

UEFI & EDK II TRAINING

Porting an Existing Project with EDK II

tianocore.org



LESSON OBJECTIVE

- ★ Define the porting task list for porting existing platforms in EDK II in order to boot to the UEFI Shell
- ★ Explain the EDK II infrastructure, porting libraries, library classes, PCDs, and directory structures
- ★ Determine the necessary porting for each phase of a new EDK II platform Project

EDK II INFRASTRUCTURE

Review of EDK II

Directory Structure

Platform/Projects in Packages

Library Class Name → Instance

Platform Configuration Database PCD

EDK II INFRASTRUCTURE SUMMARY

Packages
List of related
modules

Libraries
Same name &
interface

PCDs
Platform
Config. DB

New Package Directory

MyWorkSpace/

· · · ·

NewProjectPkg/
Include/

Library/

· · ·

PlatformDrivers/

NewProjectPkg.DSC

NewProjectPkg.FDF

NewProjectPkg.DEC

DSC

Only one to build against

Contains PCD platform values

Defines library classes

Includes other modules

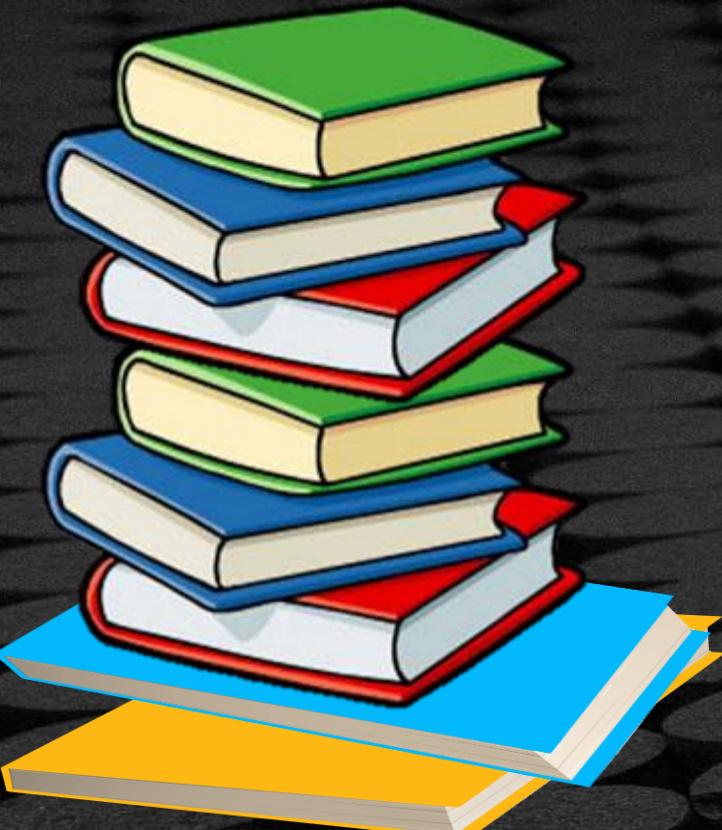
FDF

File to define flash layout

DEC

File to define PCDs within the
platform package

Libraries



DSC maps library class to library-instances

Syntax in DSC file

[libraryclasses]

LibraryClassName | Path/To/LibInstanceNameInstance1.inf

Search INF files for string: “LIBRARY_CLASS =”

Library Classes Section in DSC

DebugLib class in NewProjectPkg.dsc

Library Class Section

```
[LibraryClasses]
  DebugLib|MdePkg/Library/BaseDebugLibNull/BaseDebugLibNull.inf
  ...
  ...
```

```
[LibraryClasses.common.DXE_CORE]
  ...
  DebugLib|IntelFrameworkModulePkg/Library/PeiDxeDebugLibReportStatusCode/
    PeiDxeDebugLibReportStatusCode.inf
  ...
  ...
```

```
[LibraryClasses.common.DXE_SMM_DRIVER]
  DebugLib|MdePkg/Library/BaseDebugLibNull/BaseDebugLibNull.inf
```

Components Section

```
[Components]
  ...
  MyPath/MyModule.inf {
    <LibraryClasses>
      DebugLib|MdePkg/Library/BaseDebugLibSerialPort.inf
  }
```

FixedAtBuild**Dynamic****PatchableInModule****DyanmicEx****DynamicHii****FeatureFlag****DynamicVpd**

Syntax Examples

```
[pcdsFeatureFlag.common] [pcdsFixedAtBuild.IA32]  
[PcdsFixedAtBuild, PcdsPatchableInModule, PcdsDynamic,  
PcdsDynamicEx]
```

DEC

PCD defined in the DEC file from any package

[Guids.common]

PcdTokenSpaceGuidName={ 0x914AEBE7, 0x4635, 0x459b, { 0xAA, . . . } }

. . .

[Pcds...]

PcdTokenSpaceGuidName.PcdTokenName | Value [| DatumType [| MaxSize]] | Token

INF

Example used in **NewProjectPkg.dsc**

[PcdsPatchableInModule.common]

gEfiMdePkgTokenSpaceGuid.PcdPciExpressBaseAddress | 0x0E0000000

. . .

gNewProjectTokenSpaceGuid.PcdTCSmbaIoBaseAddress | 0x1040

gEfiCpuTokenSpaceGuid.PcdTemporaryRamBase | 0xFFE80000

gEfiCpuTokenSpaceGuid.PcdTemporaryRamSize | 0x00010000

DSC

PORTING TASK LIST

What is the best approach for how to port?

APPROACH - PORTING EDK II

Search Work Space

Find Similar Projects

Boot to UEFI Shell

Porting Task List



- 1 Create a New Project package directory
- 2 Create Build Files (DSC, DEC, and FDF)
- 3 Update target.txt to make your Project the default
- 4 Use UEFI PI phases to Port all required project modules
- 5 Update DSC w/ libraries, ported modules, and PCD values
- 6 Minimums for UEFI Shell

Porting Task List



- 1 Create a New Project package directory
- 2 Create Build Files (DSC, DEC, and FDF)
- 3 Update target.txt to make your Project the default
- 4 Use UEFI PI phases to Port all required project modules
- 5 Update DSC w/ libraries, ported modules, and PCD values
- 6 Minimums for UEFI Shell

New Directory Structure

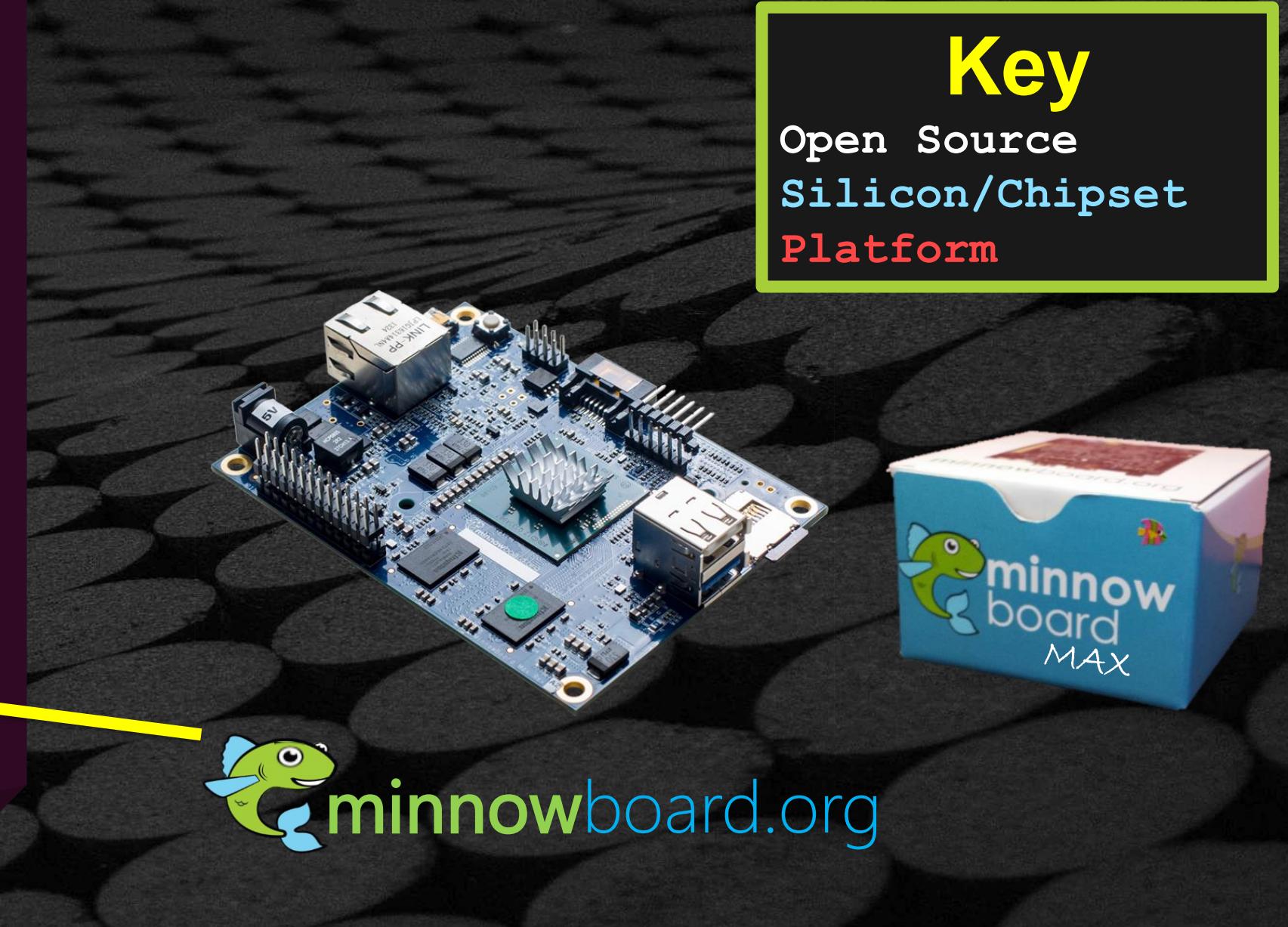
Open Source Directories

```
MyWorkSpace/  
Build/  
edk2/  
  - "edk2 Open Source"  
  - . . .  
edk2-platforms/  
  V1v2DeviceRefCodePkg/  
    AcpiTablesPCAT/  
    Include/  
    ValleyView2Soc/  
      CPU/  
      NorthCluster/  
      SouthCluster  
    V1v2TbtDevicePkg/  
Silicon/  
  IA32FamilyCpuPkg/  
  V1v2BinaryPkg  
  V1v2MiscBinariesPkg/
```

