

Intel® HD Graphics V36.19.0 Gold for Windows 10* IoT Core for Intel® Atom™ processor E3800 Family/Intel® Celeron® Processor N2807/N2900/J1900

User Guide

December 2015

***Gold Release
Revision 1.0***



You may not use or facilitate the use of this document in connection with any infringement or other legal analysis concerning Intel products described herein. You agree to grant Intel a non-exclusive, royalty-free license to any patent claim thereafter drafted which includes subject matter disclosed herein

No license (express or implied, by estoppel or otherwise) to any intellectual property rights is granted by this document.

All information provided here is subject to change without notice. Contact your Intel representative to obtain the latest Intel product specifications and roadmaps.

The products described may contain design defects or errors known as errata which may cause the product to deviate from published specifications. Current characterized errata are available on request.

Copies of documents which have an order number and are referenced in this document may be obtained by calling 1-800-548-4725 or by visiting: <http://www.intel.com/design/literature.htm>

Intel technologies' features and benefits depend on system configuration and may require enabled hardware, software or service activation. Learn more at <http://www.intel.com/> or from the OEM or retailer.

No computer system can be absolutely secure.

Intel, Atom, Celeron, and the Intel logo are trademarks of Intel Corporation in the U.S. and/or other countries.

*Other names and brands may be claimed as the property of others.

Copyright © 2015, Intel Corporation. All rights reserved.



Contents

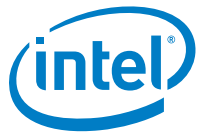
1.0	Introduction.....	5
1.1	Scope of document	5
1.2	Acronyms and Terminology.....	5
2.0	Best Known Configuration.....	6
3.0	Building Windows* 10 IoT Core Image.....	7
3.1	Installing Windows 10 Developer Tools.....	7
3.2	Updating IoT Core FFU with the packages available with Graphics driver release.....	8
3.3	Generating Graphics Driver Package from INF.....	8
3.4	Creating IoT Core FFU Image.....	11
3.5	Flashing Windows* 10 IoT Core FFU to Storage Device	12
3.5.1	Using SD or USB Storage Devices.....	12
3.5.2	Using SATA Disk.....	12
3.6	Booting up Windows* 10 IoT Core on the Bayley Bay CRB.....	13

Figures

Figure 1.	Flash the IoT Core Image onto the SD* Card	12
-----------	--	----

Tables

Table 1.	Acronyms and Terminology.....	5
Table 2.	Display of the Bayley Bay CRB IP Address.....	14



Revision History

Date	Revision	Description
December 2015	001	Intel® HD Graphics V36.19.0 Gold for Windows 10* IoT Core for Intel® Atom™ processor E3800 Family/Intel® Celeron® Processor N2807/N2900/J1900 Gold Release

§



1.0 Introduction

1.1 Scope of document

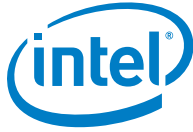
This User Guide describes how to install Microsoft* Windows* 10 developer tools to create an image for the Microsoft* Windows* 10 IoT Core 32-bit operating system, booting up the IoT Core image, upgrading I/O drivers (General Purpose Input/Output (GPIO), Inter-Integrated Circuit* (I²C*), Serial Peripheral Interface (SPI), and High Speed Universal Asynchronous Receiver/Transmitter (HSUART)) for the IoT Core image, and best known methods for platforms and software drivers.

This User Guide is intended for Original Design Manufacturer (OEMs) and Original Equipment Manufacturer (ODMs) that are enabling IoT Core drivers for the Windows* 10 operating system with the Intel® Atom™ E3800 processor, Intel® Celeron® Processor N2807/N2900/J1900.

