

UEFI & EDK II Training

UEFI Driver Wizard Lab – Windows

See also [Lab Guide.md](#) for Copy & Paste examples in labs

tianocore.org

Lesson Objective

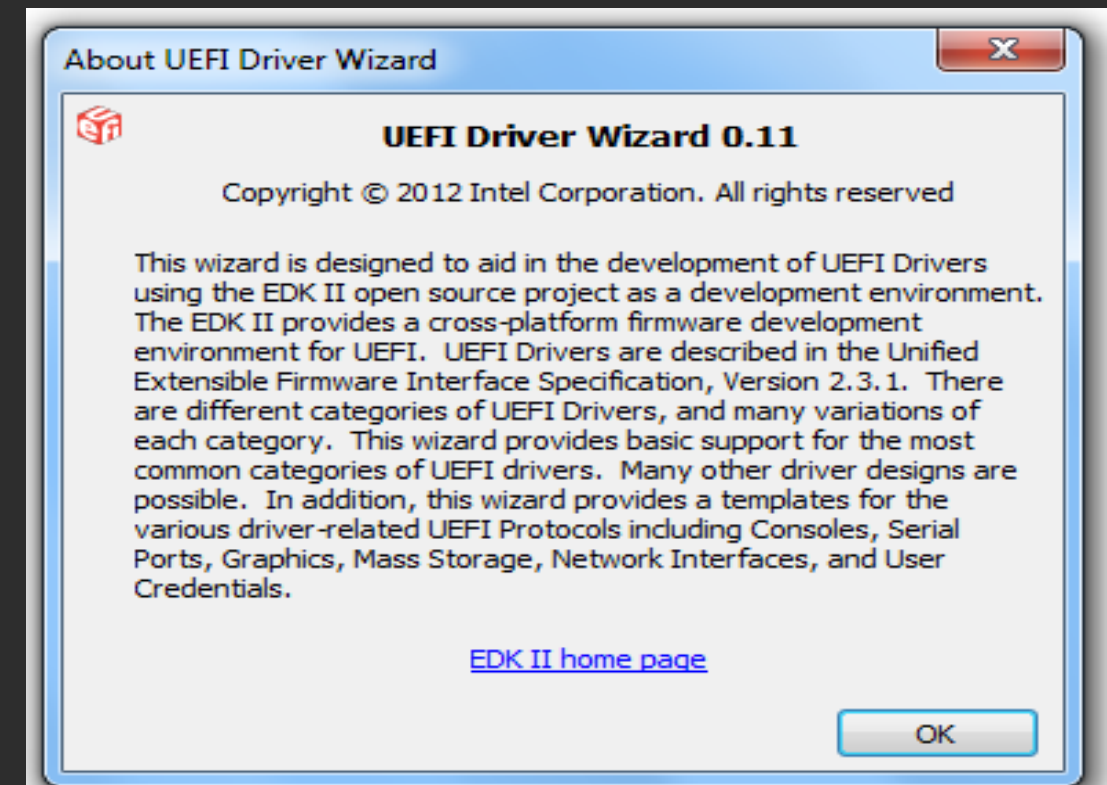
- ✿ Setup the UEFI Driver Wizard
- ✿ Create a UEFI Driver Template

UEFI DRIVER WIZARD

Creating a Template UEFI Driver with the UEFI Driver Wizard

UEFI Driver Wizard Overview

- ✓ Open source tool
- ✓ Based on *Driver Writer's Guide for UEFI 2.3.1* content
- ✓ Intel engineers contributed
- ✓ Located on www.TianoCore.org



Installing UEFI Driver Wizard

Requirements and Options

- Workspace must contain BaseTools, MdePkg & MdeModulePkg Packages from [tianocore.org edk2](https://tianocore.org/edk2) for Driver development on Tianocore.org
- Uses previous lab's setup w/ Windows C:\FW\edk2-ws\
- Python* scripts from [Github Link](#) then use instructions from README for Python and wxPython versions to install then run

```
bash$ python launch.py
```

Requirements for Your Driver



Using UEFI Driver Wizard

- UEFI Device Driver
- UEFI Version 2.7 (0x00020046)
`#define EFI_2_70_SYSTEM_TABLE_REVISION ((2<<16) | (70DEC))`
- Unloadable driver
- Support IA32 & x64 CPUs
- Returns component name information
- Byte stream device (i.e. UART / Serial I/O)
- Option to produce HII strings & forms for setup