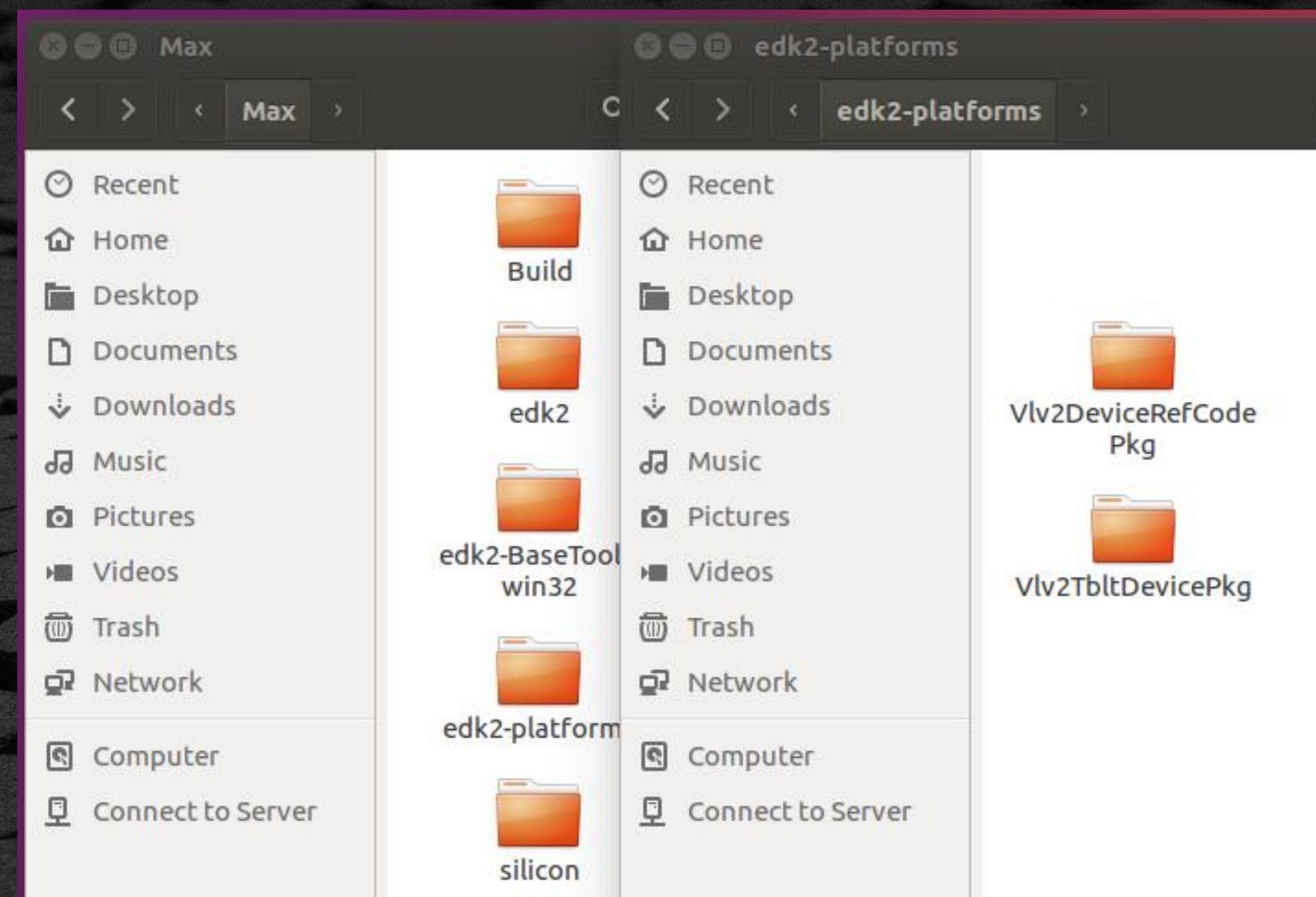
Platform and Silicon Directories

Key

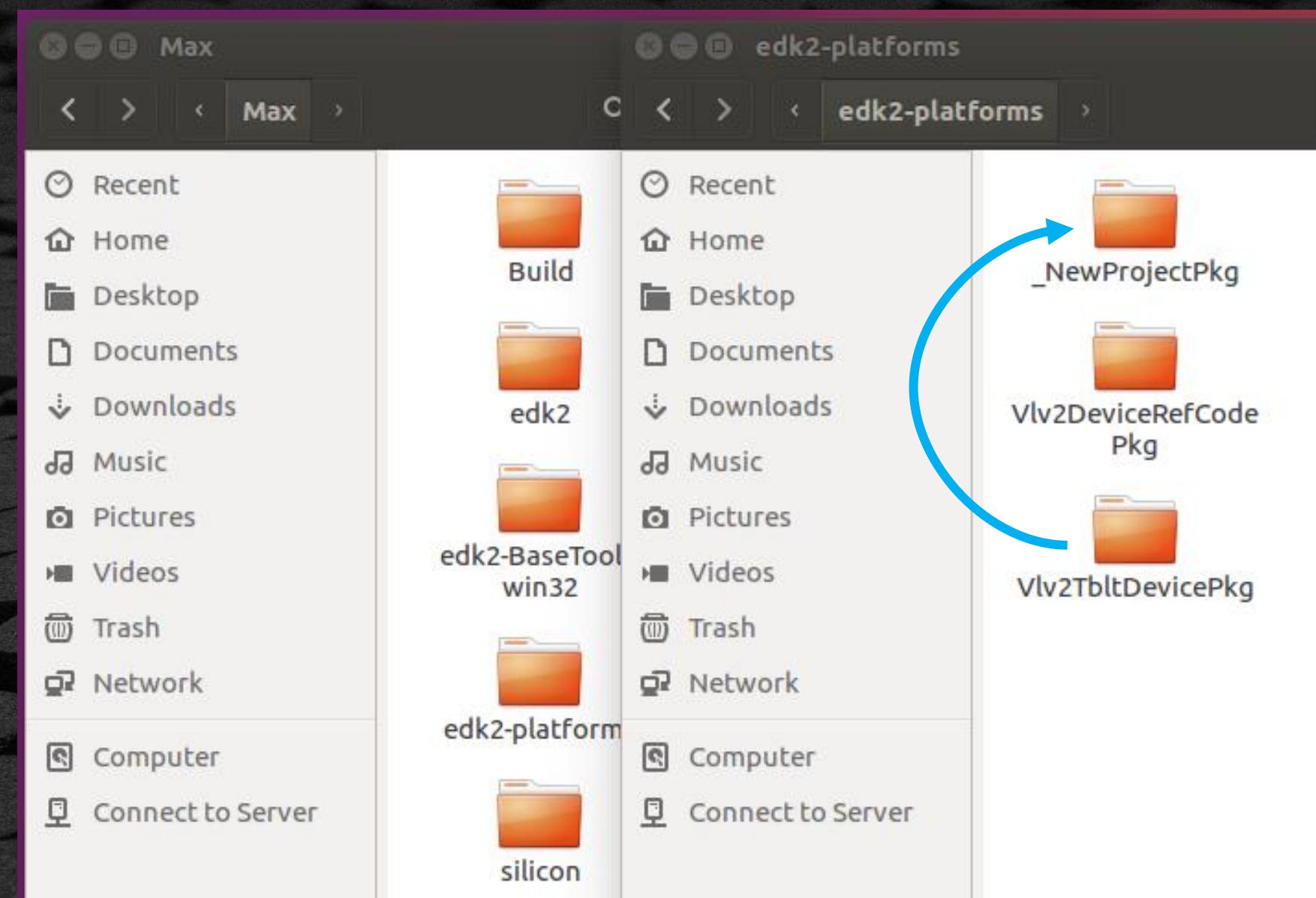
Open Source
Silicon/Chipset
Platform



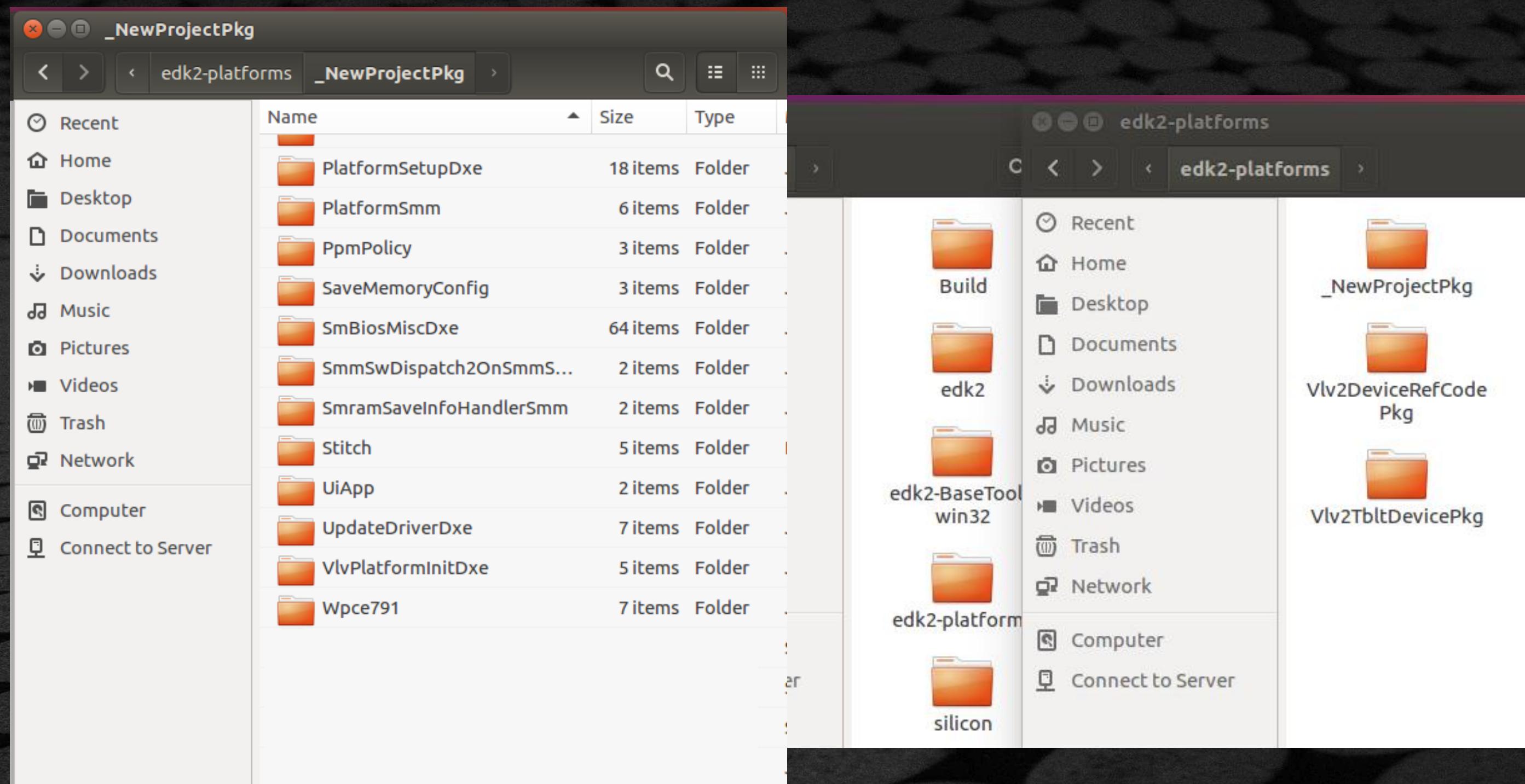
New Directory Structure



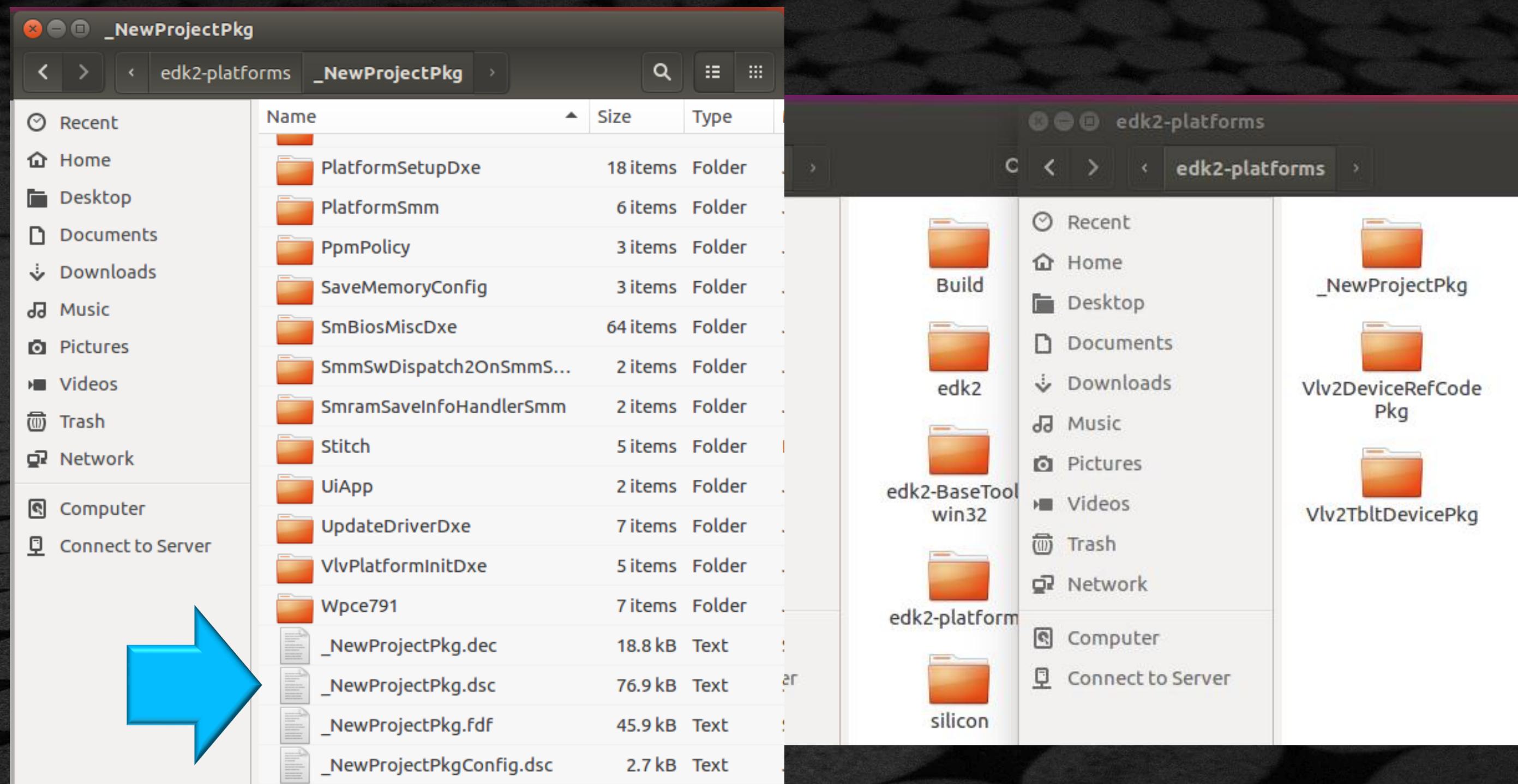
New Directory Structure



New Directory Structure



New Directory Structure



PCD Example with New Project

Open Source Directories

```
MyWorkSpace/  
Build/  
edk2/  
  - "edk2 Open Source"  
  - . . .  
edk2-platforms/  
  Vlv2DeviceRefCodePkg/  
    AcpiTablesPCAT/  
    Include/  
    ValleyView2Soc/  
      CPU/  
      NorthCluster/  
      SouthCluster  
    NewProjectPkg/  
Silicon/  
  IA32FamilyCpuPkg/  
  Vlv2BinaryPkg  
  Vlv2MiscBinariesPkg/
```

Platform and Silicon Directories

Key

Open Source
Silicon/Chipset
Platform

PCD Example with New Project

Open Source Directories

```
MyWorkSpace/  
Build/  
edk2/  
  - "edk2 Open Source"  
  - . . .  
edk2-platforms/  
  V1v2DeviceRefCodePkg/  
  AcpiTablesPCAT/  
  Include/  
    ValleyView2Soc/  
      CPU/  
      NorthCluster/  
      SouthCluster  
    NewProjectPkg/  
  Silicon/  
    IA32FamilyCpuPkg/  
    V1v2BinaryPkg  
    V1v2MiscBinariesPkg/
```

Platform and Silicon Directories

Key

Open Source
Silicon/Chipset
Platform

PCDs are defined in DEC

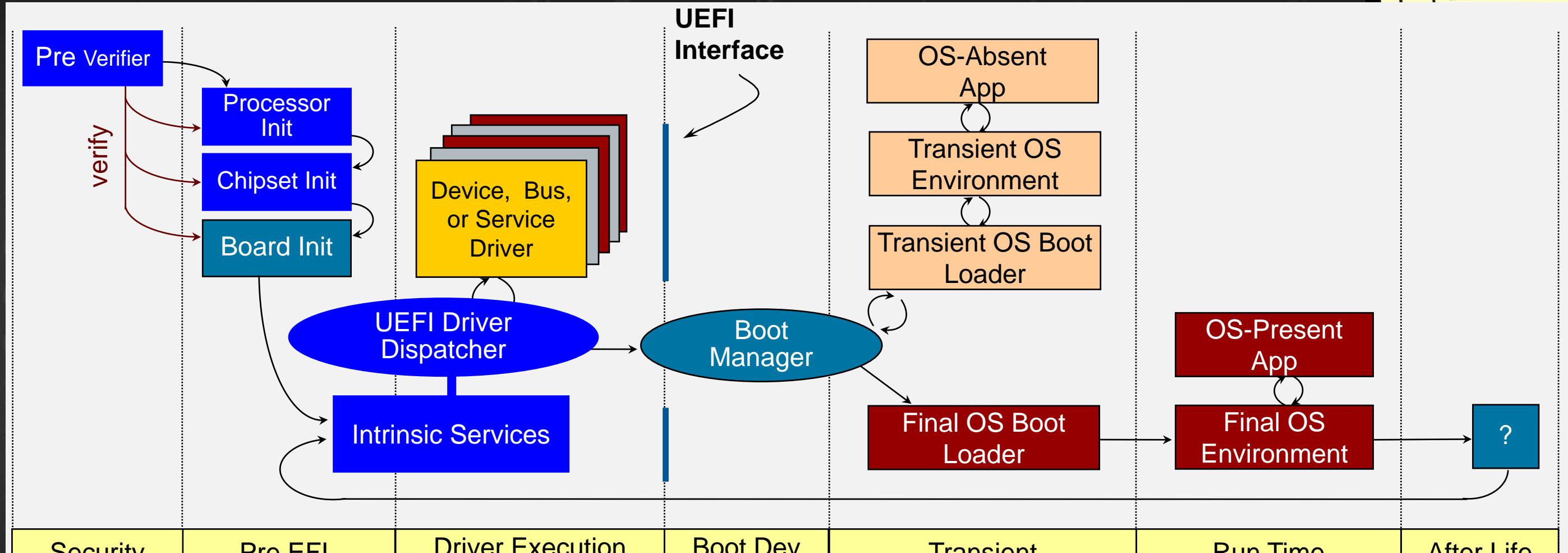