1.2 Acronyms and Terminology

Table 1. Acronyms and Terminology

Term	Description
ADK	Assessment and Deployment Kit
CRB	Customer Reference Board
EEAP	Ecosystem Engineering Access Program
FFU	Full Flash Update
GPIO	General Purpose Input/Output
HSUART	High Speed Universal Asynchronous Receiver/Transmitter
I ² C*	Inter-Integrated Circuit*
ODM	Original Design Manufacturer
OEM	Original Equipment Manufacturer
RTM	Release to Manufacturing
SDK	Software Development Kit
UART	Universal Asynchronous Receiver/Transmitter
WDK	Windows Driver Kit



2.0 *Best Known Configuration*

The following is a list of software tools required to create and deploy Windows* 10 IoT Core operating system with custom drivers on the hardware:

- Windows 10 RTM (Build 10240 or better) x86 or x64 (Windows* 8.1 latest version also works but not recommended)
- Visual Studio* Professional 2015 : 14.0.23107.0 D14REL
- Visual Studio Tools for Universal Windows Apps : 14.0.23121.00 D14OOB
- Windows IoT Core Project Templates
- Windows Software Development Kit - SDK Windows 10.0.10586
- Windows Driver Kit - WDK Windows 10.0.10586
- Windows Assessment and Deployment Kit - ADK Windows 10.0.10586
- Download Visual Studio and Windows development tool kits from <https://msdn.microsoft.com/en-us/windows/hardware/dn913721.aspx>
- **Windows_10_IoT_Core_Packages.msi** from either through the Ecosystem Engineering Access Program (EEAP) or other channels



3.0 Building Windows* 10 IoT Core Image

This section describes how to install Windows* 10 developer tools to create an image for the Windows 10* IoT Core 32-bit operating system, upgrading graphics drivers for the IoT Core image and Creating Full Flash Update (FFU) image with new graphics driver.

3.1 Installing Windows 10 Developer Tools

The following are steps to install Windows 10 developer tools:

1. Install the Windows 10 RTM operating system on a development machine.
2. Install the Microsoft Visual Studio* Professional 2015 software on the development machine.

Note: When installing any editions of Microsoft Visual Studio 2015, do a **Custom** install and select the checkbox Universal Windows App Development Tools -> **Tools and Windows SDK**.

3. Install Windows 10 IoT Core Project Templates from <https://visualstudiogallery.msdn.microsoft.com/06507e74-41cf-47b2-b7fe-8a2624202d36>
4. Download the latest release (version 10.0.10586) of the IoT Core default OS image for the Windows 10 operating system (**IoTCore.FFU**) from <http://go.microsoft.com/fwlink/?LinkId=616848>.

This image file includes the installer package `Windows_10_IoT_Core_Mbm.msi`. When installation is complete, the IoT Core FFU image for Windows 10 (**flash.ffu**) can be found at `C:\Program Files (x86)\Microsoft IoT\FFU\MinnowBoardMax`.

The Flash Tool (`IoTCoreImageHelper.exe`), which can be used to flash the FFU image to an SD* card or a USB pen drive, is also installed.

5. To build a custom FFU image, install following package to extract the FFU content `MSPackages` to the development machine:

Windows_10_IoT_Core_Packages.msi

Download `Windows_10_IoT_Core_Packages.msi` from either through the EEAP or other channels.

6. Download and install the Windows 10 SDK from the following:
<https://msdn.microsoft.com/en-us/windows/hardware/dn913721.aspx>
The SDK includes the header file and library to compile the driver.
7. Download and install the Windows 10 WDK from the following:
<https://msdn.microsoft.com/en-us/windows/hardware/dn913721.aspx>



The WDK will be used to build driver packages (refer to Upgrading I/O driver for Windows* 10 IoT Core).

8. Download and install the Windows 10 ADK from the following:

<https://msdn.microsoft.com/en-us/windows/hardware/dn913721.aspx>

In addition to new and improved deployment tools used to automate a large-scale Windows 10 deployment, the ADK includes the Windows Assessment Toolkit and Windows Performance Toolkit that access the quality and performance of systems or components.

3.2 Updating IoT Core FFU with the Packages Available with Graphics Driver Release

This section shows the graphics driver integration prior to the FFU generation. This section assumes that the driver **.CAB** and **.SPKG** files are available. However, if no such files exist except for INF files, then refer to Section 3.3.

1. The graphics driver build package will be provided with the following:

- **INTEL.GFX.BuildXXXX.cab**
- **INTEL.GFX.BuildXXX.spkg**

Note: XXXX reflects the auto generated build number.