Template File Contents

Proper UEFI driver entry point

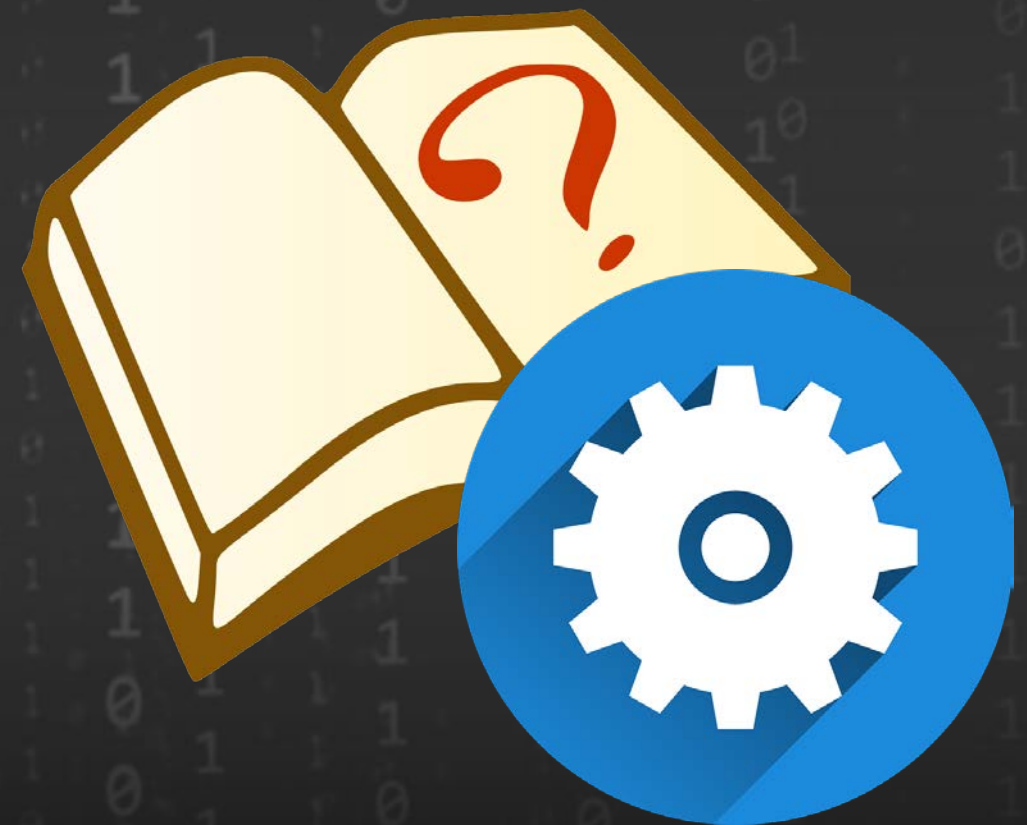
Basic driver libraries/headers

Skeletons for common driver functions

Error values until ported
EFI_UNSUPPORTED, EFI_DEVICE_ERROR

Lab 1: Create a UEFI Driver with the UEFI Driver Wizard

- In this lab, you'll create a new UEFI driver using the UEFI Driver Wizard.
- This will create a set of "c" code files to be used as a template UEFI Driver used in the subsequent driver labs



