UEFI & EDK II Training

Platform Build Lab Up Xtreme- Linux

tianocore.org

Copy and Paste see Lab Guide.md

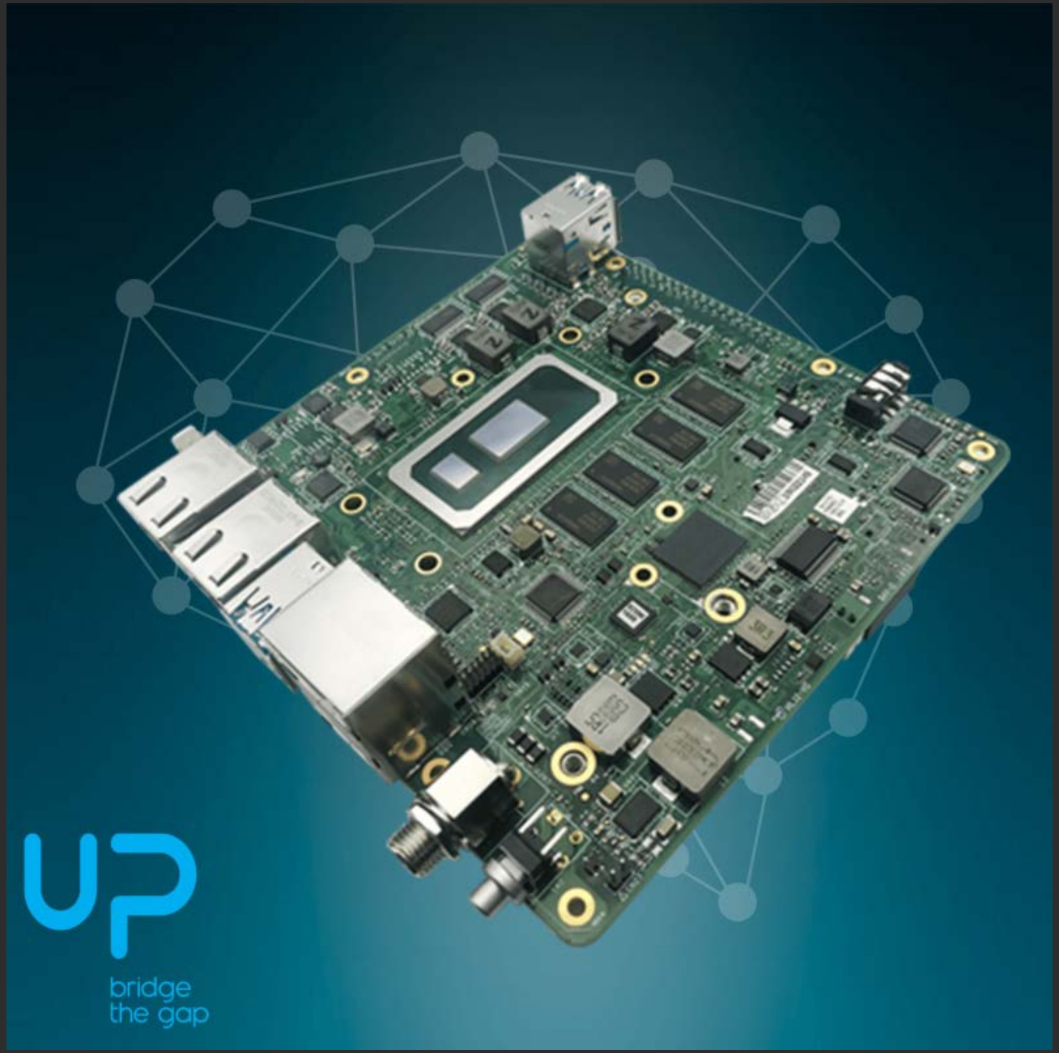
PLATFORM BUILD LABS

- Download Minplatform Using Git from tianocore.org
- Build a EDK II Platform using Up Xtreme Aaeon board

DOWNLOAD MINPLATFORM

Use Git to download EDK II and MinPlatform

EDK II Platform – Up Xtreme by Aaeon



8th Generation Intel® Core™
U-Series processors
(Formerly Whiskey Lake)

UP Board products
Up Shop

Linux setup for Up Xtreme Lab

ubuntu

Lab Setup Requirements – Ubuntu 16.04

```
bash$ sudo apt-get install build-essential uuid-dev iasl git gcc-5 nasm
bash$ sudo apt-get install screen
bash$ sudo apt-get install gcab
```

Clear
Linux*
Project

Lab Setup Requirements – Clear Linux* Project

```
bash$ sudo swupd bundle-add devpkg-util-linux
bash$ sudo swupd bundle-add devpkg-gcab
```

Open Terminal Prompt.