2. Copy **INTEL.GFX.BuildXXXX.cab** to C:\Program Files (x86)\Windows Kits\10\MSPackages\Retail\X86\.
3. Modify the **"MBMFM.xml"** file from "C:\Program Files (x86)\Windows Kits\10\FMFiles\X86".
 - a. Locate following line of **.xml script**:

```
<PackageFile
  Path="$ (mspackageroot) \Retail\$(cputype) \$(buildtype) "
  Name="INTEL.MBM.GRFX.cab" >
```
 - b. Replace **"Intel.MBM.GRFX.cab"** with **"INTEL.GFX.BuildXXXX.cab"**.
4. Skip Section [3.3, Generating Graphics Driver Package from INF](#) and proceed to Section [3.4, Creating IoT Core FFU Image](#).

3.3 Generating Graphics Driver Package from INF

This section describes the steps to build a custom **.cab** and **.spkg** file using **pkggen.exe** for graphics driver:

For more **pkggen.exe** command options, refer to [https://msdn.microsoft.com/en-us/library/dn756636\(v=vs.85\).aspx](https://msdn.microsoft.com/en-us/library/dn756636(v=vs.85).aspx).

This section also describes the steps to create a custom IoT Core FFU image.



As reference, refer to [https://msdn.microsoft.com/en-us/library/windows/hardware/dn756642\(v=vs.85\).aspx](https://msdn.microsoft.com/en-us/library/windows/hardware/dn756642(v=vs.85).aspx).

Follow the steps below to build custom **.cab** and **.spkg** files for the drivers and create an FFU image:

1. Open an elevated Deployment and Imaging Tools Environment command prompt.

Note: The Deployment and Imaging Tools Environment shortcut can be found in the Windows start menu "Windows Kits" folder.

2. Set the following environmental variables:

```
set KITSROOT=C:\Program Files (x86)\Windows Kits\10

set SIGN_OEM=1

set WPKROOT=%KITSROOT%

set WPKCONTENTROOT=C:\Program Files (x86)\Windows Kits\10

set path=%PATH%;%KITSROOT%\tools\bin\i386
```

3. To install the OEM certificates, go to the C:\Program Files (x86)\Windows Kits\10\Tools\bin\i386 directory in an elevated command console.

Note: The directory is applicable to the Windows 10 x86 development machine. Change the directory accordingly if running on a Windows 10 x64 machine.

4. Run the following command: **installoemcerts.cmd**

Note: Note: This command is only needed when building the package for the first time.

5. Create the **intel_gfx.pkg.xml** file for the graphics driver. The file names and contents are shown below:

Contents of the .pkg.xml Files:

```
<?xml version="1.0" encoding="utf-8"?>
<Package xmlns="urn:Microsoft.WindowsPhone/PackageSchema.v8.00"
  Owner="INTEL"
  OwnerType="OEM"
  ReleaseType="Production"
  Platform="Win32"
  Component="GFX"
  SubComponent="Build1227">
  <Components>
    <Driver InfSource=".\\igdlh.inf">
      <Reference Source=".\\igdkmd32.sys"/>
      <Files>
        <File Source=".\\igdkmd32.sys"/>
      </Files>
      <Reference Source=".\\igd10iumd32.dll"/>
      <Files>
        <File Source=".\\igd10iumd32.dll"
          DestinationDir="$(runtime.system32)"
          Name="igd10iumd32.dll"/>
      </Files>
    </Driver>
  </Components>
</Package>
```



```
</Files>
<Reference Source=".\\igdusc32.dll"/>
<Files>
  <File Source=".\\igdusc32.dll"
    DestinationDir="$(runtime.system32)" Name="igdusc32.dll"/>
</Files>
<Reference Source=".\\igfx11cmrt32.dll"/>
<Files>
  <File Source=".\\igfx11cmrt32.dll"
    DestinationDir="$(runtime.system32)"
Name="igfx11cmrt32.dll"/>
</Files>
<Reference Source=".\\igfxcmjit32.dll"/>
<Files>
  <File Source=".\\igfxcmjit32.dll"
    DestinationDir="$(runtime.system32)"
Name="igfxcmjit32.dll"/>
</Files>
</Driver>
</Components>
</Package>
```