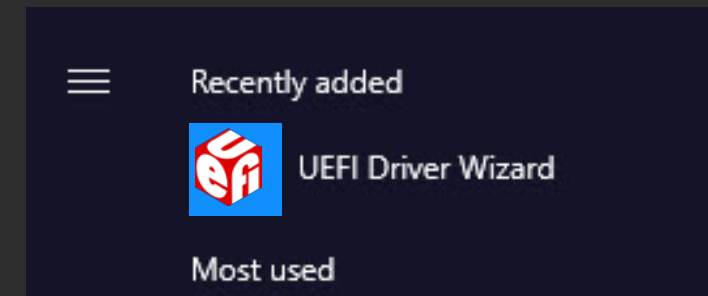
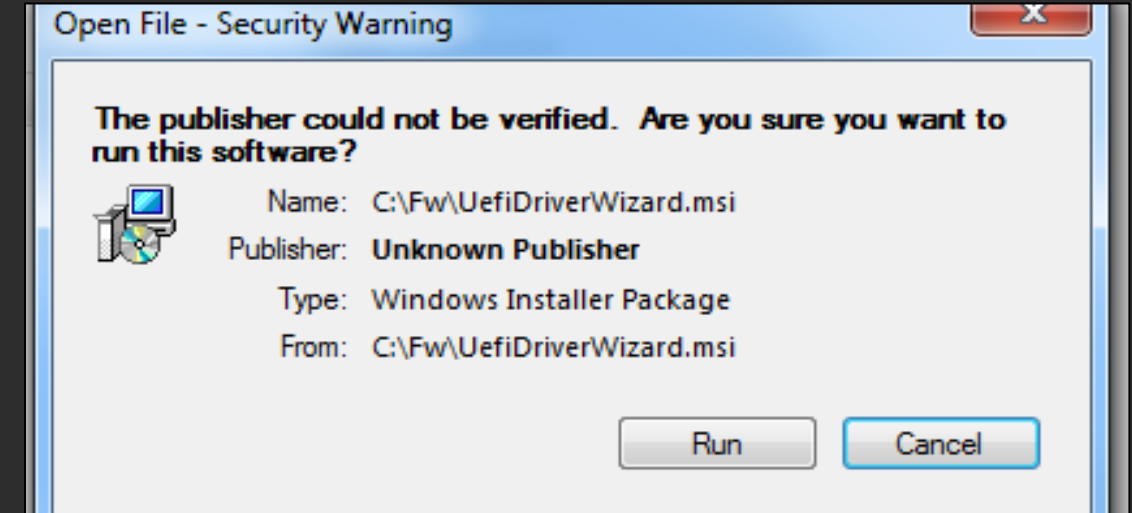
Lab 1: Install UEFI Driver Wizard

First setup for Building EDK II for Emulator, See [Lab Setup](#)

Install UEFI Driver Wizard

1. **Open and Run**
/FW/DriverWizard/UefiDriverWizard.msi
2. **Click** through “Next” until install finishes

Open the UEFI Driver Wizard

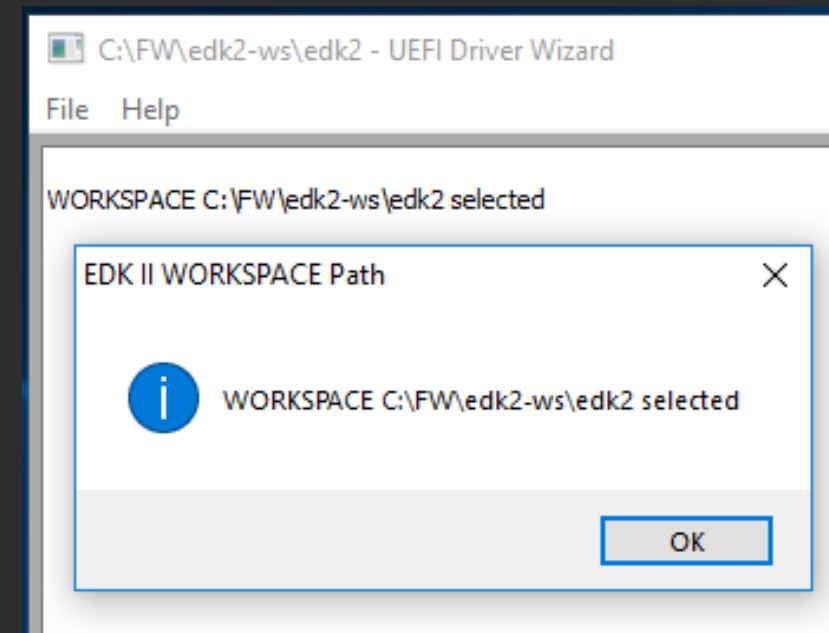
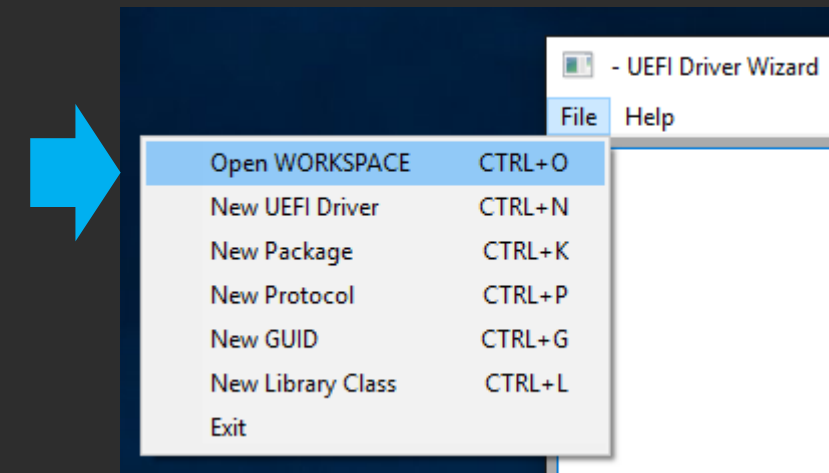