Broxton Silicon Ref package
BroxtonSoc/BroxtonSiPkg/

Porting Task List



- 1 Create a New Project package directory
- 2 Create Build Files (DSC, DEC, and FDF)
- 3 Update target.txt to make your Project the default
- 4 **Use UEFI PI phases to Port all required project modules**
- 5 Update DSC w/ libraries, ported modules, and PCD values
- 6 Minimums for UEFI Shell

Porting Task List



Port all required modules for your project using
UEFI Platform Initialization phases as a guide

Platform Initialization: SEC Phase



x86 (IA32 & x64)

MyWorkDir
/IA32FamilyCpuPkg
/SecCore/Ia32
/ResetVec.asm16

Itanium

MyWorkDir
/ItaniumFamilyCpuPkg
/SecCore/Ipf
/SecCore.s/
SALE_ENTRY

OtherArch

MyWorkDir
/XCpuPkg
/SecCore
/XCpuPkg

SEC Code Start

Location for MinnowBoard MAX: IA32FamilyCpuPkg/SecCore/Ia32/ResetVec.nasmb

-Source Code Example

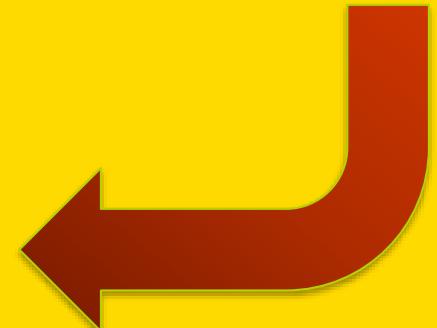
Entry point –

Binary of the compiled Firmware image at 4GB

```
; For IA32, the reset vector must be at 0xFFFFFFFF0, i.e., 4G-16 byte
; Execution starts here upon power-on/platform-reset.
;
ResetHandler:
    nop
    nop
ApStartup:
    ;
    ; Jmp Rel16 instruction to
    ; Typically, SEC entry point is the function _ModuleEntryPoint()
    ; defined in SecEntry.asm
    ;
    DB      0e9h          ; JMP Rel16 Instruction
```