Cd to the Workspace and create the Up Xtreme build directory “UpX”

```
bash$ cd ~/src
bash$ mkdir UpX
bash$ cd UpX
```

Download the source for Edk II, MinPlatform and FSP

From a terminal prompt at `~/src/Upx` , do the following:

- Edk2

For SHA to checkout see Lab Guide.md

```
$ git clone --recursive https://github.com/tianocore/edk2
```

- Edk2-platforms

```
$ git clone https://github.com/tianocore/edk2-platforms.git
```

- Edk2-non-osi

```
$ git clone https://github.com/tianocore/edk2-non-osi.git
```

- FSP

```
$ git clone https://github.com/IntelFsp/FSP.git
```

Set PROXYS FIRST

```
$ git config --global https.proxy=proxy.hf.intel.com:911
$ git config --global http.proxy=proxy.hf.intel.com:911
```

Takes about 6 minutes

Download MinPlatform Lab Material

Download the PlatformBuildLab_MinPlatform_FW.zip from : github.com
PlatformBuildLab2_FW.zip
OR
Use **`git clone`** to download the PlatformBuildLab_MinPlatform_FW

```
C:/> git clone https://github.com/tianocore-training/PlatformBuildLab_MinPlatform_FW.git
```

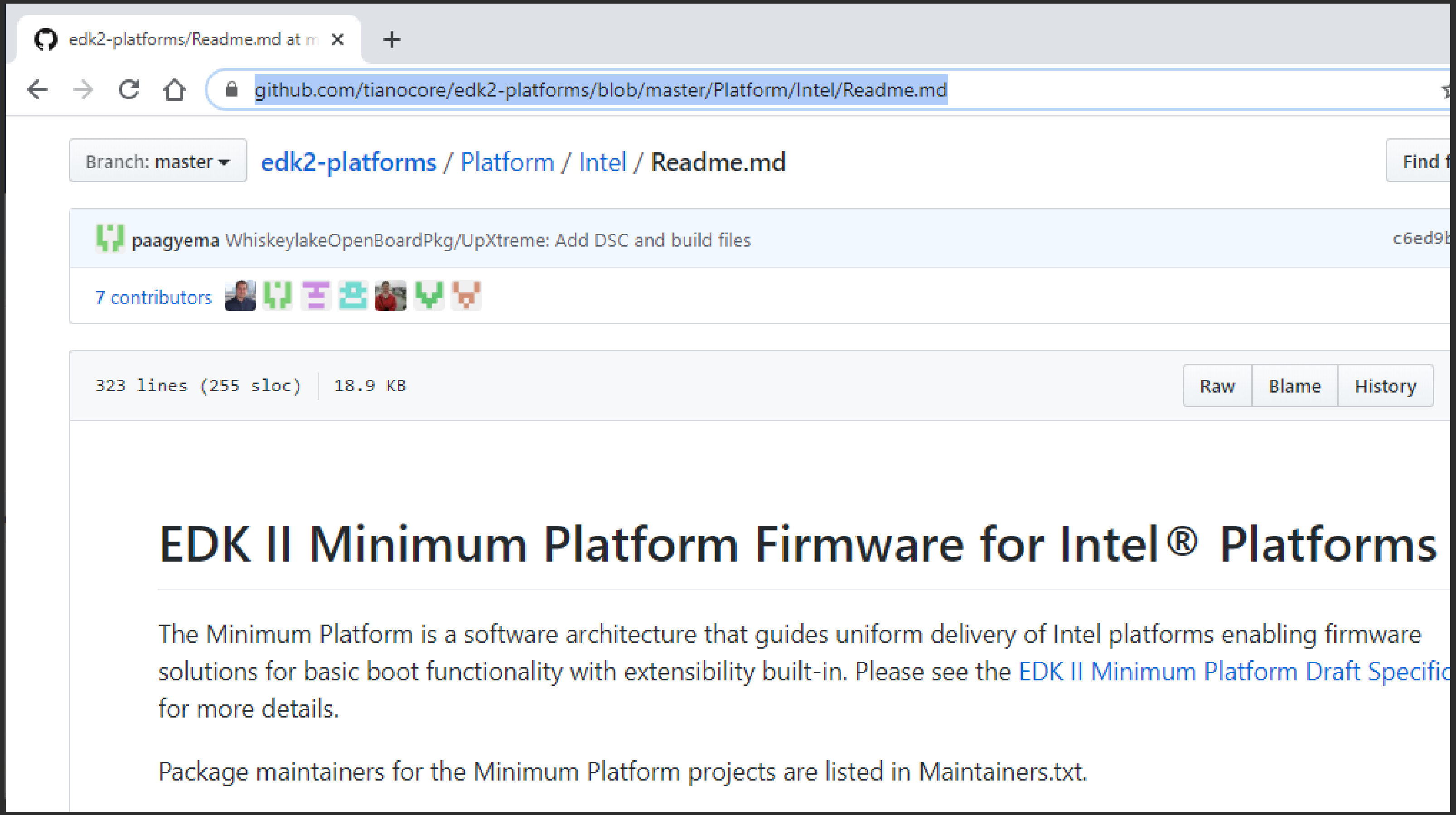
Directory PlatformBuildLab_MinPlatform_FW will be created

```
/FW
 /MinPlatformBuild
     - UpX_Lab                          - Lab Material
      . . .
```