6. Copy **intel_gfx.pkg.xml**, **graphics** drivers INF and SYS files to C:\Intel_gfx.

7. Run the following command:

```
cd c:\intel_gfx
```

8. Run the following command to generate the package for the graphics driver:

```
pkggen intel_gfx.pkg.xml
/config:"%KITSROOT%\Tools\bin\i386\pkggen.cfg.xml" /cpu:x86
/variables:"HIVE_ROOT=%WPDKCONTENTROOT%\CoreSystem\10.0.10586.
0\x86" /version:"10.0.1004.0"
```

9. After the package generation for Graphics driver is complete, the following files can be found at the C:\intel_gfx folder:

- **INTEL.GFX.Build1227.cab**
- **INTEL.GFX.Build1227.spkg**

10. Copy **INTEL.GFX.Build1227.cab** to C:\Program Files (x86)\Windows Kits\10\MSPackages\Retail\X86\fre.

11. Modify the "**MBMFM.xml**" file from "C:\Program Files (x86)\Windows Kits\10\FMFiles\x86" and Locate following line of .xml script:

```
<PackageFile
Path="$(mspackageroot)\Retail\$(cputype)\$(buildtype)"
Name="INTEL.MBM.GRFX.cab" >
```

12. Replace the "**Intel.MBM.GRFX.cab**" with "**INTEL.GFX.Build1227.cab**".



3.4 Creating IoT Core FFU Image

This section describes the steps to create a custom IoT Core FFU image.

As reference, refer to:

- [https://msdn.microsoft.com/en-us/library/windows/hardware/dn756642\(v=vs.85\).aspx](https://msdn.microsoft.com/en-us/library/windows/hardware/dn756642(v=vs.85).aspx)
- <https://ms-iot.github.io/content/en-US/win10/ImageCreation.htm>

Follow these steps to build custom **.cab** and **.spkg** files for the drivers and to create an FFU image:

1. Open an elevated Deployment and Imaging Tools Environment command prompt:

You may find the “Deployment and Imaging Tools Environment” shortcut from the Windows start menu **“Windows Kits”** folder. Type below commands

```
set KITSROOT=C:\Program Files (x86)\Windows Kits\10
set path=%PATH%;%KITSROOT%\tools\bin\i386
set AKROOT=%KITSROOT%
md C:\FFU
cd /d C:\FFU
imggen.cmd IoTCore.FFU
"%KITSROOT%\OEMInputSamples\MBM\ProductionOEMInput.xml"
"%KITSROOT%\MSPackages" x86
```

2. The build process may take 20 to 30 minutes and “IoTCore.FFU” will be available in the C:\FFU folder.
3. Copy the FFU flash utility **DISM** folder from the ADK installation to C:\FFU.

For Windows 10 x64 development machine – Copy it from “C:\Program Files (x86)\Windows Kits\10\Assessment and Deployment Kits\Deployment Tool\amd64”

For Windows 10 x86 development machine – Copy it from “C:\Program Files (x86)\Windows Kits\10\Assessment and Deployment Kits\Deployment Tool\x86”

3.5 Flashing Windows* 10 IoT Core FFU to Storage Device

This section describes how to flash the IoT core image to different storage device.

