXX

Note that the shared keys are sample one and they should be updated accordingly before programming keys. Moreover, the key cannot be updated unless SWD lock/unlock procedure and this procedure can be processed up to 4 times.

MSBL File Programming

To flash the application to the MAX78000 by using the MAX78000 bootloader, use the following steps:

At the command prompt or in the PowerShell window, enter the following commands, replacing COMxx with the correct USB serial device COM port found in section Programming the MAX32630FTHR at step 4.b.

./download_fw_over_host.exe -f MAX78000_Hello_World.msbl -p COMXX

Figure 10. Downloading firmware with the download_fw_over_host.exe

9. The application has now been flashed to the chip. For advanced usage of flasher script, refer to the *Firmware_downloader_usage.txt* file.

Locking the SWD Interface

To lock the SWD Interface, use the following steps:

1. At file max78000_bl_config.cfg: Change the swd_lock value to 1 and save the file.

```
[BootConfig]
# Enter Bootloader Pin Check 0: Disable,  1: Enable
enter_bl_check = 0
# Enter Bootloader Pin [0,13]
ebl_pin = 5
# Enter Bootloader Pin Polarity 0: active low, 1: active high
ebl_pol = 0
# Valid Mark Check 0: Disable,  1: Enable
valid_mark_check = 1
# UART Interface Enable 0: Disable,  1: Enable
uart_enable = 1
# I2C Interface Enable 0: Disable,  1: Enable
i2c_enable = 1
# SPI Interface Enable 0: Disable,  1: Enable
spi_enable = 1
# I2C Slave Addr Valid Between 8 - 119
i2c_addr = 85
# CRC Check 0: Disable,  1: Enable
crc_check = 0
# Timeout to exit bootloader
ebl_timeout = 0
# Timeout mode 0 : 'Jump immediately', 1 : 'Wait for programmable delay', 2 : 'remain in bootloader until receive exit command'
exit_bl_mode = 1
# Set SWD Lock
swd_lock = 1
```

Figure 11. Bootloader Configuration File.

2. Run the following command:
   configure_bootloader.exe –f max78000_bl_config.cfg –p COMXX

3. Disconnect the micro-USB cable.
4. Reconnect the micro-USB cable. SWD is locked now.

Unlocking the SWD Interface

To unlock the SWD Interface, use the following steps:

1. At file max78000_bl_config.cfg: Change the swd_lock value to 0 and save the file.
2. Run the following command:

   ./configure_bootloader.exe –f max78000_bl_config.cfg –p COMXX

3. Disconnect the micro-USB cable.
4. Reconnect the micro-USB cable Now key and application are erased, SWD is unlocked, so a new key can be loaded.

   Note that SWD can only be locked and unlocked up to 4 times.

Revision History

REVISION NUMBER	REVISION DATE	DESCRIPTION	PAGES CHANGED
0	3/21	Initial release	—
1	3/21	• Programming the MAX78000 with the Bootloader Firmware section added • Pin Connection between the MAX32630FTHR and • MAX78000FTHR Figure added • Programming Keys to the Bootloader Section added	5, 8, 16
2	5/21	• SWD Lock and Unlock Sections are Added	16,17
3	8/21	• Generating Bootloader Compatible Binary Using Maxim SDK Project Section added	15
4	10/21	• Improve readability	—
4.1	11/21	• General Improvements • Update logo	
4.2	2/24	• General Improvements • Generating Bootloader Compatible Binary Using Maxim SDK Project Section updated	8, 14, 15

DEVICES, OR TECHNOLOGY DESCRIBED HEREIN OR OTHERWISE. The information contained within this document has been verified according to the general principles of electrical and mechanical engineering or registered trademarks of Maxim Integrated Products, Inc. All other product or service names are the property of their respective owners.