BUILD UP XTREME

Where to get Open Source Up Xtreme

How to Download & Build: Open Source MinPlatform Readme.md

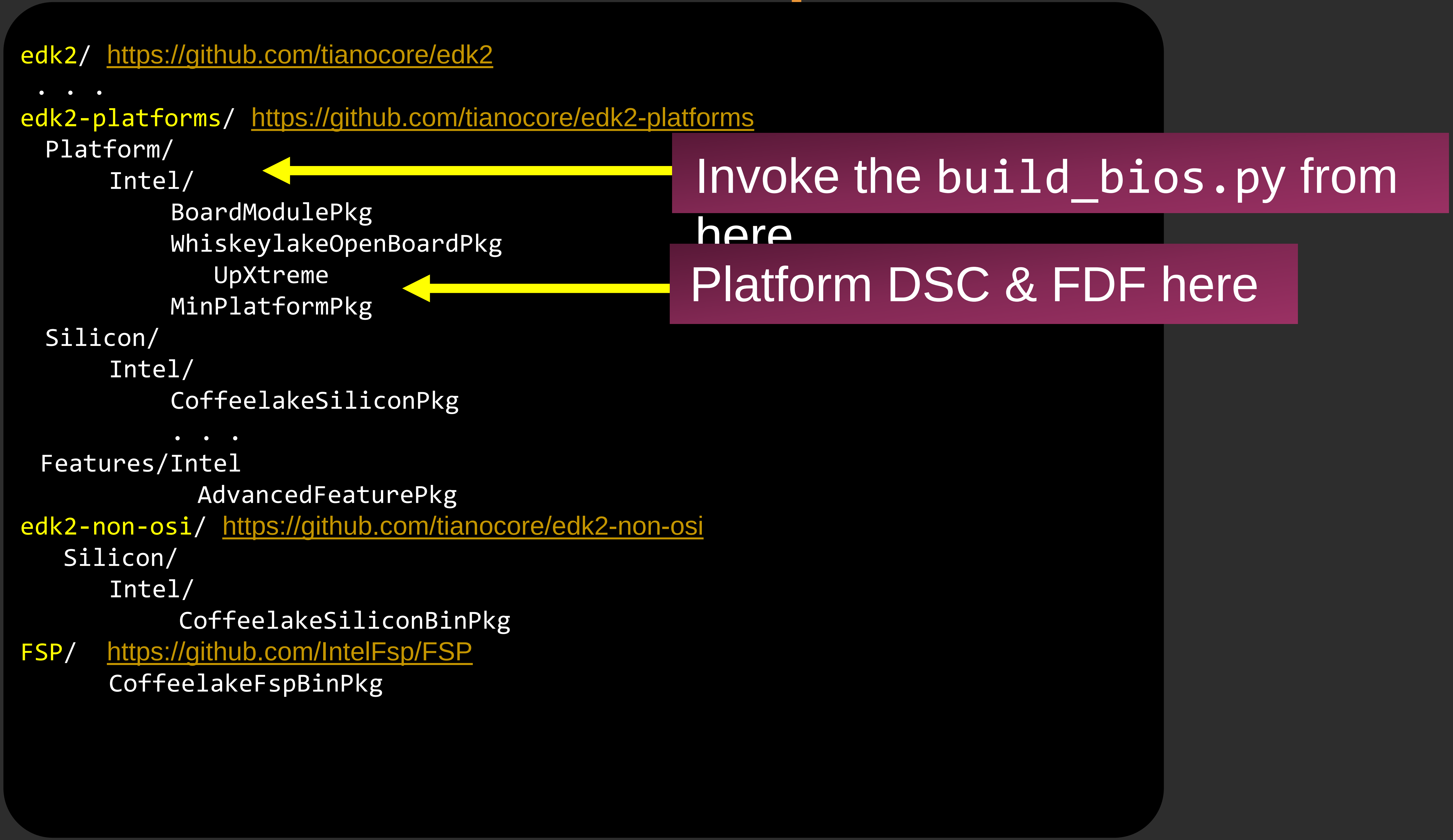

edk2-platforms/Readme.md at m
github.com/tianocore/edk2-platforms/blob/master/Platform/Intel/Readme.md
Branch: master
edk2-platforms / Platform / Intel / Readme.md
Find f
paagyema WhiskeylakeOpenBoardPkg/UpXtreme: Add DSC and build files
c6ed9b
7 contributors
323 lines (255 sloc) | 18.9 KB
Raw
Blame
History

EDK II Minimum Platform Firmware for Intel® Platforms

The Minimum Platform is a software architecture that guides uniform delivery of Intel platforms enabling firmware solutions for basic boot functionality with extensibility built-in. Please see the EDK II Minimum Platform Draft Specific for more details.

Package maintainers for the Minimum Platform projects are listed in Maintainers.txt.

MinPlatform Open Board Tree Structure



```
edk2/ https://github.com/tianocore/edk2
 . . .
edk2-platforms/ https://github.com/tianocore/edk2-platforms
  Platform/
      Intel/
           BoardModulePkg
           WhiskeylakeOpenBoardPkg
              UpXtreme
           MinPlatformPkg
  Silicon/
      Intel/
           CoffeelakeSiliconPkg
           . . .
  Features/Intel
             AdvancedFeaturePkg
edk2-non-osi/ https://github.com/tianocore/edk2-non-osi
   Silicon/
      Intel/
            CoffeelakeSiliconBinPkg
FSP/   https://github.com/IntelFsp/FSP
      CoffeelakeFspBinPkg
```