Lab 1: UefiDriverWizard -Select Work Space

Click on File and Select
“Open WORKSPACE”
Or
Control+O

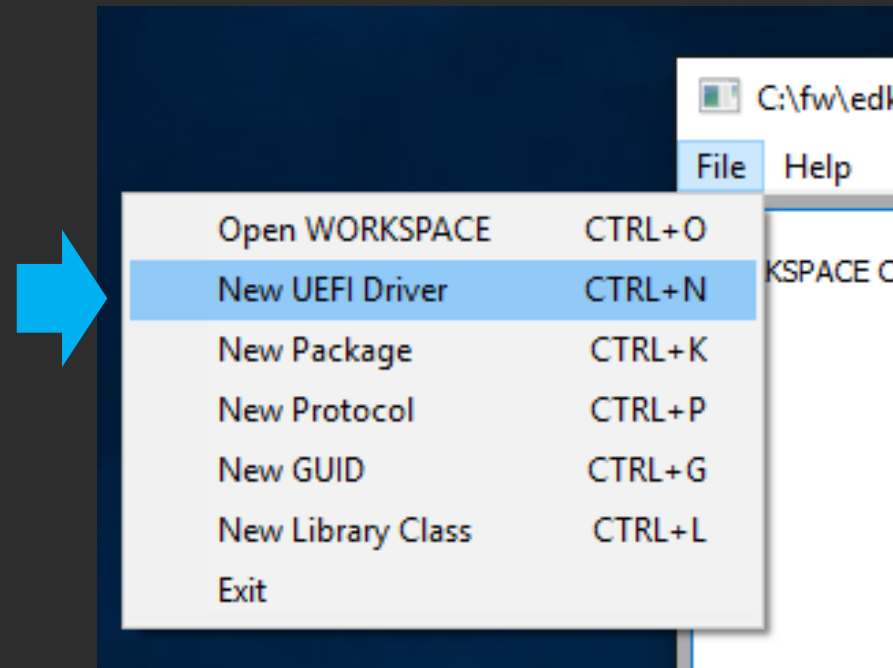
Browse to C:/FW/edk2-ws/edk2
Select “OK”
Should say
“WORKSPACE C:\FW\edk2-ws\edk2 selected”

Note: the environment for EDK II must be setup with edksetup.bat



Lab 1: Create a New UEFI Driver

Control+N – to Open Menu



New UEFI Driver

UEFI Driver Path: C:\FW\edk2-ws\edk2\ Browse

UEFI Driver Name:

UEFI Driver Version:

UEFI Driver GUID: f2d8664f-bebf-11e9-9980-4485002a4585 Generate GUID

UEFI Driver Type:

- ☒ UEFI Driver Model Device Driver
- ☐ UEFI Driver Model Bus Driver
- ☐ UEFI Driver Model Hybrid Driver
- ☐ Root Bridge Driver
- ☐ Service Driver
- ☐ Initializing Driver

Driver Binding Version: 0x00000000

Optional Features Common to all UEFI Driver Types:

- ☐ Unloadable
- ☐ Driver Supported EFI Version Protocol
- ☐ HII Packages for Strings, Fonts, or Images
- ☐ Service Binding Protocol

UEFI Specification Version: 0x0002001E

CPU Architectures:

- ☒ All CPU Architectures
- ☐ IA32
- ☐ X64
- ☐ IPF
- ☐ EBC
- ☐ ARM

<< Prev Next >> Finish Cancel

Lab 1: New UEFI Driver Menu

- UEFI Driver Path” – Type: “MyWizardDriver”