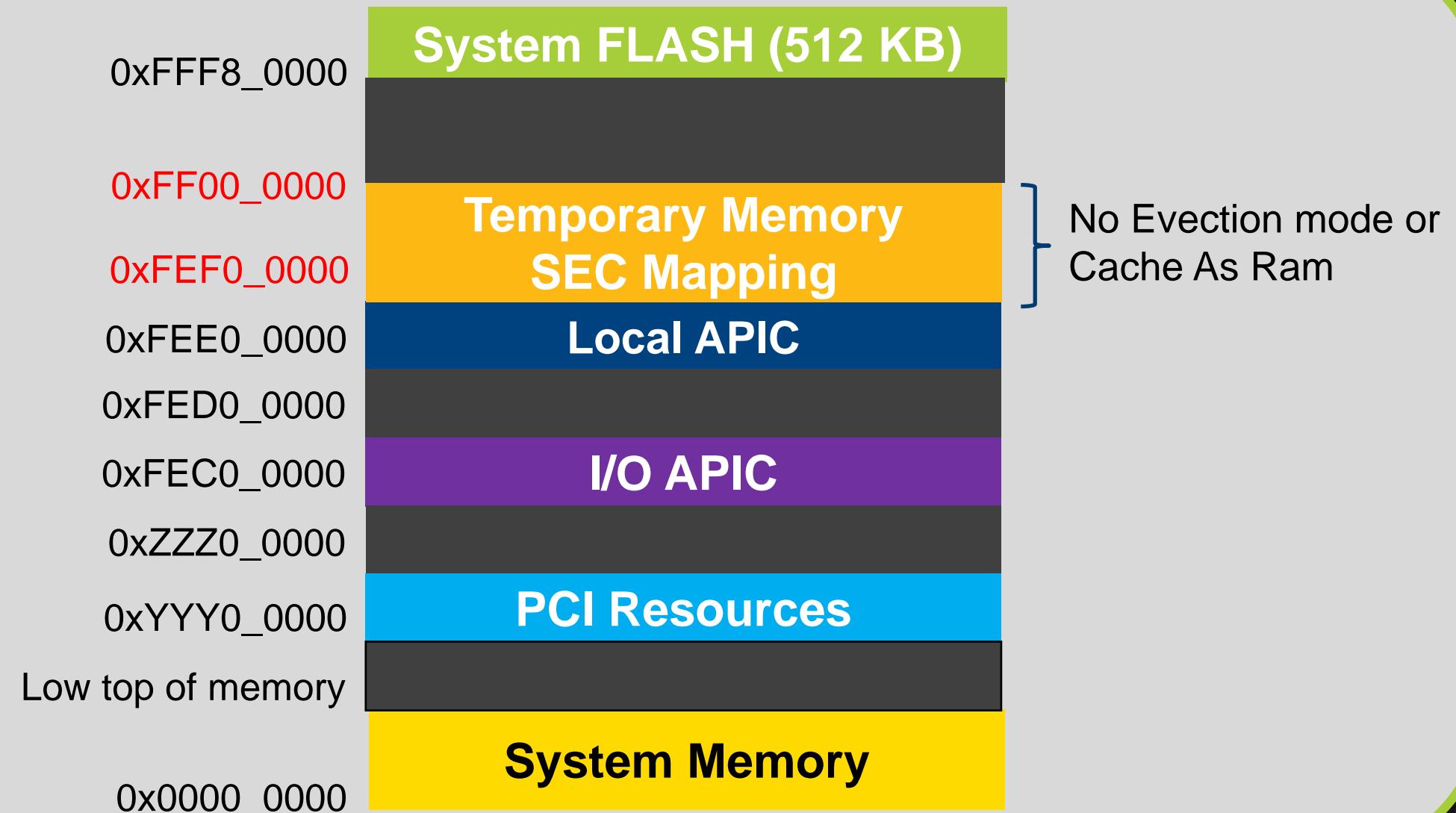
Platform SEC Lib for MinnowBoard MAX

```
NewProjectPkg/  
Library/  
  PlatformSecLib/  
    PlatformSecLib.c  
    PlatformSecLib.h  
    PlatformSecLib.inf  
IA32/  
  Chipset.inc  
  Flat32.asm  
  Platform.inc  
  SecCore.inc  
  Ia32.inc
```



ModuleEntryPoint
OR Flat32Start

Temporary Memory Example



SEC Lib, PCD Example

Defined in Package DEC **NewProjecPkg.dec**



[PcdsFixedAtBuild]
gPlatformModuleTokenSpaceGuid.PcdFlashMicroCodeAddress | 0xFFFF0000 | UINT32 | 0x20000015



NewProjectPkg/Library/PlatformSecLib/PlatformSecLib.inf

[Pcd]

gPlatformModuleTokenSpaceGuid.PcdFlashMicroCodeAddress

Value to use in Project **NewProjectPkg.dsc**

[PcdsFixedAtBuild]

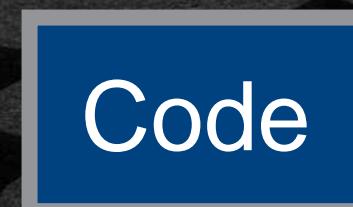
gPlatformModuleTokenSpaceGuid.PcdFlashMicroCodeAddress | 0xFFFF90000

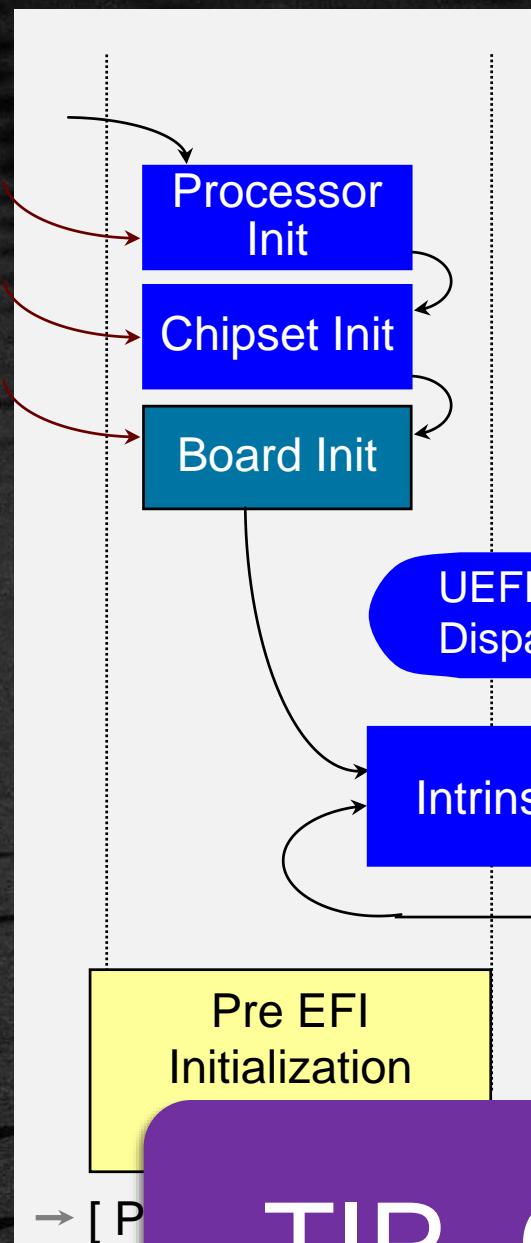
SEC - Referenced in the SEC library code

NewProjectPkg/Library/PlatformSecLib/Ia32/flat.asm

```
; Get the Flash Microcode address
```

```
mov     esi, PcdGet32 (PcdFlashMicroCodeAddress)
```





Main platform PEI code is in `PlatformPei` and `PlatformInitPei`

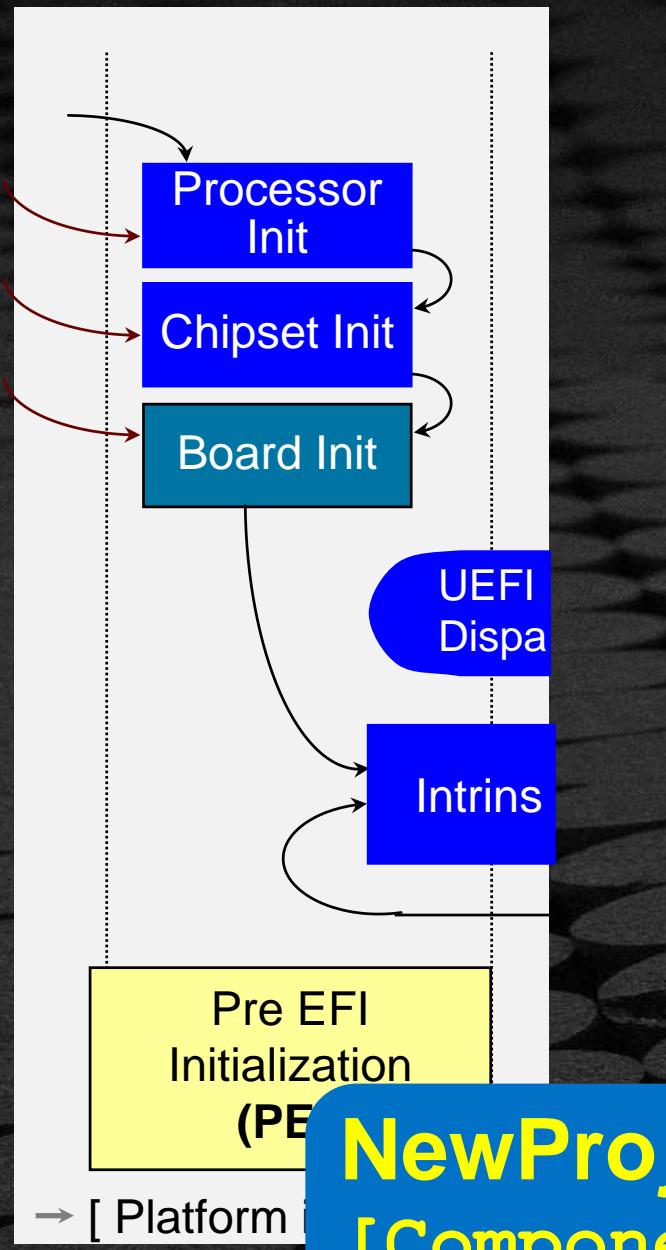
PEI code is dependent on the PEI drivers

Platform PEI code can be divided into 3 phases

- 1 Pre-Memory Initialization
- 2 Memory Initialization (MRC)