Invoke the build_bios.py from here

Platform DSC & FDF here

Build Environment

Open a Terminal Command Prompt

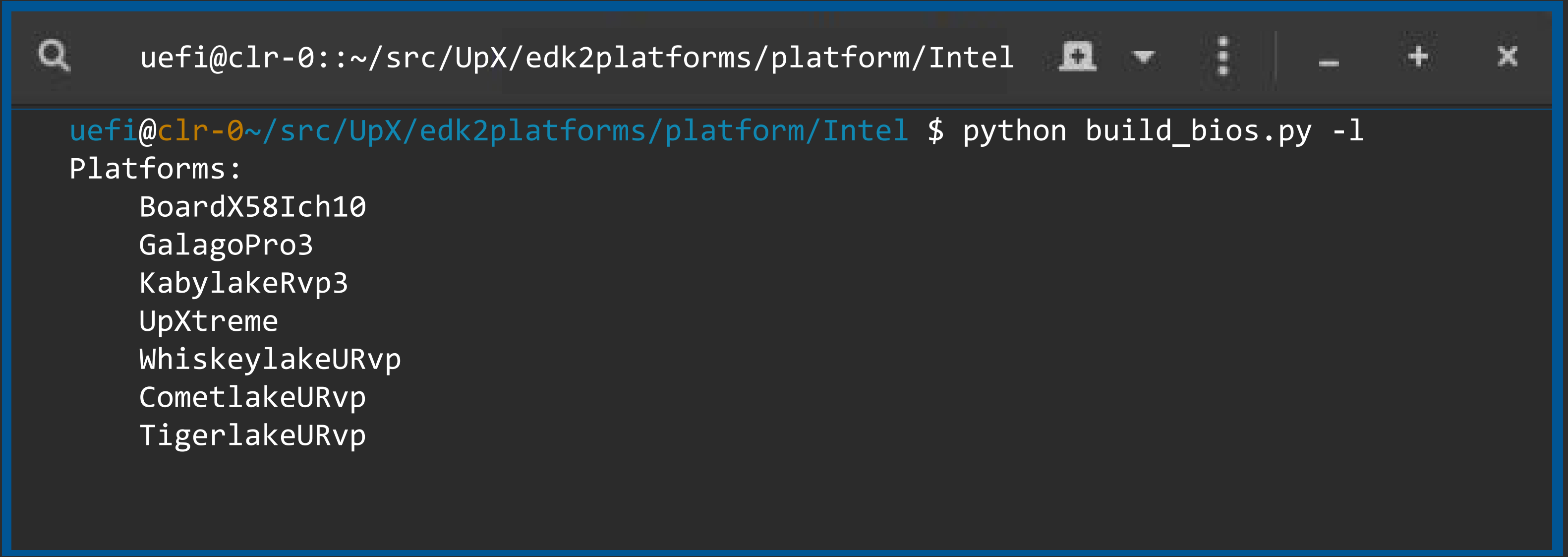
```
bash$ cd ~/src/UpX/edk2
bash$ source edksetup.sh
bash$ cd ..
bash$ cd edk2-platforms/Platform/Intel
```

Check if Python okay (may also need to set PYTHON_HOME)

```
bash$  python --version
Python 3.8.2
```

Check for available `MinPlatform` Boards

```
bash$ python build_bios.py -l
```



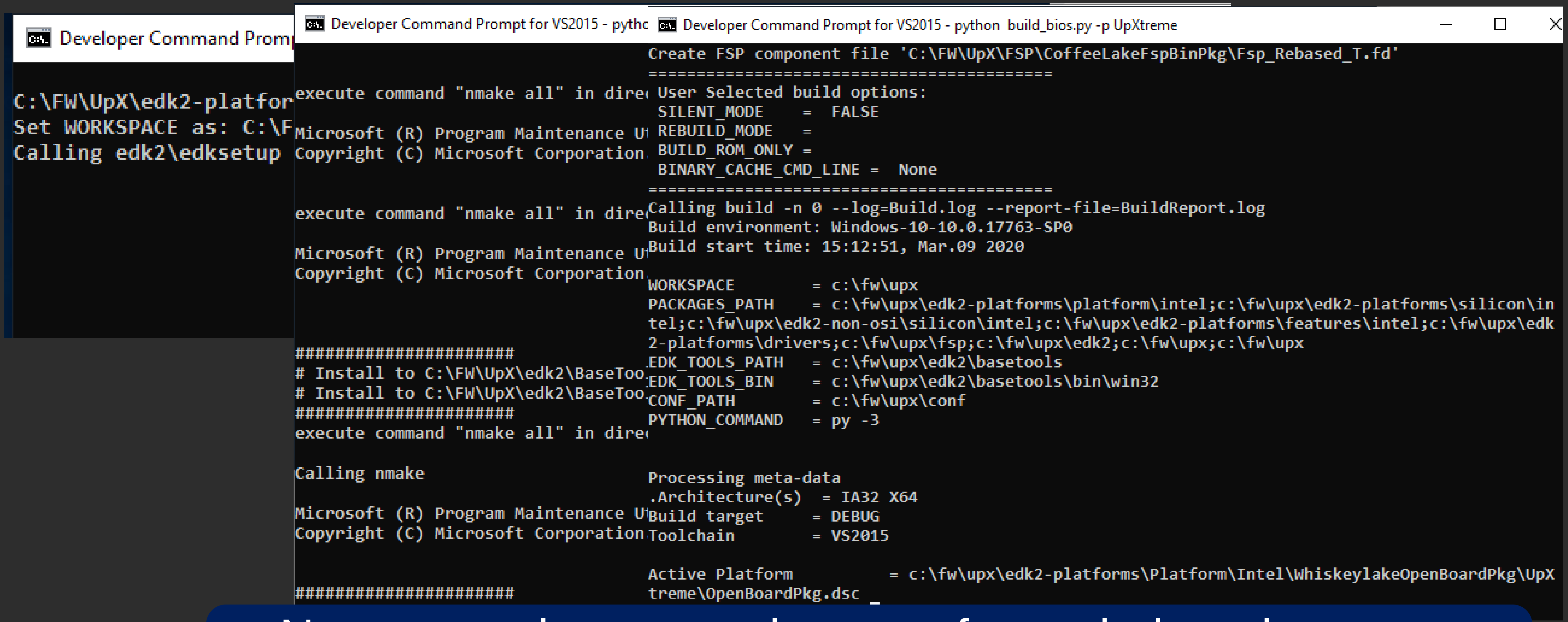
```
uefi@clr-0~/src/UpX/edk2platforms/platform/Intel $ python build_bios.py -l
Platforms:
    BoardX58Ich10
    GalagoPro3
    KabylakeRvp3
    UpXtreme
    WhiskeylakeURvp
    CometlakeURvp
    TigerlakeURvp
```

