

**Windows 10 IoT Core for
Maaxboard
V1.0
07 July 2020**

© 2020 Avnet. All rights reserved. All trademarks and registered trademarks are the property of their respective owners. All specifications are subject to change without notice.

NOTICE OF DISCLAIMER: Avnet is providing this design, code, or information "as is." By providing the design, code, or information as one possible implementation of this feature, application, or standard, Avnet makes no representation that this implementation is free from any claims of infringement. You are responsible for obtaining any rights you may require for your implementation. Avnet expressly disclaims any warranty whatsoever with respect to the adequacy of the implementation, including but not limited to any warranties or representations that this implementation is free from claims of infringement and any implied warranties of merchantability or fitness for a particular purpose.

Table of Contents

1	Windows 10 IoT Core for NXP i.MX Processors.....	3
2	Board List	3
3	Getting Ready.....	3
3.1	Cloning the Repository	3
3.2	Required Tools	3
3.2.1	Visual Studio 2017	3
3.2.2	Windows Kits from Windows 10, version 1809	4
3.2.3	IoT Core OS Packages	4
3.3	One-Time Environment Setup.....	4
4	FFU Generation	5
4.1	Building the FFU for other boards	5
4.2	Installing to an SD Card	5
4.3	Adding a New Driver	6
4.4	Building the FFU with the IoT ADK AddonKit.....	6
5	Directories	6
6	Info.....	6
7	Additional Support	6
8	Revision History.....	6

1 Windows 10 IoT Core for NXP i.MX Processors

Important! Please read this section first.

This code is provided as a public preview, it is still under development which means not all platform features are enabled or fully optimized. Notwithstanding the license attached to this code, should not be used in any commercial application at this time. For any questions and feedback on how the BSP can better support your targeted solution, please contact your Microsoft or NXP representative or post to the [NXP community](#).

This code is available under the [MIT](#) license except where stated otherwise such as the `imxnetmini` driver and `OpteeClientLib`.

2 Board List

SoC Type	Board Vendor	Board Name	Board Package Name
i.MX 8M	AVNET	MaaxBoard	MaaxBoard_iMX8M_2GB
i.MX 8M	Embest	EM-MC-SBC	EMMCSBC_iMX8M_2GB

A table of the currently enabled features for each board can be found [here](#). For hardware issues, please contact the hardware vendor.

3 Getting Ready

3.1 Cloning the Repository

This repository uses submodules and should be cloned with `git clone --recurse-submodules https://github.com/Witekio/imx-iotcore`

3.2 Required Tools

The following tools are required to build the driver packages and IoT Core FFU: Visual Studio 2017, Windows Kits (ADK/SDK/WDK), and the IoT Core OS Packages.

3.2.1 Visual Studio 2017

- Make sure that you **install Visual Studio 2017 before the WDK** so that the WDK can install a required plugin.
- Download [Visual Studio 2017](#).
- During install select **Desktop development with C++**.
- During install select the following in the Individual components tab. If these options are not available try updating VS2017 to the latest release:
 - **VC++ 2017 version 15.9 v14.16 Libs for Spectre (ARM)**
 - **VC++ 2017 version 15.9 v14.16 Libs for Spectre (ARM64)**
 - **VC++ 2017 version 15.9 v14.16 Libs for Spectre (X86 and x64)**
 - **Visual C++ compilers and libraries for ARM**

- **Visual C++ compilers and libraries for ARM64**

3.2.2 Windows Kits from Windows 10, version 1809

- **IMPORTANT: Make sure that any previous versions of the ADK and WDK have been uninstalled!**
- Install [ADK version 1809](#)
- Install [WDK version 1809](#)
 - Make sure that you allow the Visual Studio Extension to install after the WDK install completes.
- If the WDK installer says it could not find the correct SDK version, install [SDK version 1809](#)

3.2.3 IoT Core OS Packages

- Visit the [Windows IoT Core Downloads](#) page and download "Windows 10 IoT Core Packages – Windows 10 IoT Core, version 1809 (LTSC)".
- Open the iso and install `Windows_10_IoT_Core_ARM_Packages.msi`
- Install `Windows_10_IoT_Core_ARM64_Packages.msi` for ARM64 builds.

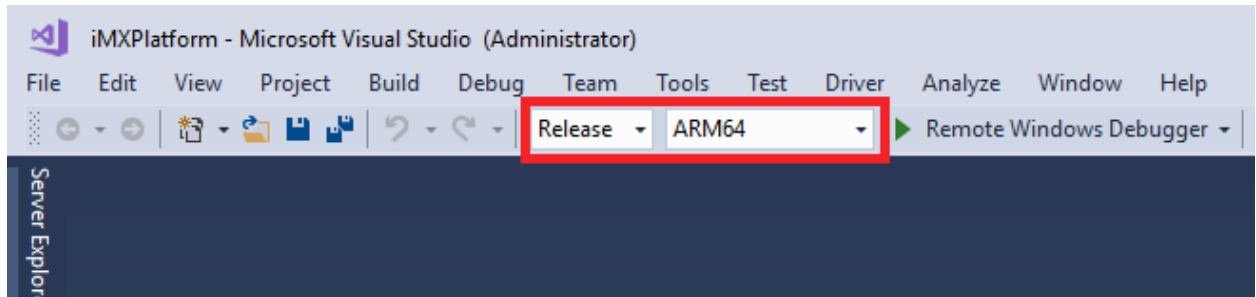
3.3 One-Time Environment Setup

Test certificates must be installed to generate driver packages on a development machine.

