

# UEFI & EDK II TRAINING

Porting to a New Board

[tianocore.org](https://tianocore.org)



# LESSON OBJECTIVE

- ★ Use Existing Platform based on Apollo Lake
- ★ Clone & Define a the New Board
- ★ Update Hardware Related Changes
- ★ Other Platform Customizations



# OPEN SOURCE PORTING GUIDE

Porting Guide for a new board with the Intel Atom® Processor E3900 Series Platforms (formerly Apollo Lake) Platform

Download [PDF](#)

## WHITE PAPER

Intel® Platform Firmware



## Open Source UEFI Firmware Porting Guide: Intel Atom® Processor E3900 Series Platforms

September 20, 2018

# COPY EXISTING BOARD

Clone existing Board from a current project



# Defining a new Board

Process to add a new platform (“board”) to the existing firmware project, based on a reference design.

The first step is to simply “clone” an existing board and make the appropriate changes

Download Project Source Code – use the Leaf Hill CRB as the starting project

To download, see latest build instructions: at

<https://firmware.intel.com/projects/IntelAtomProcessorE3900>



# Add a New Board to the Project

- Support for Multiple Boards already available
- Many platforms with minor variations can share common code
- Board definitions are under the “Board” directory:


`edk2-platforms\Platform\BroxtonPlatformPkg\Board`








Developers can clone the reference project by copying the directory, renaming the directory, and adding references in several configuration files.



# Copy Existing Board Dir as a Reference

```
bash$ cd edk2-platforms/Platform/BroxtonPlatformPkg/Board/  
bash$ cp -R LeafHill NewBoard
```



Name	Size	Modified	Modified — Time
 AuroraGlacier	4 items	7 Sep	7 Sep 03:23
 BensonGlacier	4 items	7 Sep	7 Sep 03:23
 LeafHill	5 items	7 Sep	7 Sep 03:23
 MinnowBoard3	5 items	7 Sep	7 Sep 03:23
 MinnowBoard3Module	6 items	7 Sep	7 Sep 03:23
 NewBoard	5 items	7 Sep	7 Sep 03:23
 UP2	5 items	7 Sep	7 Sep 03:23



# Copy Existing Board Dir as a Reference

## Changes to Functions, Variables, and File GUIDs

Functions and global variables of libraries under the NewBoard folder need to be changed so they do not conflict with libraries of existing boards. These are referenced in .INF files copied from the reference project:

BoardInitPreMem.inf  
BoardInitPostMem.inf  
BoardInitDxe.inf

UEFI and EDK II associate a globally unique identifier (GUID) with various functions, files, and protocols.

New file GUIDs must also be generated for .INF files under the NewBoard folder to avoid conflict with GUIDs from existing projects.

Use <https://www.guidgenerator.com/>



# Copy Existing Board Dir as a Reference

**Add Components to DSC File –**

**Add DXE Library**

**Add PEI Libraries**

**Paths to Binary Stitching Files - Modify the post-build stitch file**

**Windows – IFWISitch\_Simple.bat**

**Linux – edk2-platforms\BuildBIOS.sh**

**Build and Test with the NEW Board**



# HARDWARE RELATED CHANGES



# Hardware-Related Changes

The next step is editing the cloned project based on the board configuration.

Update required UEFI IA Firmware changes for custom platforms that vary from a reference hardware design.

1. Detection of Board ID & Fab ID
2. Change UART serial port for UEFI IA Firmware debug messages
3. Change system memory parameters
4. Change display devices, peripherals
5. Modify I/O & GPIO configuration
6. Microcode updates