Invoke the Build

Invoke the Python Build script for Up Xtreme

```
$> python build_bios.py -p UpXtreme  -t GCC5
```

Takes about 16 minutes



```
C:\FW\UpX\edk2-platfor
Set WORKSPACE as: C:\F
Calling edk2\edksetup
```

```
execute command "nmake all" in dire

Microsoft (R) Program Maintenance U
Copyright (C) Microsoft Corporation


execute command "nmake all" in dire

Microsoft (R) Program Maintenance U
Copyright (C) Microsoft Corporation



######################
# Install to C:\FW\UpX\edk2\BaseToo
# Install to C:\FW\UpX\edk2\BaseToo
######################
execute command "nmake all" in dire

Calling nmake

Microsoft (R) Program Maintenance U
Copyright (C) Microsoft Corporation


######################
```

Developer Command Prompt for VS2015 - python build_bios.py -p UpXtreme

```
Create FSP component file 'C:\FW\UpX\FSP\CoffeeLakeFspBinPkg\Fsp_Rebased_T.fd'
=========================================
User Selected build options:
 SILENT_MODE     =  FALSE
 REBUILD_MODE    =
 BUILD_ROM_ONLY =
 BINARY_CACHE_CMD_LINE =  None
=========================================
Calling build -n 0 --log=Build.log --report-file=BuildReport.log
Build environment: Windows-10-10.0.17763-SP0
Build start time: 15:12:51, Mar.09 2020

WORKSPACE        = c:\fw\upx
PACKAGES_PATH    = c:\fw\upx\edk2-platforms\platform\intel;c:\fw\upx\edk2-platforms\silicon\in
tel;c:\fw\upx\edk2-non-osi\silicon\intel;c:\fw\upx\edk2-platforms\features\intel;c:\fw\upx\edk
2-platforms\drivers;c:\fw\upx\fsp;c:\fw\upx\edk2;c:\fw\upx;c:\fw\upx
EDK_TOOLS_PATH   = c:\fw\upx\edk2\basetools
EDK_TOOLS_BIN    = c:\fw\upx\edk2\basetools\bin\win32
CONF_PATH        = c:\fw\upx\conf
PYTHON_COMMAND   = py -3


Processing meta-data
.Architecture(s)  = IA32 X64
Build target     = DEBUG
Toolchain        = VS2015

Active Platform          = c:\fw\upx\edk2-platforms\Platform\Intel\WhiskeylakeOpenBoardPkg\UpX
treme\OpenBoardPkg.dsc _
```

Note example screen shots are from windows but same information would be on the terminal screen for Linux

Platform Build Scripts

Platform Config

Many Platforms have a bash, bat or Python script file to pre or post process the EDK II build process

For MinPlatform platform specific config
Build processing:
`Build_config.cfg` – Lists directories required for the build and build settings

Link to Up Xtreme Build_config.cfg

Examine Build Parameters

```
Python build_bios.py -p UpXtreme

. . .
Calling build -n 0 --log=Build.log --report-file=BuildReport.log
 and from UpX\conf\target.txt
```

TARGET	= DEBUG	Build Mode
TARGET_ARCH	= IA32 X64	CPU Architecture
TOOL_CHAIN_TAG	= GCC5	VS Tool Chain
ACTIVE_PLATFORM	= ... /WhiskylakeOpenBoardPkg/ UpXtreme/OpenBoardPkg.dsc	Platform DSC file
Report file created (via python script)	= BuildReport.log	PCDs, Libs, etc.