1. Open an Administrator Command Prompt.
2. Navigate to your newly cloned repo and into the folder `imx-iotcore\build\tools`.
3. Launch `StartBuildEnv.bat`.
4. Run `SetupCertificate.bat` to install the test certificates.
5. Make sure that submodules have been cloned. If you cloned with `--recurse-submodules` then this step won't output anything.
6. `git submodule init`
7. `git submodule update`

4 FFU Generation

1. Launch Visual Studio 2017 as **Administrator**.
2. Open the solution iMXPlatform.sln (imx-iotcore\build\solution\iMXPlatform).
3. Change the build type from Debug to Release. Change the build flavor from ARM to ARM64 if building for iMX8.



4. To build press Ctrl-Shift-B or choose Build -> Build Solution from menu. This will compile all driver packages then generate the FFU.
5. Depending on the speed of the build machine FFU generation may take around 10-20 minutes.
6. After a successful build the new FFU will be located in imx-iotcore\build\solution\iMXPlatform\Build\FFU\HummingBoardEdge_iMX6Q_2GB\ for ARM builds and imx-iotcore\build\solution\iMXPlatform\Build\FFU\MaaxBoard_iMX8M_2GB for ARM64 builds.
7. The FFU contains firmware components for the HummingBoard Edge with the Quad Core SOM or MaaxBoard iMX8M 2GB with i.MX8M Quad Core SOM depending on build flavor. This firmware is automatically applied to the SD Card during the FFU imaging process.

4.1 Building the FFU for other boards

In order to build an FFU for another board you'll need to modify GenerateFFU.bat in the Build Scripts folder of the Solution Explorer. Comment out the default HummingBoardEdge_iMX6Q_2GB or MaaxBoard_iMX8M_2GB builds with REM and uncomment any other boards you want to build.

```
REM cd /d %BATCH_HOME%
REM echo "Building MaaxBoard_iMX8M_2GB FFU"
REM call BuildImage MaaxBoard_iMX8M_2GB MaaxBoard_iMX8M_2GB_TestOEMInput.xml

cd /d %BATCH_HOME%
echo "Building EMMCSBC_iMX8M_2GB FFU"
call BuildImage EMMCSBC_iMX8M_2GB EMMCSBC_iMX8M_2GB_TestOEMInput.xml
```

4.2 Installing to an SD Card

- Follow the instructions in the [IoT Core Manufacturing Guide](#) to flash the FFU to an SD Card using the Windows IoT Core Dashboard.
- Do not specify a Wi-Fi network as the MaaXBoard only supports Ethernet at this time
- In the order connect up your hardware after the SD Card is flashed.
 - Insert SD Card
 - Plug HDMI Cable to HDMI monitor

- Power the MaaXBoard using a USB type C power supply
- Plug in your Ethernet Cable

4.3 Adding a New Driver

- Follow the instructions in the [Adding a New Driver](#) document.

4.4 Building the FFU with the IoT ADK AddonKit

1. Build the GenerateBSP project to create a BSP folder in the root of the repository.
 2. Clone the [IoT ADK AddonKit](#).
 3. Follow the [Create a basic image instructions](#) from the IoT Core Manufacturing guide with the following changes.
- When importing a BSP use one of the board names from the newly generated BSP folder in the imx-iotcore repo.
 - `Import-IoTBSP MaaxBoard_iMX8M_2GB <Path to imx-iotcore\BSP>`
 - When creating a product use the same board name from the BSP import.
 - `Add-IoTProduct ProductA MaaxBoard_iMX8M_2GB`

5 Directories

- BSP - Generated at build time. Contains Board Support Packages for the IoT ADK AddonKit.
- build - Contains Board Packages, build scripts, and the VS2017 solution file.
- driver - Contains driver sources.
- documentation - Contains usage documentation.
- hal - Contains hal extension sources.

6 Info

For more information about Windows 10 IoT Core, see Microsoft online documentation [here](#)

We are working hard to improve Windows 10 IoT Core and deeply value any feedback we get.

This project has adopted the [Microsoft Open Source Code of Conduct](#). For more information see the [Code of Conduct FAQ](#) or contact opencode@microsoft.com with any additional questions or comments.

7 Additional Support

To find support for your MaaXBoard and to access all support material, please visit:

<http://avnet.me/maaxboard>

8 Revision History

Date	Version	Revision
07 July 2020	1.0	Initial Release