**Getting Started with the Microchip SAME70 Xplained Ultra (Wired)**

This tutorial provides instructions for getting started with the Microchip SAME70 Xplained Ultra board. If you do not have the following components, visit the AWS Partner Device Catalog to purchase one from our partner.

* [SAME70 Xplained Ultra Evaluation Kit](https://www.microchip.com/Developmenttools/ProductDetails/DM320113)
* [LAN8720 PHY daughter board](http://www.microchip.com/DevelopmentTools/ProductDetails.aspx?PartNO=ac320004-3)

**Pre-Requisites:**

Before you begin, you must configure AWS IoT and your Amazon FreeRTOS download to connect your device to the AWS Cloud.

To get started with Amazon FreeRTOS, you need an AWS account, an IAM user with permission to access AWS IoT and Amazon FreeRTOS cloud services, and you need one of the supported hardware platforms. You also need to download Amazon FreeRTOS and configure your board's Amazon FreeRTOS demo project to work with AWS IoT. The following sections walk you through these requirements.

1. [Setting Up Your AWS Account and Permissions](https://docs.aws.amazon.com/freertos/latest/userguide/freertos-account-and-permissions.html)
2. [Registering Your MCU Board with AWS IoT](https://docs.aws.amazon.com/freertos/latest/userguide/get-started-freertos-thing.html)
3. From the [MicrochipTech repository](https://github.com/MicrochipTech/amazon-freertos/tree/mchpdev/), download the amazon-freertos repository (mchpdev branch). In this tutorial, the path to the Amazon FreeRTOS download directory is referred to as *<amazon-freertos>*.
4. [Configuring the Amazon FreeRTOS Demos](https://docs.aws.amazon.com/freertos/latest/userguide/freertos-configure.html)

**Important**

The maximum length of a file path on Microsoft Windows is 260 characters. To accommodate the files in the Amazon FreeRTOS projects, make sure that the path to the Root path directory is fewer than 40 characters long.

**Overview**

This tutorial contains instructions for the following getting started steps:

1. Connecting your board to a host machine.
2. Installing software on the host machine for developing and debugging embedded applications for your microcontroller board.
3. Cross compiling an Amazon FreeRTOS demo application to a binary image.
4. Loading the application binary image to your board, and then running the application.
5. Interacting with the application running on your board across a serial connection, for monitoring and debugging purposes.

## Set Up the Microchip SAME70 Xplained Ultra Hardware

1. Connect the DEBUG USB port of the board to the host computer using a Type A to Micro B USB Cable.
2. Connect the PIC32 LAN8720 PHY daughter board to the ETHERNET PHY MODULE header on the SAME70 Xplained Ultra Kit.
3. Connect one end of an Ethernet cable to the ethernet PHY available in the board, Connect the other end to your router or other internet port.

## Set Up Your Development Environment

**Note**

The Amazon FreeRTOS project for this device is based on MPLAB Harmony v3. To build the project, you need to use versions of the MPLAB tools that are compatible with Harmony v3, like v2.30 or higher of the MPLAB XC32 Compiler and the latest MPLAB IDE.

1. Install [Python version 3.x](https://www.python.org/downloads/) or later.
2. Install the MPLAB X IDE:
   * [MPLAB X Integrated Development Environment for Windows](http://www.microchip.com/mplabx-ide-windows-installer)
   * [MPLAB X Integrated Development Environment for macOS](http://www.microchip.com/mplabx-ide-osx-installer)
   * [MPLAB X Integrated Development Environment for Linux](http://www.microchip.com/mplabx-ide-linux-installer)
3. Install the MPLAB XC32 Compiler:
   * [MPLAB XC32/32++ Compiler for Windows](http://www.microchip.com/mplabxc32windows)
   * [MPLAB XC32/32++ Compiler for macOS](http://www.microchip.com/mplabxc32osx)
   * [MPLAB XC32/32++ Compiler for Linux](http://www.microchip.com/mplabxc32linux)
4. Start up a UART terminal emulator and open a connection with the following settings:
   * Baud rate: 115200
   * Data: 8 bit
   * Parity: None
   * Stop bits: 1
   * Flow control: None

## Build and Run Amazon FreeRTOS Demo Project

### Open the Amazon FreeRTOS Demo in the MPLAB IDE

1. Open MPLAB IDE. If you have more than one version of the compiler installed, you need to select the compiler that you want to use from within the IDE.
2. From the **File** menu, choose **Open Project**.
3. Browse to and open

projects\microchip\same70\_xult\mplab\aws\_demos\firmware\ aws\_demos.X.

1. Choose **Open project**.

**Note**

When you open the project for the first time, you might get an error message about the compiler. In the IDE, navigate to **Tools**, **Options**, **Embedded**, and then select the compiler that you are using for your project. The demo kit comes with on-board EDBG debugger/programmer, which would be used for Programming/Debugging the Amazon FreeRTOS project.

### Run the Amazon FreeRTOS Demo Project

1. Rebuild your project.
2. On the **Projects** tab, right-click the aws\_demos top-level folder, and then choose **Debug**.
3. When the debugger stops at the breakpoint in main(), from the **Run** menu, choose **Resume**.

### Monitoring MQTT Messages on the Cloud

You can 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](https://console.aws.amazon.com/iotv2/).
2. In the navigation pane, choose **Test** to open the MQTT client.
3. In **Subscription topic**, enter **iotdemo/#**, and then choose **Subscribe to topic**.

## Troubleshooting

For general troubleshooting information about Getting Started with Amazon FreeRTOS, see [Troubleshooting Getting Started](https://docs.aws.amazon.com/freertos/latest/userguide/gsg-troubleshooting.html).