Platform Build and PCD Parameters

Platform Parameters

Many Platform Parameters are defined in a top .DSC file that controls PCD and build switches

For Up Xtreme : `edk2-platforms/Platform/Intel/WhiskeylakeOpenBoardPkg/UpXtreme`
`OpenBoardPkgPcd.dsc and OpenBoardPkgBuildOption.dsc`

Example:

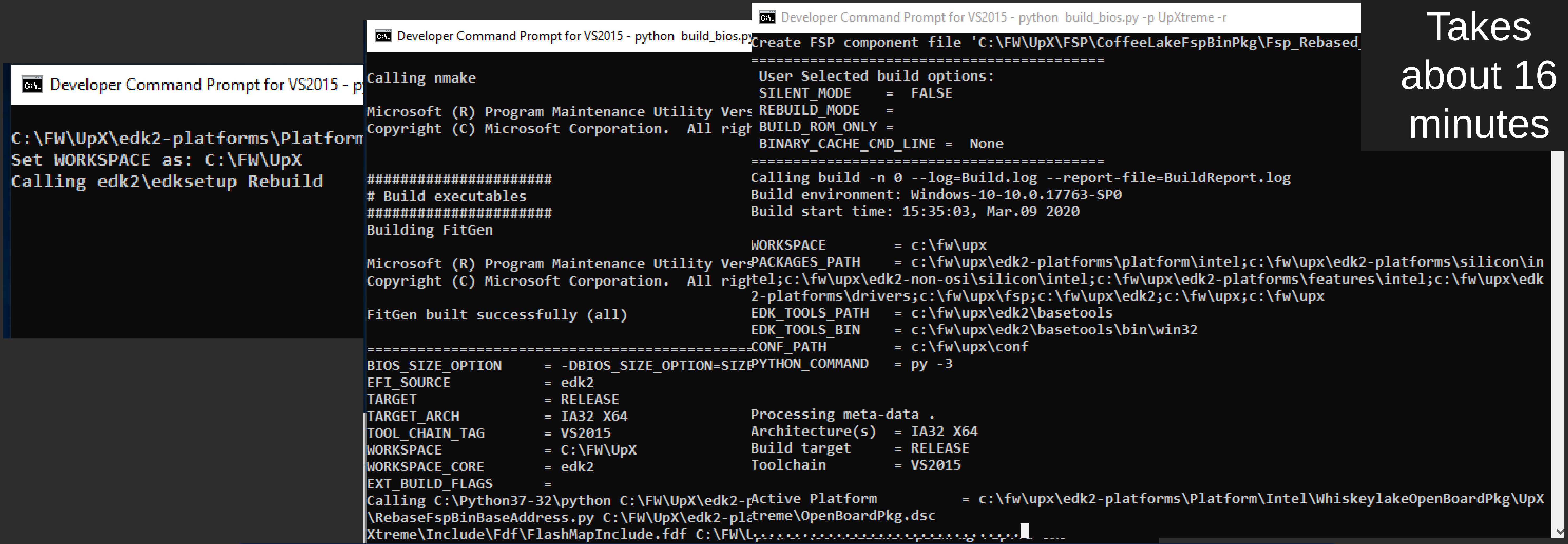
```
# Define Build Options both for EDK and EDKII drivers.

  DEFINE DSC_S3_BUILD_OPTIONS =
  DEFINE DSC_CSM_BUILD_OPTIONS =

!if gSiPkgTokenSpaceGuid.PcdAcpiEnable == TRUE
  DEFINE DSC_ACPI_BUILD_OPTIONS = -DACPI_SUPPORT=1
!else
  DEFINE DSC_ACPI_BUILD_OPTIONS =
!endif

  DEFINE BIOS_GUARD_BUILD_OPTIONS =
  DEFINE OVERCLOCKING_BUILD_OPTION =
```

Build Process for RELEASE Target

Invoke the Python Build script for Up Xtreme

```
bash$ python build_bios.py -p UpXtreme -r -t GCC5
```

Takes about 16 minutes



Developer Command Prompt for VS2015 - p

```
C:\FW\UpX\edk2-platforms\Platform
Set WORKSPACE as: C:\FW\UpX
Calling edk2\edksetup Rebuild
```

Developer Command Prompt for VS2015 - python build_bios.py

```
Calling nmake

Microsoft (R) Program Maintenance Utility Vers
Copyright (C) Microsoft Corporation.  All righ


######################
# Build executables
######################
Building FitGen

Microsoft (R) Program Maintenance Utility Vers
Copyright (C) Microsoft Corporation.  All righ

FitGen built successfully (all)

=============================================
BIOS_SIZE_OPTION      = -DBIOS_SIZE_OPTION=SIZE
EFI_SOURCE            = edk2
TARGET                = RELEASE
TARGET_ARCH           = IA32 X64
TOOL_CHAIN_TAG        = VS2015
WORKSPACE             = C:\FW\UpX
WORKSPACE_CORE        = edk2
EXT_BUILD_FLAGS       =
Calling C:\Python37-32\python C:\FW\UpX\edk2-p
\RebaseFspBinBaseAddress.py C:\FW\UpX\edk2-pla
Xtreme\Include\Fdf\FlashMapInclude.fdf C:\FW\U
```