TIP- Check the DSC file for the Modules used

PEI Phase – MAX has 2 PEIM Modules



NewProjectPkg/

• • •

PlatformInitPei/

PlatformEarlyInit.inf

BootMode.c

CpuInitPeim.c

PchInitPeim.c

MchInit.c

MemoryCallback.c

MemoryPeim.c

• • •

PlatformEarlyInit.c

PlatformEarlyInit.h

PlatformInfoInit.c

LegacySpeaker.c

LegacySpeaker.h

Stall.c

PowerFailureHandle.c

PlatformSsaInitPeim.c

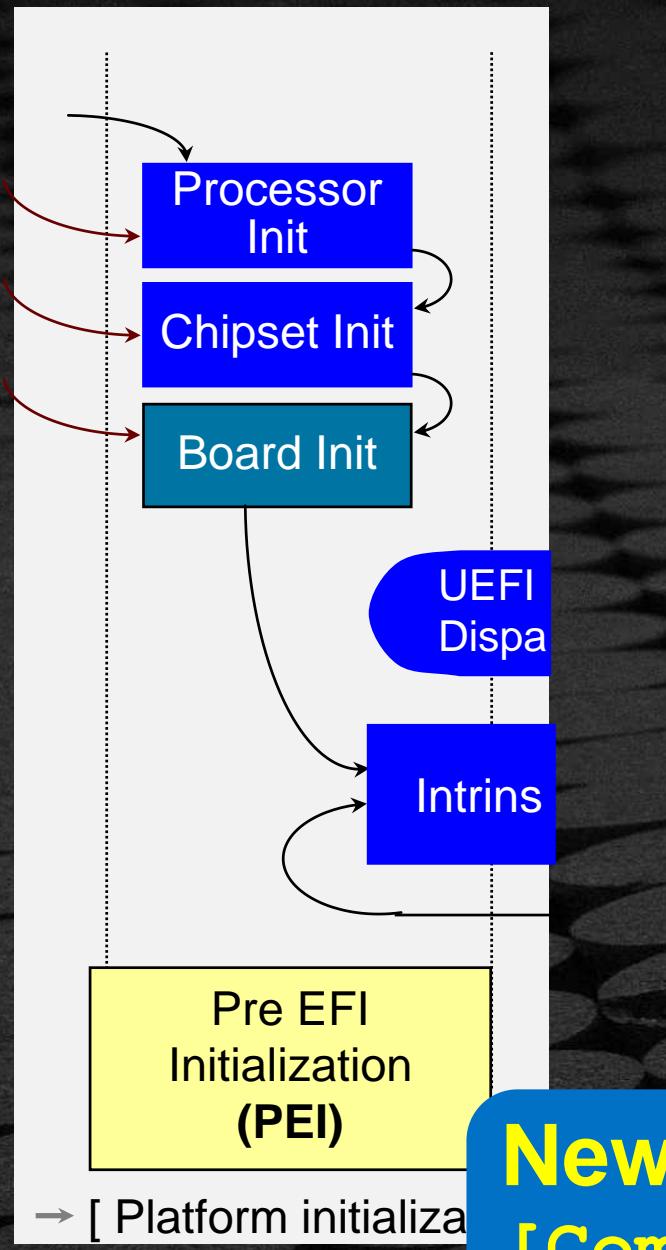
IsctWakeReason.c

NewProject.DSC file:

[Components.IA32]

NewProject/PlatformInitPei/PlatformEarlyInit.inf

PEI Phase – MAX has 2 PEIM Modules



NewProjectPkg/

• • •

PlatformPei/

PlatformPeiBB.inf

Platform.c

Platform.h

MemoryCallback.c

CommonHeader.h

Stall.c

BootMode.c

NewProject.DSC file:

[Components.IA32]

NewProject/PlatformPei/PlatformBB.inf

Recommended PEI Addresses

Base addresses

SMBUS addresses

GPIO addresses

Memory configuration (channels,
ports, etc...)

ACPI settings
(base address register, etc...)

How to search for Addresses in the Workspace



MyWorkspace

/edk2

/edk2-Platforms

/Silicon

Base Addr
Base
addr

Ports
Port
Ports

GPIOS
GPIO
GPIOS

How to search for GPIOs for NewProjectPkg

Make a Search in the *reference* and *platform* code .h files for “**GPIO**”

- NOTE: There will be many instances

How to search for GPIOs for NewProjectPkg

Make a Search in the *reference* and *platform* code .h files for “**GPIO**”

Within the results narrow the search by finding example structure tables.

- NOTE: There will be many instances
 - NewProjectPkg/Library/MultiPlatformLib/BoardGpios/
BoardGpios.h

Make a Search in the *reference*
and *platform* code .h files for
“**GPIO**”

Within the results narrow the search by finding example structure tables.

- **NOTE:** There will be many instances

- NewProjectPkg/Library/MultiPlatformLib/BoardGpios/
BoardGpios.h

How to search for GPIOs for NewProjectPkg

Make a Search in the *reference* and *platform* code .h files for “**GPIO**”

Within the results narrow the search by finding example structure tables.

Find which .c files use the structure **GPIO_CONF_PAD_INIT**

- NOTE: There will be many instances
 - NewProjectPkg/Library/MultiPlatformLib/BoardGpios/**BoardGpios.h**
 - NewProjectPkg/Library/MultiPlatformLib/BoardGpios/**BoardGpios.c**
 - Several calls to **InternalGpioConfig()** function called by **MultiPlatformInfoInit()**

How to search for GPIOs for NewProjectPkg

Make a Search in the *reference* and *platform* code .h files for “**GPIO**”

Within the results narrow the search by finding example structure tables.

Find which .c files use the structure **GPIO_CONF_PAD_INIT**

Check **NewProjectPkg.dsc** for the **.inf** file

- NOTE: There will be many instances
 - NewProjectPkg/Library/MultiPlatformLib/BoardGpios/**BoardGpios.h**
 - NewProjectPkg/Library/MultiPlatformLib/BoardGpios/**BoardGpios.c**
 - Several calls to **InternalGpioConfig()** function called by **MultiPlatformInfoInit()**
- NewProjectPkg/PlatformInitPei/**PlatformEarlyInit.inf**

PEI Memory Base Address, PCD Example

DEC

Defined in Package DEC

```
[PcdsFixedAtBuild]  
gEfiIchTokenSpaceGuid.PcdPeiIchEhciControllerMemoryBaseAddress | 0 | UINT32 | 0x39
```

INF

Module INF lists which PCDs get access

PlatformInitPei/PlatformEarlyInit.inf

```
[Pcd]  
gEfiIchTokenSpaceGuid.PcdPeiIchEhciControllerMemoryBaseAddress
```

DSC

Value to use in **NewProjectPkg.dsc**

```
[PcdsFixedAtBuild]  
gEfiIchTokenSpaceGuid.PcdPeiIchEhciControllerMemoryBaseAddress | 0xFC000000
```

PEI - Referenced in the PEI code

/PlatformInitPei/PchInitPeim.c **InstallPeiPchUsbPolicy()**

Code

```
PeiPchUsbPolicyPpi->EhciMemBaseAddr =  
PcdGet32(PcdPeiIchEhciControllerMemoryBaseAddress);
```

Critical PEIMs

PEIM NAME	CATEGORY	FUNCTION
CPU	Platform	Init and CPU I/O
DXE IPL	Generic	Starts DXE
PCI Configuration	PCAT	I/O 0xCF8 0xCFC
PCD	Generic	PCD services
Status Code	Generic	On StatusCodeLib
SMBus	Platform (<I/O Cntrl>)	Trending green
Memory Controller	Platform (<Memory Cntrl>)	
Motherboard	Platform	(including Stall PPI)

