

# Getting Started with NXP MIMXRT1050-EVK and MIMXRT1060-EVK Boards

If you do not have any [MIMXRT1050-EVKB](#) or [MIMXRT1060-EVK](#) boards, visit the AWS Partner Device Catalog to purchase them from our partner, following links above. Use a USB cable to connect your MIMXRT1050-EVK or MIMXRT1060-EVK boards to your computer.

Before you begin, configure the AWS IoT and your Amazon FreeRTOS download to connect your device to the AWS Cloud. See [prerequisites](#) for instructions. In this tutorial, the path to the Amazon FreeRTOS download directory is called `BASE_FOLDER`.

## Hardware requirements

1. [MIMXRT1050-EVKB](#) or [MIMXRT1060-EVK](#) board
2. [Embedded Artists Type 1DX M.2](#) EVB Wi-Fi module
3. [Murata uSD-M.2 Adapter Kit](#)

## Setting Up Your Environment

Two toolchains are supported on the MIMXRT1050-EVK/MIMXRT1060-EVK boards: IAR Embedded Workbench and GNU Arm Embedded Toolchain.

Before you begin, install one of these IDEs.

### To install IAR Embedded Workbench for ARM:

4. Browse to [Software for NXP Kits](#) and select **Download Software**.

#### Note

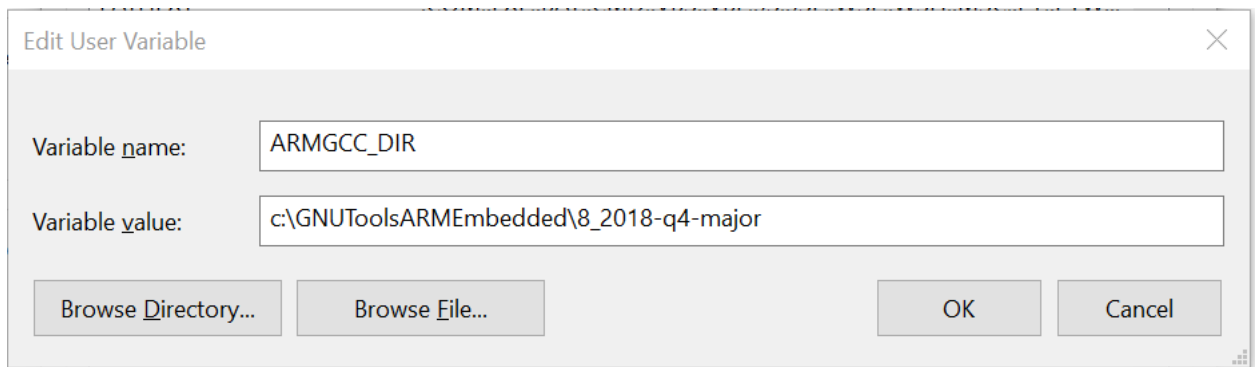
IAR Embedded Workbench for ARM requires Microsoft Windows.

5. Unzip and run the installer. Follow the prompts.
6. In **License Wizard**, select **Register with IAR Systems to get an evaluation license**.

### To install the GNU Arm Embedded Toolchain:

1. GNU Arm Embedded toolchain is supported for Windows and Linux OS.
2. Download and run the GNU Arm Embedded toolchain installer from <https://launchpad.net/gcc-arm-embedded> on Windows OS.  
Extract gcc-arm-none-eabi-8-2018-q4-major-linux.tar.bz2 archive on Linux OS.
3. Download and install cmake from <https://cmake.org/>.
4. Create the "ARMGCC\_DIR" environment variable which contains the GCC location on Windows OS.

For example:

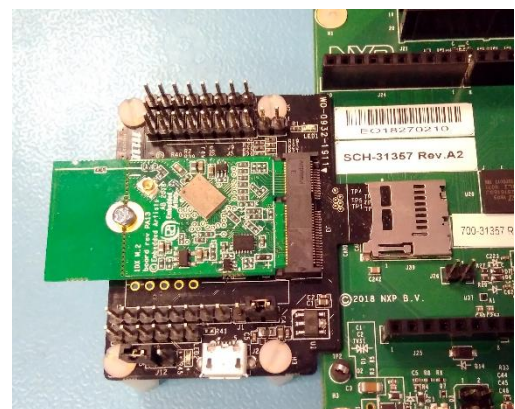


On Linux OS execute command “export ARMGCC\_DIR=~/.Downloads/ gcc-arm-none-eabi-8-2018-q4-major-linux” and append the command into ~/.bashrc file.

5. Builder scripts are located in "armgcc" project subdirectory. Run the "build\_flexspi\_nor\_debug.bat" on Windows OS and “source build\_flexspi\_nor\_debug.sh” on Linux OS.

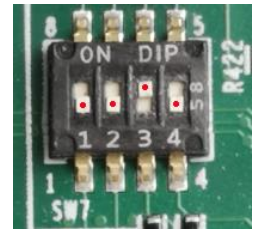
## Connecting Wi-Fi module

Plug the muRata Wi-Fi uSDAdapter to the micro-SD card slot on the MIMXRT1050-EVK or MIMXRT1060-EVK boards.