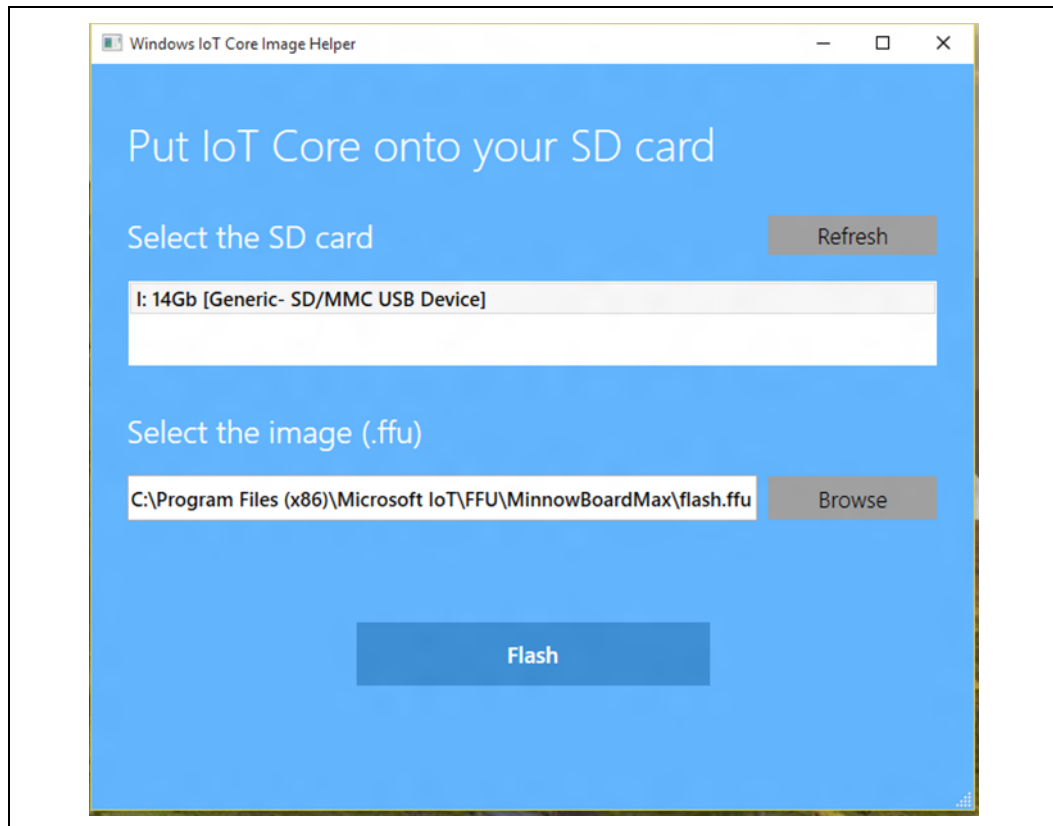
3.5.1 Using SD or USB Storage Devices

1. Insert a SD card or USB pen drive into the development machine.
2. Run the **IoTCoreImageHelper.exe** tool to flash the storage device.

This tool is located at C:\Program Files (x86)\Microsoft IoT

3. This tool will enumerate the device as shown in [Figure 1](#). Select the destination device, provide the FFU image location, and click **Flash** to flash the image.

Figure 1. Flash the IoT Core Image onto the SD* Card



4. Click on the **Safely Remove Hardware** icon in your task tray and select your device to safely remove it from the system. Failing to do this can cause corruption on the image.

3.5.2 Using SATA Disk

1. Connect the SATA disk to the development machine.
2. Open a Command Prompt and log in as the **Administrator**.



3. Run the “**diskpart**” command and then “**list disk**”, and find the number of this SATA disk.
4. Run “**exit**” command to quit the DiskPart terminal.
5. Run the following command to flash the FFU image to SATA disk:

```
C:\FFU\DISM\dism.exe /apply-image /ImageFile:C:\FFU\IoTCore.ffu  
/ApplyDrive:\\.\PhysicalDrive# /skipplatformcheck
```

Note: The PhysicalDrive# number is the number as the same disk# detected by the “diskpart” utility.

3.6 Booting up Windows* 10 IoT Core on the Bayley Bay CRB

This section describes how to boot the IoT Core image on a Bayley Bay Customer Reference Board (CRB).