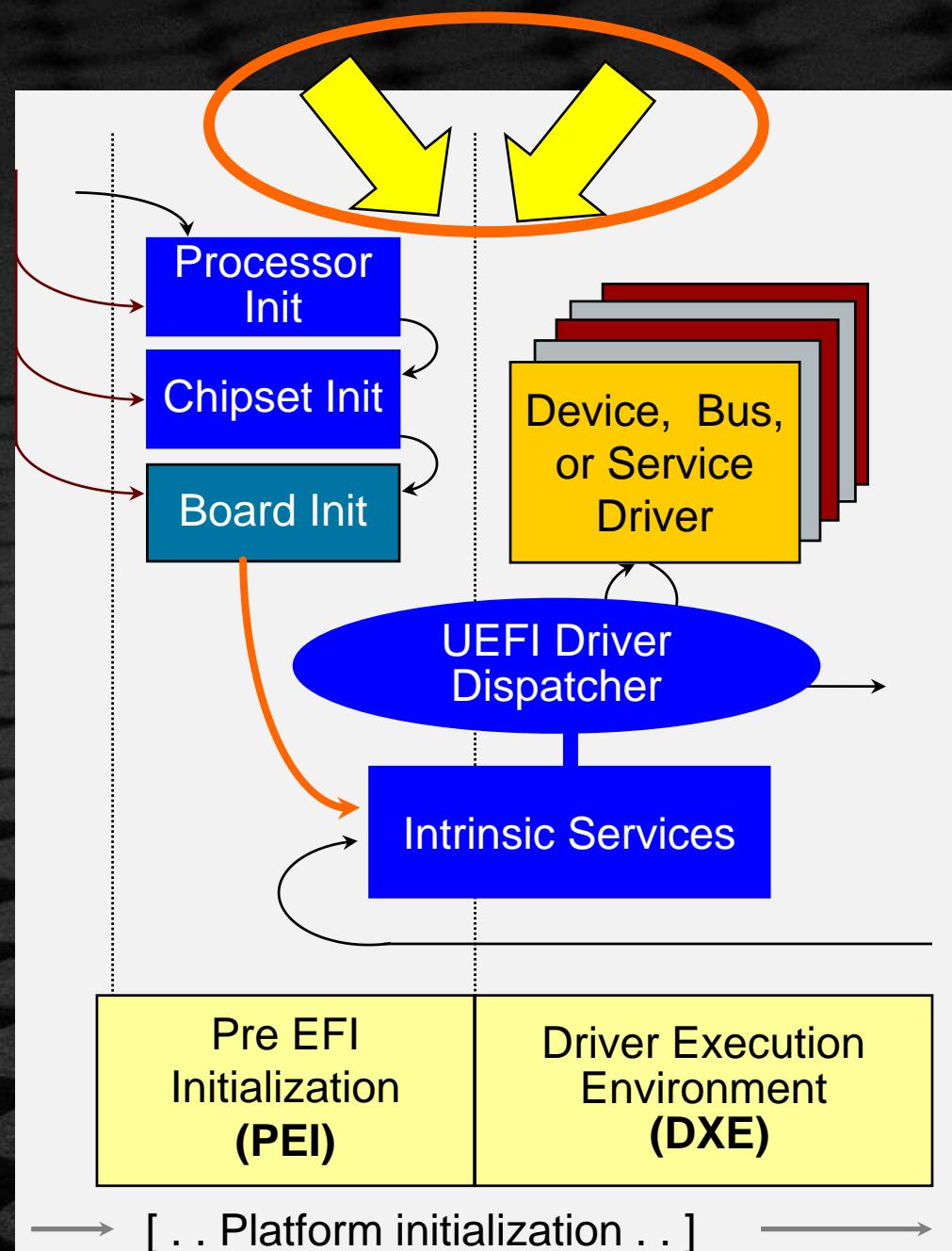
Porting Key

Not Likely
Medium Likely
Very Likely

MinnowBoard Max Silicon

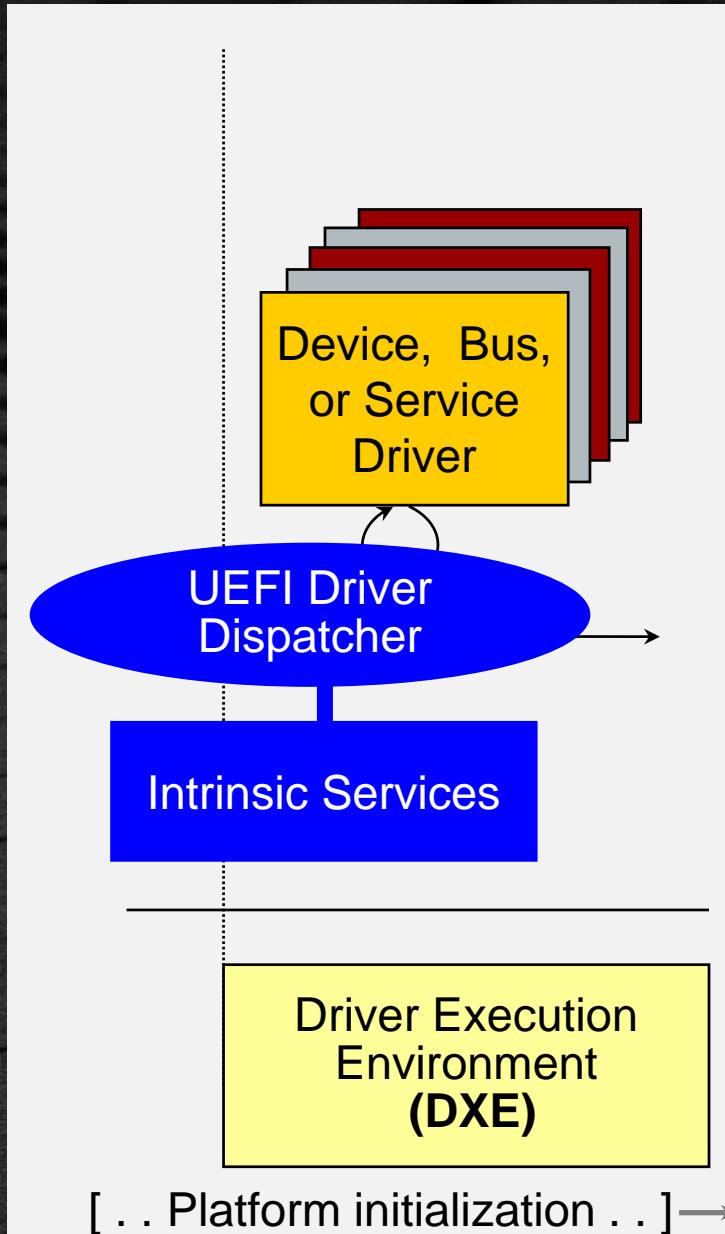
- <Memory Cntrl North>
`v1v2DeviceRefCodePkg/ValleyView2Soc/NorthCluster`
- <I/O Cntrl South>
`v1v2DeviceRefCodePkg/ValleyView2Soc/SouthCluster`

Initial Program Load (IPL) PEIM for DXE



Creates HOBs
Locates DXE main
Switch Stacks
→ **DxeMain()**

DXE Phase



Main platform DXE code in subdirectories of the platform package containing “DXE” in the name

Establish Architectural Protocols

Install any required DXE / UEFI Drivers

DXE Architectural Protocols (AP)

DXE DRIVER NAME	CATEGORY
Watchdog	Generic
Monotonic Counter	Generic
Runtime	Generic
Status Code	Generic
Variable	Generic
CPU	Platform
BDS	Platform
Timer	PC/AT
Metronome	PC/AT
Reset	PC/AT
RTC	PC/AT
Security	Platform

Porting Key

Not Likely

Medium Likely

How to Search for AP Modules



Architecture Protocols

 tianocore

How to Search for AP Modules in the Project

Search the Work Space .INF files
for the Architectural Protocols

Examples

`gEfiCpuArchProtocolGuid`

How to Search for AP Modules in the Project

Search the Work Space .INF files
for the Architectural Protocols

Examples

`gEfiCpuArchProtocolGuid`

`UefiCpuPkg/CpuDxe/` ##PRODUCES
`Vlv2DeviceRefCodePkg/ValleyView2Soc/CPU/CpuInit/Dxe/`
`IA32FamilyCpuPkg/CpuArchDxe` ##PRODUCES

How to Search for AP Modules in the Project

Search the Work Space .INF files
for the Architectural Protocols

Find which module Install the
protocol (i.e Search .c)

Examples

`gEfiCpuArchProtocolGuid`

`gEfiCpuArchProtocolGuid`

How to Search for AP Modules in the Project

Search the Work Space .INF files
for the Architectural Protocols

Find which module Install the
protocol (.i.e Search .c)

Check the NewProjectPkg .DSC to
see which .INF is included in the
[Components] section

Examples

`gEfiCpuArchProtocolGuid`

`gEfiCpuArchProtocolGuid`

`v1v2DeviceRefCodePkg/ValleyView2
Soc/CPU/CpuInit/Dxe/MpCpu.inf`

Location of Architectural Protocols for MAX

Architectural Protocol

EDK II Location

<code>gEfiBdsArchProtocolGuid</code>	<code>MdeModulePkg/Universal/BdsDxe/</code>
<code>gEfiCapsuleArchProtocolGuid</code>	<code>MdeModulePkg/Universal/CapsuleRuntimeDxe</code>
<code>gEfiCpuArchProtocolGuid</code>	<code>Vlv2DeviceRefCodePkg/ValleyView2Soc/CPU/CpuInit/Dxe/MpCpu</code>
<code>gEfiMetronomeArchProtocolGuid</code>	<code>Vlv2TbtDevicePkg/Metronome</code>
<code>gEfiMonotonicCounterArchProtocolGuid</code>	<code>MdeModulePkg/Universal/MonotonicCounterRuntimeDxe</code>
<code>gEfiRealTimeClockArchProtocolGuid</code>	<code>PcAtChipsetPkg/PcatRealTimeClockRuntimeDxe</code>
<code>gEfiResetArchProtocolGuid</code>	<code>Vlv2DeviceRefCodePkg/ValleyView2Soc/SouthCluster/Reset/RuntimeDxe/</code>
<code>gEfiRuntimeArchProtocolGuid</code>	<code>MdeModulePkg/Core/RuntimeDxe</code>
<code>gEfiSecurityArchProtocolGuid</code>	<code>MdeModulePkg/Universal/SecurityStubDxe</code>
<code>gEfiStatusCodeRuntimeProtocolGuid</code>	<code>MdeModulePkg/Universal/ReportStatusCodeRouter/RuntimeDxe</code>
<code>gEfiTimerArchProtocolGuid</code>	<code>Vlv2DeviceRefCodePkg/ValleyView2Soc/SouthCluster/SmartTimer/Dxe</code>
<code>gEfiVariableArchProtocolGuid</code>	<code>MdeModulePkg/Universal/Variable</code>
<code>gEfiVariableWriteArchProtocolGuid</code>	<code>MdeModulePkg/Universal/Variable/RuntimeDxe</code>
<code>gEfiWatchdogTimerArchProtocolGuid</code>	<code>MdeModulePkg/Universal/WatchdogTimerDxe</code>

Platform-Dependent DXE Drivers

Host Bridge Driver

NewProjectPkg/PciPlatformDxe/PciPlatform.c
<Memory Controller North>Pkg/PciHostBridgeDxe/

PCH Initialize Driver

NewProjectPkg/LegacyBiosDxe/Platform
<I/O Cntl South>Pkg/PchInitDxe

SATA Controller Driver

<I/O Cntl South> Pkg/SataControllerDxe/PchSataController

MinnowBoard Max Silicon

- <Memory Cntrl North>
Vlv2DeviceRefCodePkg/ValleyView2Soc/NorthCluster
- <I/O Cntrl South>
Vlv2DeviceRefCodePkg/ValleyView2Soc/SouthCluster

Super I/O

<xSioXVer>Pkg/XSioXVerDxe

MinnowBoard Max SoC Specific Modules

Vlv2DeviceRefCodePkg/
ValleyView2Soc/
CPU
NorthCluster
SouthCluster

Other SOC
Broxton Silicon Ref package
BroxtonSoc/BroxtonSiPkg/

CPU

CpuInit
CpuS3
Dts
EcpOnly
PowerManagement
SmmAccess
SmmControl

North Cluster

ISPDxe
MemoryInit
PciHostBridge
SmBiosMemory
VlvInit

South Cluster

PchInit
Smbus
Spi
SmmControl
Usb
S3Support
SmartTimer
LegacyInterrupt
ActiveBios
Reset
PchSmiDispatcher
Pcie
Pnp
SDControllerDxe
SDMediaDeviceDxe
SataController
SysFwUpdateCapsule
I2cStack

DXE Base Address, PCD Example

DEC

Defined in Package MdePkg DEC

[PcdsFixedAtBuild, PcdsPatchableInModule, PcdsDynamic, PcdsDynamicEx]
gEfiMdePkgTokenSpaceGuid.PcdPciExpressBaseAddress | 0x0 | UINT64 | 0x0000000a

INF

Module INF lists which PCDs get accessed . . .