Developer Command Prompt for VS2015 - python build_bios.py -p UpXtreme -r

```
Create FSP component file 'C:\FW\UpX\FSP\CoffeeLakeFspBinPkg\Fsp_Rebased_
==========================================
 User Selected build options:
 SILENT_MODE     =  FALSE
 REBUILD_MODE    =
 BUILD_ROM_ONLY  =
 BINARY_CACHE_CMD_LINE =  None
==========================================
Calling build -n 0 --log=Build.log --report-file=BuildReport.log
Build environment: Windows-10-10.0.17763-SP0
Build start time: 15:35:03, Mar.09 2020

WORKSPACE         = c:\fw\upx
PACKAGES_PATH     = c:\fw\upx\edk2-platforms\platform\intel;c:\fw\upx\edk2-platforms\silicon\in
tel;c:\fw\upx\edk2-non-osi\silicon\intel;c:\fw\upx\edk2-platforms\features\intel;c:\fw\upx\edk
2-platforms\drivers;c:\fw\upx\fsp;c:\fw\upx\edk2;c:\fw\upx;c:\fw\upx
EDK_TOOLS_PATH    = c:\fw\upx\edk2\basetools
EDK_TOOLS_BIN     = c:\fw\upx\edk2\basetools\bin\win32
CONF_PATH         = c:\fw\upx\conf
PYTHON_COMMAND    = py -3


Processing meta-data .
Architecture(s)   = IA32 X64
Build target      = RELEASE
Toolchain         = VS2015

Active Platform          = c:\fw\upx\edk2-platforms\Platform\Intel\WhiskeylakeOpenBoardPkg\UpX
treme\OpenBoardPkg.dsc
.................................. ...
```

Note example screen shots are from windows but same information would be on the terminal screen for Linux

DEBUG & RELEASE Differences

Slower boot because the time it takes to display debug info

Larger image because of debug code & embedded info

Uses the serial port for debug string output

Contains detailed debug strings that show
the boot process and various ASSERT/TRACE errors

Make a Change

Directory: `~/MinPlatformBuildLab_FW/FW/MinPlatformBuildLab/UpX_Lab`

Copy `Logo.bmp` to `~/src/UpX/edk2/MdeModulePkg/Logo`

Or create a .BMP with your favorite Paint application

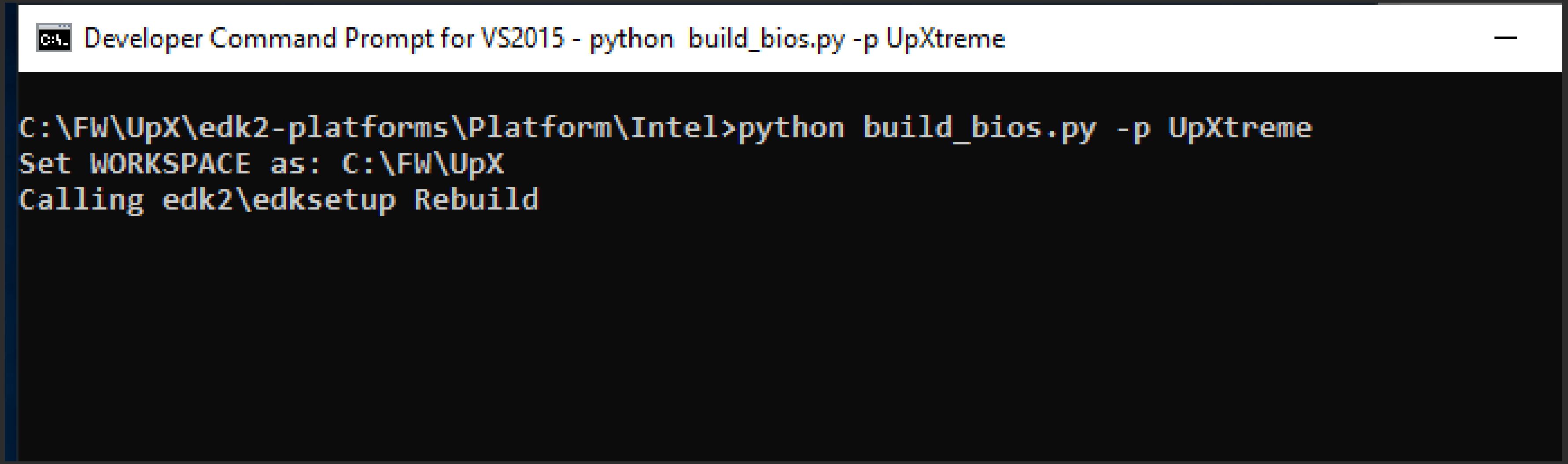
See . . . `WhiskeylakeOpenBoardPkg/UpXtreme/OpenBoardPkg.fdf` line 285

Build with new logo

Invoke the Python Build script for Up Xtreme

```
$> python build_bios.py -p UpXtreme -t GCC5
```