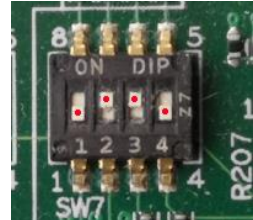


## Connecting JTAG Debugger

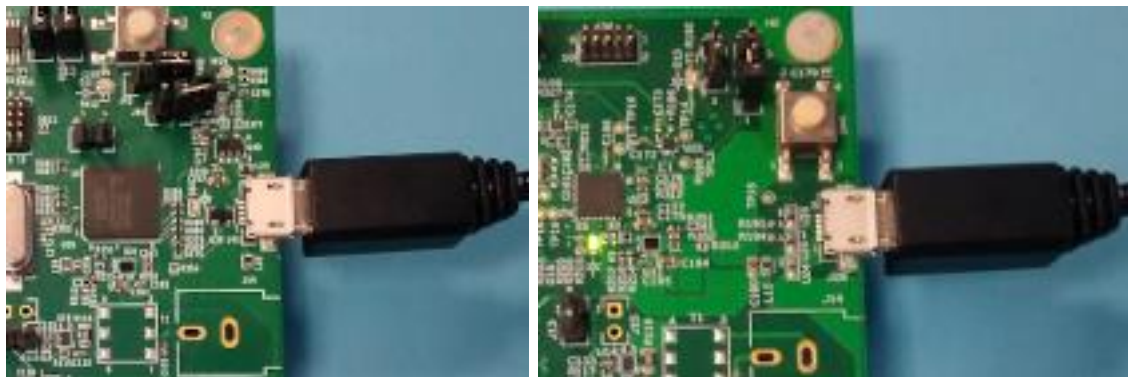
On MIMXRT1060-EVK set dip switch SW7 in position:  
OFF, OFF, ON, OFF to enable boot from QSPI Flash (U33).



On MIMXRT1050-EVK set dip switch SW7 in position:  
OFF, ON, ON, OFF to enable boot from HyperFlash (U19).



Plug a micro-USB cable to the debug USB port (J28 for MIMXRT1050-EVK, J41 for MIMXRT1060-EVK). Your PC OS should recognize and set up a new COM Port device that is used as a default UART for the development board. The debugger uses the preset CMSIS-DAP firmware. However, it is possible also to use J-Link OpenSDA - Board-Specific Firmware, available at <https://www.segger.com/downloads/jlink/>.



## Building and Running Amazon FreeRTOS Demo Project

### Import the Amazon FreeRTOS Demo into Your IDE

To import the Amazon FreeRTOS sample code into the IAR Embedded Workbench IDE:

1. Open IAR Embedded Workbench, and select **Open Workspace** from the **File** menu.
2. Enter `<BASE_FOLDER>\demos\nxp\<BOARD>\iar` into the **search-directory** text box and select **aws\_demos.eww**.
3. From the **Project** menu, select **Rebuild All**.

### Build the Amazon FreeRTOS sample code using CMake and the GNU Arm Embedded Toolchain.

1. Enter the `<BASE_FOLDER>\demos\nxp\<BOARD>\armgcc` project subdirectory and run the "build\_flexspi\_nor\_debug.bat" on Windows OS or "source build\_flexspi\_nor\_debug.sh" on Linux OS.

### Running Amazon FreeRTOS Demo Project

1. Configure the Amazon FreeRTOS Demo Project by following the prerequisite steps found here:
  - a. <https://docs.aws.amazon.com/freertos/latest/userguide/freertos-account-and-permissions.html>
  - b. <https://docs.aws.amazon.com/freertos/latest/userguide/get-started-freertos-thing.html>
  - c. <https://docs.aws.amazon.com/freertos/latest/userguide/freertos-configure.html>
2. Connect the USB port on the NXP board to your host computer, open a terminal program, and connect to the port identified as the USB Serial Device.
3. In your IDE, select **Build** from the **Project** menu.
4. If you are using IAR Embedded Workbench:
  - a. From the **Project** menu, select **Download and Debug**.
  - b. From the **Debug** menu, select **Start Debugging**.
  - c. When the debugger stops at the breakpoint in `main`, select **Go** from the **Debug** menu.

#### Note

If a **J-Link Device Selection** dialog box opens, select **OK** to continue.

5. If you are using the GNU Arm Embedded Toolchain with JLink:

- a. Download and install Ozone – The J-Link Debugger from <https://www.segger.com/downloads/jlink/#Ozone>.
  - b. Run Ozone and select "File > New Project Wizard" to download and debug the application. In the first step, set the "Target device" to "MIMXRT1052DVL6B" for the MIMXRT1050-EVK board or "MIMXRT1064xxx6A" for the MIMXRT1060-EVK board. In next step, set the "Target interface" to "SWD". In the last step, set the "Program File" to the application binary located in  
`<BASE_FOLDER>\demos\nxp\<BOARD>\armgcc\<TARGET>\aws_demos`  
and click "Finish".
  - c. Run "Debug > Start Debugging > Download and Reset Program".
6. When the debugger stops at the breakpoint in `main()`, select **Continue** from the **Debug** menu.

Use the MQTT client in the AWS IoT console to monitor the messages that your device sends to the AWS Cloud.

#### **To subscribe to the MQTT topic with the AWS IoT MQTT client:**

1. Sign in to the [AWS IoT console](#).
2. In the navigation pane, select **Test** to open the MQTT client.
3. In **Subscription topic**, enter `freertos/demos/echo`, and select **Subscribe to topic**.