<PchX>Pkg/Library/DxeRuntimePciLibPciExpress.inf
[Pcd]
gEfiMdePkgTokenSpaceGuid.PcdPciExpressBaseAddress

DSC

Value to use in **NewProjectPkg.dsc**

[PcdsPatchableInModule]
gEfiMdePkgTokenSpaceGuid.PcdPciExpressBaseAddress | 0xE0000000

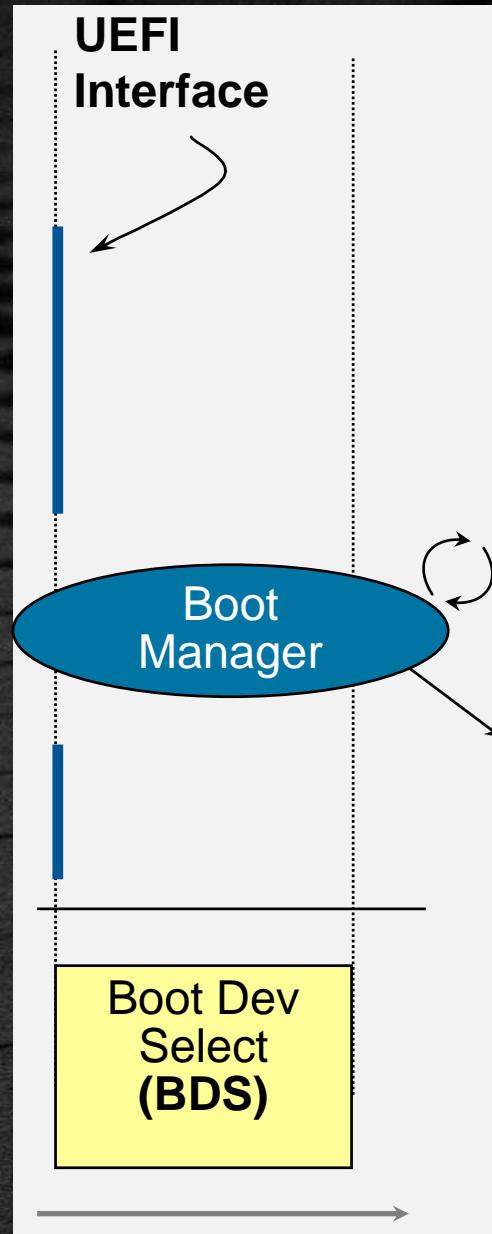
DXE - Referenced in DXE code in

<PchX>Pkg/Library/DxeRuntimePciLibPciExpress.c

```
mPciExpressBaseAddress = (UINTN) PatchPcdGet64  
(PcdPciExpressBaseAddress)
```

Code

BDS Phase



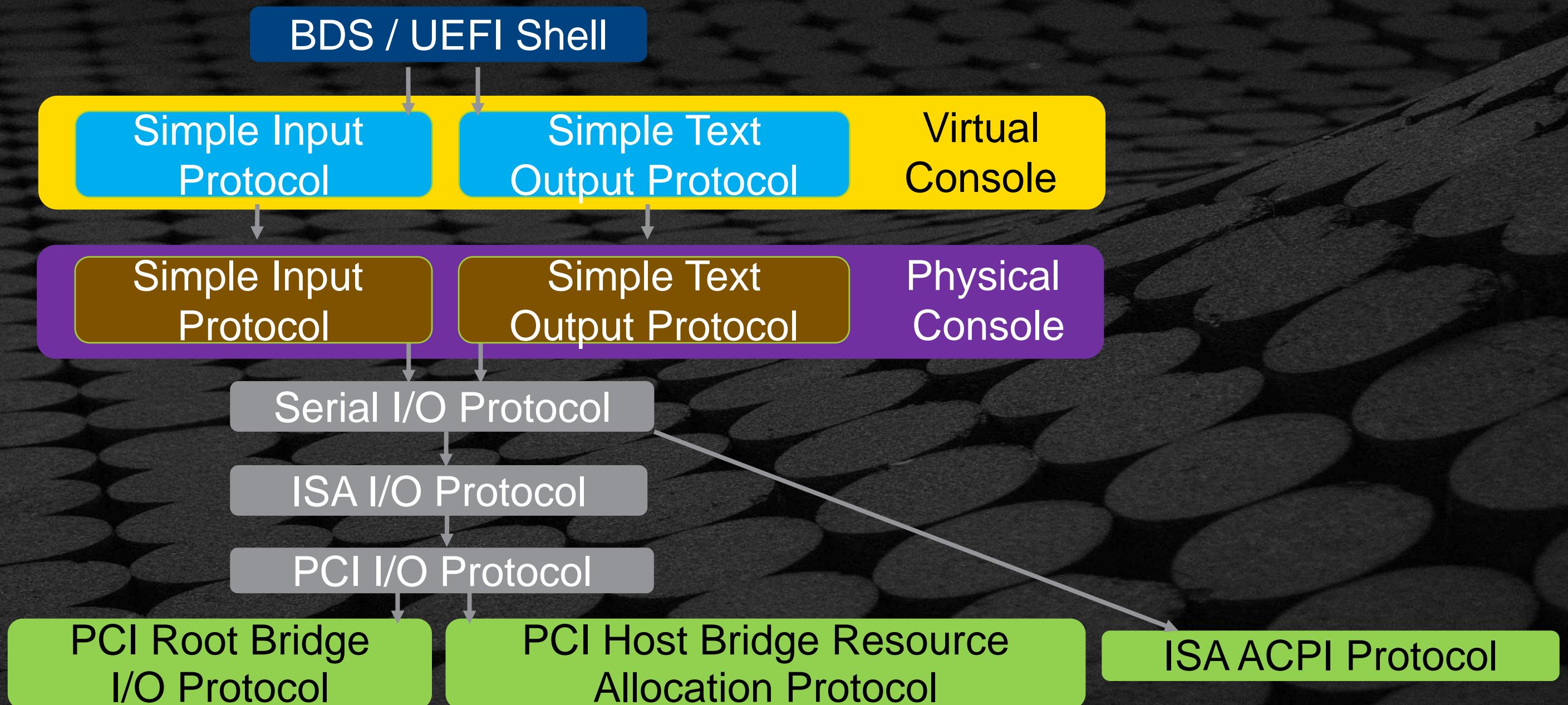
Detect console devices (input and output)

Check enumeration of all devices' preset

Detect boot policy

Ensure BIOS “front page” is loaded

Serial Terminal Console Services



Serial Terminal Console Drivers

DXE DRIVER	CATEGORY	SOURCE
BDS / UEFI Shell	Generic	EDK II
Console Splitter	Generic	
Terminal	Generic	
ISA Serial	PCAT	
ISA Bus	Generic	
PCI Bus	Generic	
Console Platform	Platform	Platform Policy
PCI Root Bridge	Memory Controller North	Intel Silicon
PCI Host Bridge	Memory Controller North	Intel Silicon
ISA ACPI	Super I/O	Super I/O Vendor

Porting Key

Not Likely

Medium Likely

Very Likely

Porting Serial Terminal Console

NewProjectPkg/

 PlatformDxe/
 PlatformSetupDxe /
 ...

 Library/
 PlatformBdsLib/

MdeModulePkg/

 Universal/
 Console/
 ConPlatformDxe/
 ConSplitterDxe/
 GraphicsConsoleDxe/
 TerminalDxe/

Platform

Open Source

Porting Task List



- 1 Create a New Project package directory
 - 2 Create Build Files (DSC, DEC, and FDF)
 - 3 Update target.txt to make your Project the default
 - 4 Use UEFI PI phases to Port all required project modules
 - 5 Update DSC w/ libraries, ported modules, and PCD values
- Update build text files with libraries, ported modules, and PCD values to configure modules

Build Text Files for the Platform

MyWorkSpace/
NewProjectPkg/
Include/
Library/

PlatformDrivers/

Flash Map layout →

NewProjectPkg.DSC ← **PCD Values - Library classes**
NewProjectPkg.FDF
NewProjectPkg.DEC ← **Defines for a platform**

Check the [EDK II Specifications](#)

Update the New Project DEC File

Declare

Syntax:

```
<DECfile> ::=  <Defines>
               Include
               [<LibraryClass>]
               [<Guids>]
               [<Protocols>]
               [<Ppis>]
               [<Pcd>]
               [<UserExtensions>]
```

Example: DEC File

```
[Defines]
DEC_SPECIFICATION          = 0x00010005
PACKAGE_NAME                = NewProjectPkg
PACKAGE_GUID                 = 463B3B00-0D18-4a5f-90C0-D5B851D2574B
PACKAGE_VERSION              = 0.1

[Includes]
.
Include
Include\Library

[Ppis]
gPeiSpeakerInterfacePpiGuid = { 0x30ac275e, 0xbb30, 0x4b84, { 0xa1, 0xcd, 0x0a, 0xf1, 0x32, 0x2c, 0x89, 0xc0 } }
gPeiUsbControllerPpiGuid    = { 0x3BC1F6DE, 0x693E, 0x4547, { 0xA3, 0x00, 0x21, 0x82, 0x3C, 0xA4, 0x20, 0xB2 } }
gPeiMfgMemoryTestPpiGuid   = { 0xab294a92, 0xeaf5, 0x4cf3, { 0xab, 0x2b, 0x2d, 0x4b, 0xed, 0x4d, 0xb6, 0x3d } }
gPeiSha256HashPpiGuid      = { 0x950e191b, 0x8524, 0x4f51, { 0x80, 0xa1, 0x5c, 0x4f, 0x1b, 0x03, 0xf3, 0x5c } }

[Guids]
gEfiPlatformBootModeGuid   = { 0xce845704, 0x1683, 0x4d38, { 0xa4, 0xf9, 0x7d, 0x0b, 0x50, 0x77, 0x57,
```

[Link to DEC File](#)

Update the New Project DSC File

Syntax:

```
DSCfile ::= [<Header>]  
           [<Defines>]  
           [<SkuIds>]  
           [<Libraries>]  
           [<LibraryClasses>]  
           [<Pcds>]  
           [<Components>]  
           [<UserExtensions>]
```

Description

Example: DSC File

```
[Defines]
PLATFORM_NAME          = NewProjectPkg
PLATFORM_GUID           = 465B0A0B-7AC1-443b-8F67-7B8DEC145F90
PLATFORM_VERSION         = 0.1
DSC_SPECIFICATION       = 0x00010005
#
# Set platform specific package/folder name, same as passed from PREBUILD script.
# PLATFORM_PACKAGE would be the same as PLATFORM_NAME as well as package build folder
# DEFINE only takes effect at R9 DSC and FDF.
#
DEFINE      PLATFORM_PACKAGE          = NewProjectPkg
DEFINE      PLATFORM_ECP_PACKAGE      = R8V1v2TbltDevicePkg
DEFINE      PLATFORM_RC_PACKAGE        = V1v2DeviceRefCodePkg
DEFINE      DEVICE_COMMON_PACKAGE     = DeviceCommonPkg
DEFINE      PLATFORM_AMI_CSM_PACKAGE   = V1v2DeviceAmiCsmPkg
DEFINE      PLATFORM_INTEL_RESTRICT_PACKAGE = V1v2TbltDeviceMiscPkg
OUTPUT_DIRECTORY          = Build/${(PLATFORM_PACKAGE)}
SUPPORTED_ARCHITECTURES    = IA32|X64
BUILD_TARGETS              = DEBUG|RELEASE
SKUID_IDENTIFIER           = DEFAULT

DEFINE CPU_ARCH                =ValleyView2
DEFINE PROJECT_SC_FAMILY       =IntelPch
DEFINE PROJECT_SC_ROOT         =../../$(PLATFORM_RC_PACKAGE)/ValleyView2Soc/SouthCluster
DEFINE PROJECT_VLV_ROOT         =../../$(PLATFORM_RC_PACKAGE)/ValleyView2Soc/NorthCluster
DEFINE CHIPSET_MEMORY_CONTROLLER=ValleyView2
```