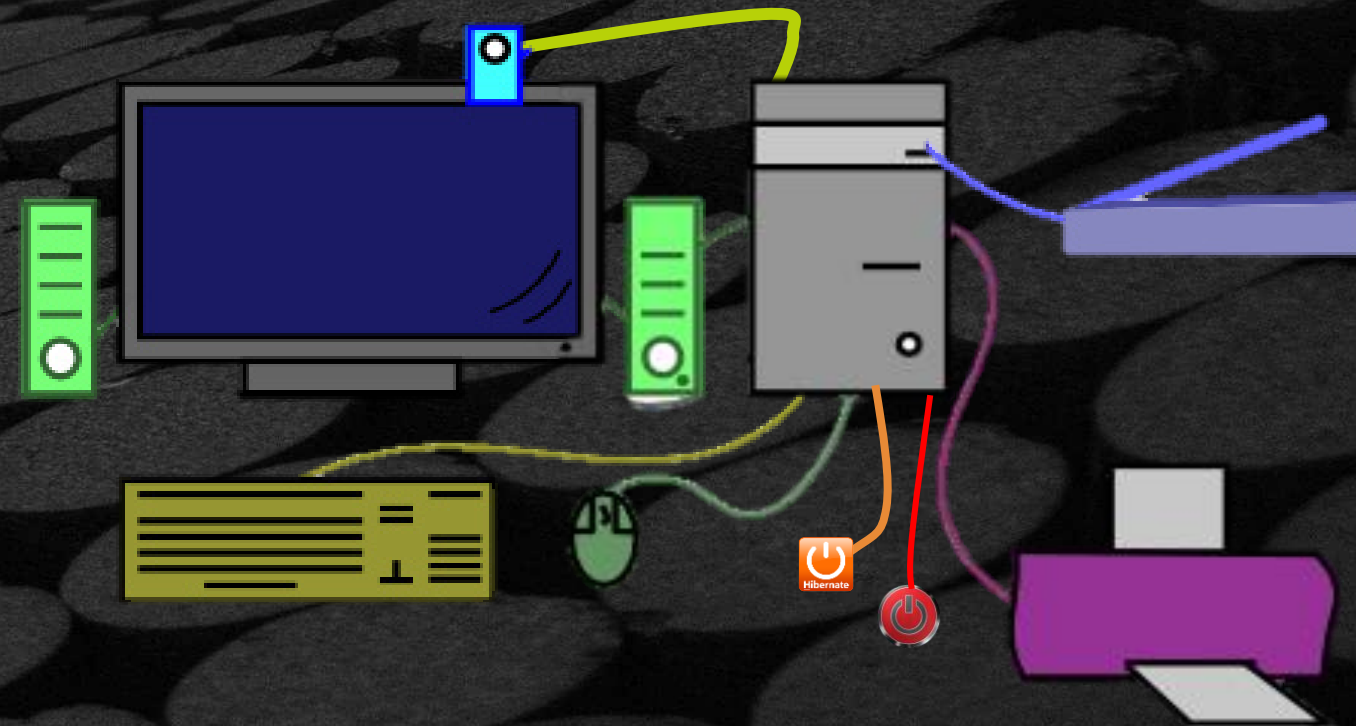


# Hardware-Related Changes- ACPI

## Add & Remove Peripherals in ACPI SSDT

Add New Peripherals – New Board .ASL file

Remove Peripherals – Update Global NVS

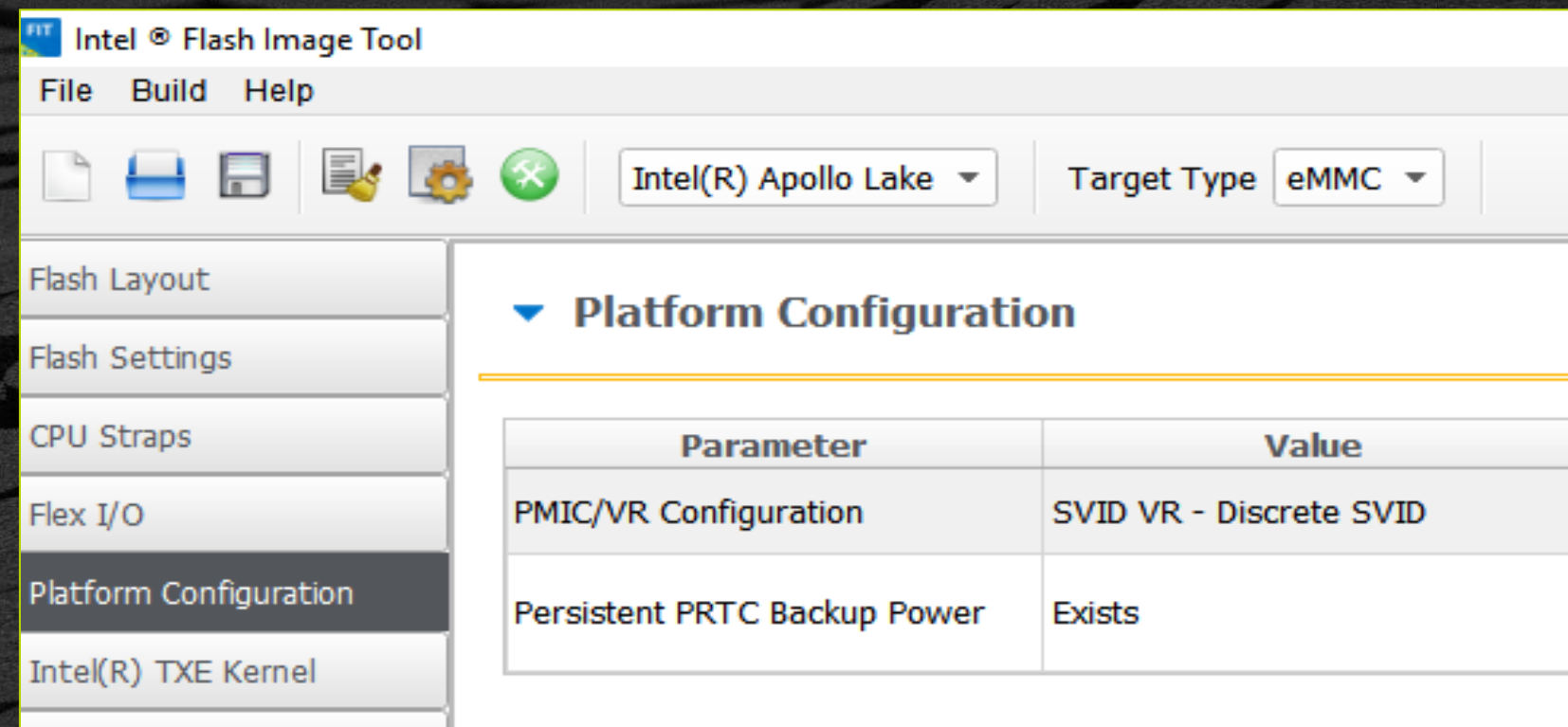




# Hardware-Related Changes – Flash Image Tool

## Configuration via Intel® Flash Image Tool-

1. Change PMIC/VR config
2. Modify I/O Config (PCIe & USB)
3. Microcode Updates



Selecting a pre-defined PMIC/VR configuration with the Intel® FIT tool



# OTHER PLATFORM CONFIGURATION



# Other Platform Configuration

- Intel® FSP Configuration –
  - Using Intel BCT to Configure Intel FSP Parameters
  - Using UPD to Override Intel FSP Parameters
- Change Default Value for User Setup Options
- Custom Boot Logo
- Default Boot Order
- Remove UEFI Shell
- SMBIOS Table
- Signed Capsule Update
- Enabling Verified Boot
- Trusted Platform Module (TPM)
- UEFI Secure Boot
- UEFI Networking





# Example: Custom Boot Logo

1. Create a new Directory for the Logo BMP

. . . BroxtonPlatformPkg/Board/NewBoard/Logo

2. Create a new GUID For the new Logo

```
gPeiNewBoardLogoGuid = { 0x02c2a0ef, 0x98ba, 0x417a, . . .
```

3. Add the reference to the bin logo to the board .fdf file

```
FILE FREEFORM = PCD(gPlatformModuleTokenSpaceGuid.PcdLogoFileGuid) {  
    SECTION RAW = $(PLATFORM_NAME)/Board/NewBoard/Logo/Logo.bmp }
```

4. Add the reference to the source code: NewBoard\BoardInitPostMem\BoardInit.c

```
// Board specific Logo
```

```
//
```

```
BufferSize = sizeof (EFI_GUID);
```

```
PcdSetPtr(PcdOemLogoFileGuid,, &BufferSize, (UINT8 *)& gPeiNewBoardLogoGuid);
```



# SUMMARY

- ★ Use Existing Platform based on Apollo Lake
- ★ Clone & Define a the New Board
- ★ Update Hardware Related Changes
- ★ Other Platform Customizations



# Questions?