1. Insert the SD card, or USB pen drive, into the Bayley Bay CRB.
2. When it boots, press **F2** to get to the BIOS settings.
3. Navigate to Device Manager -> **System Setup**.
 - a. Boot -> **OS Selection**: Select “**Windows 8.X**”.
 - b. South Cluster Configuration -> LPSS & SCC Configuration -> **LPSS & SCC Devices Mode**: Select “**ACPI Mode**”.
 - c. South Cluster Configuration -> LPSS & SCC Configuration -> **ACPI Reporting MMC/SD Media As**: Select “**Non-Removable**”.
4. Press **F4** to save these changes, or select “**Commit Changes and Exit**”.
5. Navigate back to the first level and select Boot Maintenance Manager -> Boot Options -> **Add Boot Option**.
6. In File Explorer, select “EFI/ESP,” item and press Enter, then navigate to “bootia32.efi” located at EFI/Boot/**bootia32.efi**. Press **Enter** to input the description, type any name, e.g. “IoTCoreImage”.
7. Select “**Commit Changes and Exit**”.
8. Navigate to “**Change Boot Order**”. Highlight the boot order list and press **Enter**.
9. Highlight “**IoTCoreImage**” and press “**+**” to move it to the top of the list.
10. Select “**Commit Changes and Exit**”.
11. The Bayley Bay CRB should automatically boot to “**IoTCoreImage**”.

Note: After selecting the Language the device will boot up. If the Language is not selected, the device may restart. After the device has booted, the default application from the USB thumb drive will launch and display the IP address of the Bayley Bay CRB.

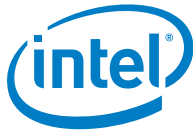
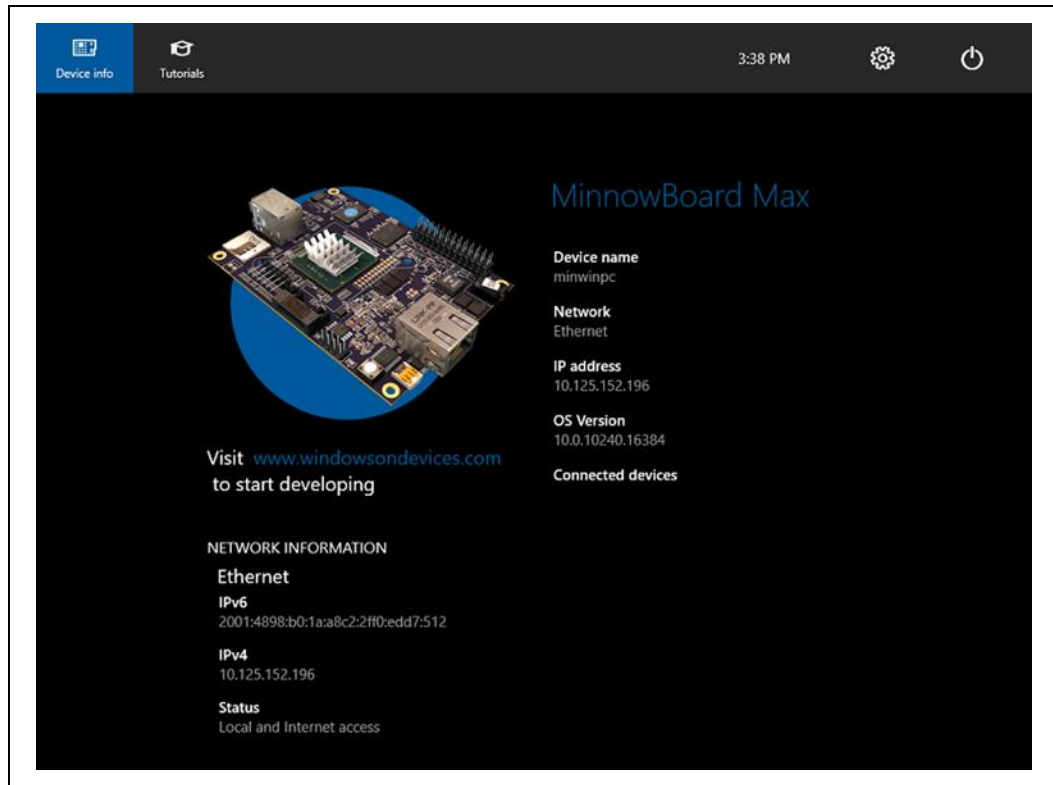


Table 2. Display of the Bayley Bay CRB IP Address



§