[Link to DSC File](#)

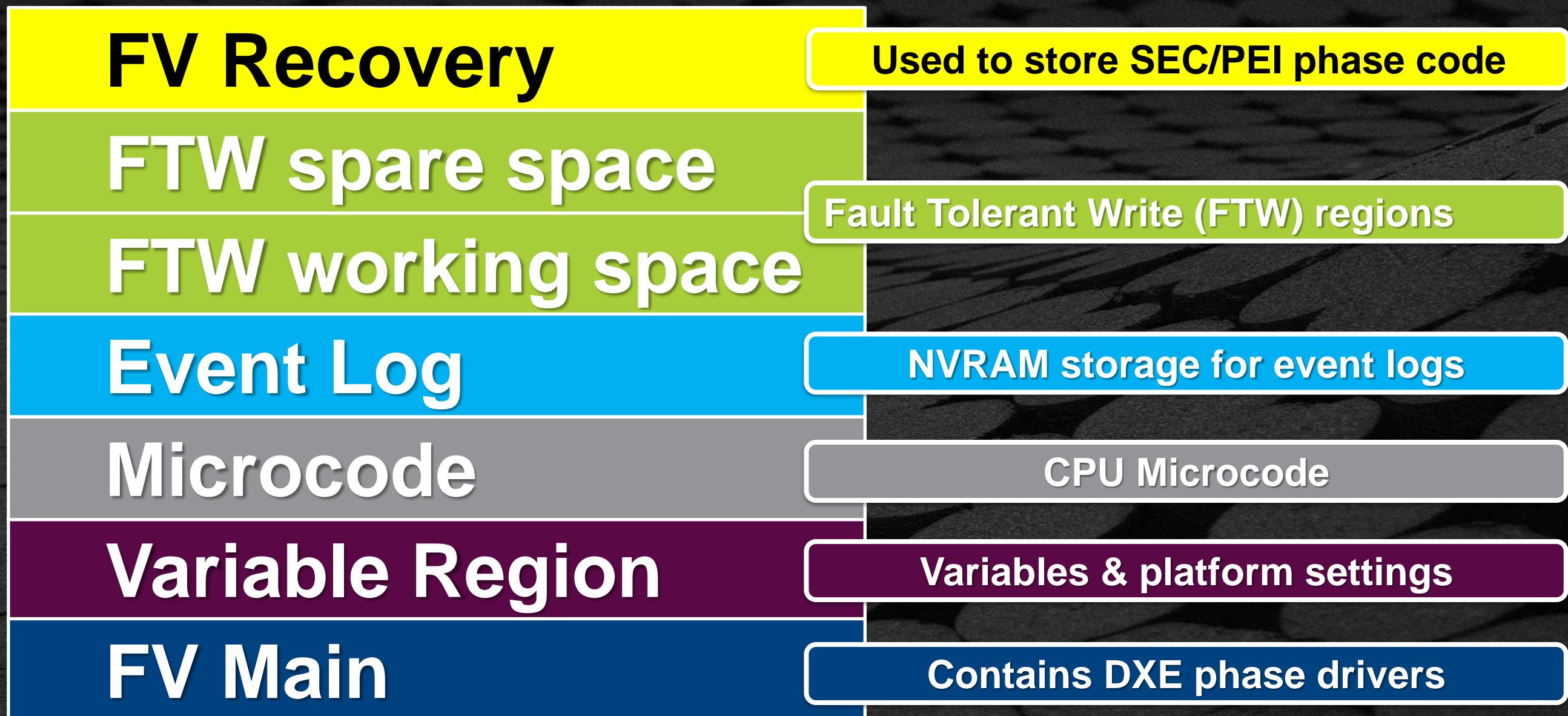
Update the New Project FDF File

Syntax:

```
FDFfile ::= [<Header>]  
          [<Defines>]  
          <FD>  
          <FV>  
          [<Capsule>]  
          [<VTF>]  
          [<Rules>]  
          [<OptionRom>]  
          [<UserExtensions>]
```

Flash Layout

Update the New Project FDF File - Layout



Example: FDF File

```
[Defines]
DEFINE FLASH_BASE      = 0xFFC00000          #The base address of the 4Mb FLASH Device.
DEFINE FLASH_SIZE       = 0x00400000          #The flash size in bytes of the 4Mb FLASH Device.
DEFINE FLASH_BLOCK_SIZE = 0x1000              #The block size in bytes of the 4Mb FLASH Device.
DEFINE FLASH_NUM_BLOCKS = 0x400               #The number of blocks in 4Mb FLASH Device.
DEFINE FLASH_AREA_BASE_ADDRESS
DEFINE FLASH_AREA_SIZE

DEFINE FLASH_REGION_VLVMICROCODE_OFFSET
DEFINE FLASH_REGION_VLVMICROCODE_SIZE
DEFINE FLASH_REGION_VLVMICROCODE_BASE

DEFINE FLASH_REGION_VPD_OFFSET
DEFINE FLASH_REGION_VPD_SIZE

DEFINE FLASH_REGION_NVSTORAGE_SUBREGION_NV_FTW_WORKING_OFFSET = 0x0007E000
DEFINE FLASH_REGION_NVSTORAGE_SUBREGION_NV_FTW_WORKING_SIZE     = 0x00002000

DEFINE FLASH_REGION_NVSTORAGE_SUBREGION_NV_FTW_SPARE_OFFSET    = 0x00080000
DEFINE FLASH_REGION_NVSTORAGE_SUBREGION_NV_FTW_SPARE_SIZE      = 0x00040000
```

Link FDF File Example

FDF File Example

```
[Fv.Root]
FvAlignment = 64
ERASE_POLARITY = 1
MEMORY_MAPPED = TRUE
STICKY_WRITE = TRUE
. .
INF VERSION = "1" $(WS)/EdkNt32Pkg/ Dxe/WinNtThunk/Cpu/Cpu.inf
FILE DXE_CORE = B5596C75-37A2-4. . .
SECTION COMPRESS {
  DEFINE DC = $(WS)/Build/NewProjectPkg /DEBUG_MYTOOLS
    SECTION PE32 = $(DC)/B5596C75-3 . .
    SECTION VERSION "1.2.3"
}
FILE FV_IMAGE = EF41A0E1-40B1-481 . .
FvAlignment = 512K
WRITE_POLICY_RELIABLE = TRUE
SECTION GUIDED 3EA022A4-1439-4 . .
SECTION FV_IMAGE = Dxe {
  APRIORI DXE {
    INF NewProject/a/a.inf
    INF MdePkg/x/y/z.inf
    INF NewProject/a/b/b.inf
  }
  INF a/d/d.inf
  . .
}
}

[ ]
FDF File Example
```

Porting Task List



- 1 Create a New Project package directory
- 2 Create Build Files (DSC, DEC, and FDF)
- 3 Update target.txt to make your Project the default
- 4 Use UEFI PI phases to Port all required project modules
- 5 Double check the Protocol, Driver dependencies and libraries to boot to the UEFI Shell
- 6 Minimums for UEFI Shell

Minimum Drivers for UEFI Shell

DRIVER	PHASE	TARGET DEVICE
Memory Controller	PEI	<Memory Cntrl North>
SMBUS	PEI	<I/O Cntrl South>
Motherboard	PEI	Platform
Security	DXE	Platform
Status Code	DXE	Platform
Console Platform	DXE	Platform
PCI Root Bridge	DXE	<Memory Cntrl North>
PCI Host Bridge	DXE	<Memory Cntrl North>
ISA ACPI	DXE	Super I/O /PcAtChipset

Porting Key

Not Likely

Medium Likely

Very Likely

MinnowBoard Max

- <Memory Cntrl North>
`V1v2DeviceRefCodePkg/ValleyView2Soc/NorthCluster`
- <I/O Cntrl South>
`V1v2DeviceRefCodePkg/ValleyView2Soc/SouthCluster`

Minimum Libraries for UEFI Shell

Library	Phase	Notes
PCI	PEI	Only for non-standard IO
TimerLib	PEI	<I/O Cntrl South>
SmBusLib	PEI	<I/O Cntrl South>
SerialPortLib	PEI	<I/O Cntrl South>
Flash Reading Lib(s)	DXE	Flash layout
FVB	DXE	<I/O Cntrl South> Firmware Vol Block
Section Reading Lib(s)	DXE	Extraction Methods
Status Code Output Lib(s)	DXE	Depends on method

Porting Key

Not Likely

Medium Likely

Very Likely

MinnowBoard Max

- <I/O Cntrl South>
Vlv2DeviceRefCodePkg/ValleyView2Soc/SouthCluster

Porting Summary: New Package Directory

MyWorkSpace/
NewProjectPkg/
Include/
Library/

PlatformDrivers/

NewProjectPkg.DSC

NewProjectPkg.FDF

NewProjectPkg.DEC

Ported Package

- ★ All ported libraries
- ★ All ported drivers
- ★ Values – Library classes
- ★ Flash Map layout
- ★ Defines per platform

SUMMARY

- ★ Define the porting task list for porting existing platforms in EDK II in order to boot to the UEFI Shell
- ★ Explain the EDK II infrastructure, porting libraries, library classes, PCDs, and directory structures
- ★ Determine the necessary porting for each phase of a new EDK II platform Project

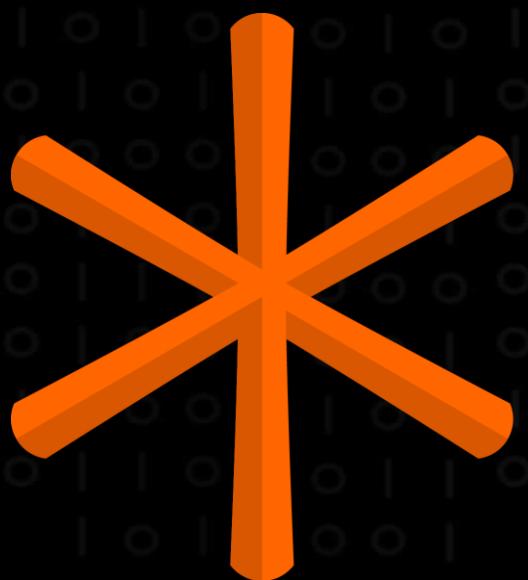
Questions?



Return to Main Training Page



Return to Training Table of contents for next presentation [link](#)



tianocore