Takes about 2 minutes



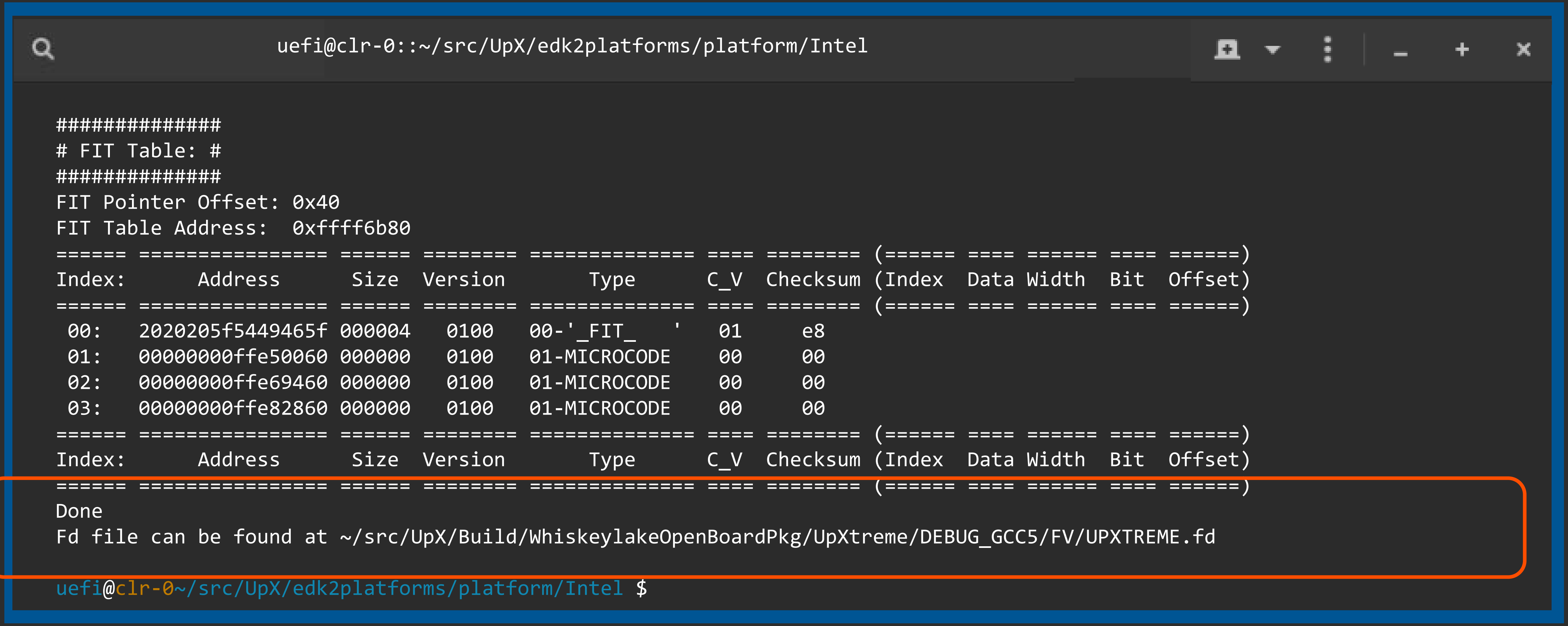
Developer Command Prompt for VS2015 - python build_bios.py -p UpXtreme

```
C:\FW\UpX\edk2-platforms\Platform\Intel>python build_bios.py -p UpXtreme
Set WORKSPACE as: C:\FW\UpX
Calling edk2\edksetup Rebuild
```

Note example screen shots are from windows but same information would be on the terminal screen for Linux

Build Process Completed

Locate the build .fd images



```
##############
# FIT Table: #
##############
FIT Pointer Offset: 0x40
FIT Table Address:  0xffff6b80
====== ================ ====== ======== ============== ==== ======== (====== ==== ====== ==== ======)
Index:      Address      Size  Version       Type      C_V  Checksum (Index  Data Width  Bit  Offset)
====== ================ ====== ======== ============== ==== ======== (====== ==== ====== ==== ======)
 00:   2020205f5449465f 000004   0100   00-'_FIT_   '   01      e8
 01:   00000000ffe50060 000000   0100   01-MICROCODE    00      00
 02:   00000000ffe69460 000000   0100   01-MICROCODE    00      00
 03:   00000000ffe82860 000000   0100   01-MICROCODE    00      00
====== ================ ====== ======== ============== ==== ======== (====== ==== ====== ==== ======)
Index:      Address      Size  Version       Type      C_V  Checksum (Index  Data Width  Bit  Offset)
====== ================ ====== ======== ============== ==== ======== (====== ==== ====== ==== ======)
Done
Fd file can be found at ~/src/UpX/Build/WhiskeylakeOpenBoardPkg/UpXtreme/DEBUG_GCC5/FV/UPXTREME.fd

uefi@clr-0~/src/UpX/edk2platforms/platform/Intel $
```

The script displays the location of the final .fd files

SUMMARY

- Download Minplatform Using Git Bash
- Build a EDK II Platform using Up Xtreme Aaeon board

Questions?

Return to Main Training Page

Return to Training Table of contents for next presentation link

tianocore

www.tianocore.org

ACKNOWLEDGEMENTS

Redistribution and use in source (original document form) and 'compiled‘ forms (converted to PDF, epub, HTML and other formats) with or without modification, are permitted provided that the following conditions are met:

Redistributions of source code (original document form) must retain the above copyright notice, this list of conditions and the following disclaimer as the first lines of this file unmodified.

Redistributions in compiled form (transformed to other DTDs, converted to PDF, epub, HTML and other formats) must reproduce the above copyright notice, this list of conditions and the following disclaimer in the documentation and/or other materials provided with the distribution.

THIS DOCUMENTATION IS PROVIDED BY TIANOCORE PROJECT "AS IS" AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL TIANOCORE PROJECT BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS DOCUMENTATION, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.