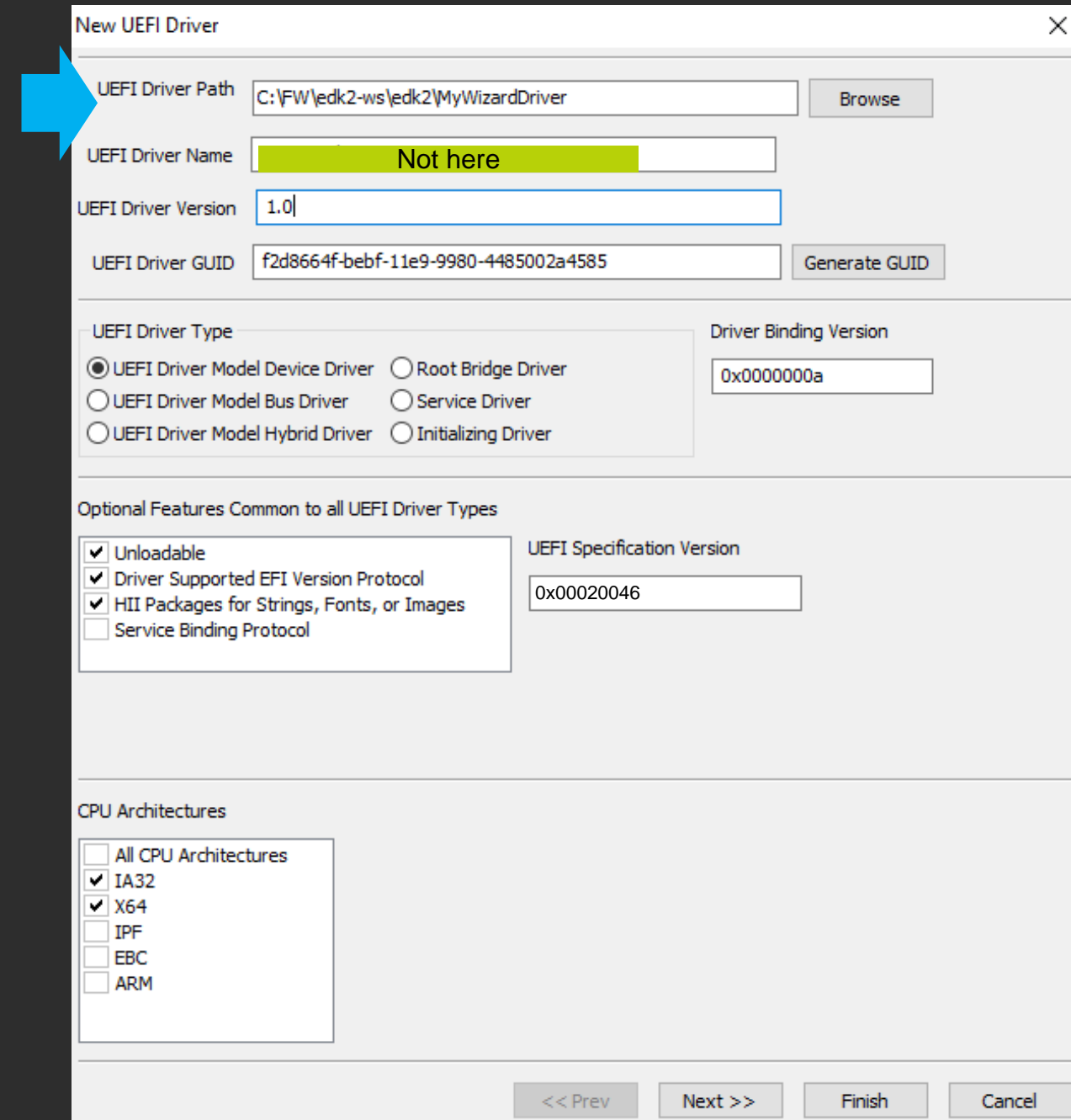
Note: “UEFI Driver Name” is filled in.

- **Ensure** all the forms, radio buttons, and boxes are filled in and **selected exactly** like the image to the right. (except GUID)

- **Note:** A new, specific driver GUID will populate, so it will be different than this image

Click

Next >>



New UEFI Driver

UEFI Driver Path: C:\FW\edk2-ws\edk2\MyWizardDriver [Browse]

UEFI Driver Name: Not here

UEFI Driver Version: 1.0

UEFI Driver GUID: f2d8664f-bebf-11e9-9980-4485002a4585 [Generate GUID]

UEFI Driver Type:

- ☒ UEFI Driver Model Device Driver
- ☐ Root Bridge Driver
- ☐ UEFI Driver Model Bus Driver
- ☐ Service Driver
- ☐ UEFI Driver Model Hybrid Driver
- ☐ Initializing Driver

Driver Binding Version: 0x0000000a

Optional Features Common to all UEFI Driver Types:

- ☒ Unloadable
- ☒ Driver Supported EFI Version Protocol
- ☒ HII Packages for Strings, Fonts, or Images
- ☐ Service Binding Protocol

UEFI Specification Version: 0x00020046

CPU Architectures:

- ☐ All CPU Architectures
- ☒ IA32
- ☒ X64
- ☐ IPF
- ☐ EBC
- ☐ ARM

<< Prev Next >> Finish Cancel

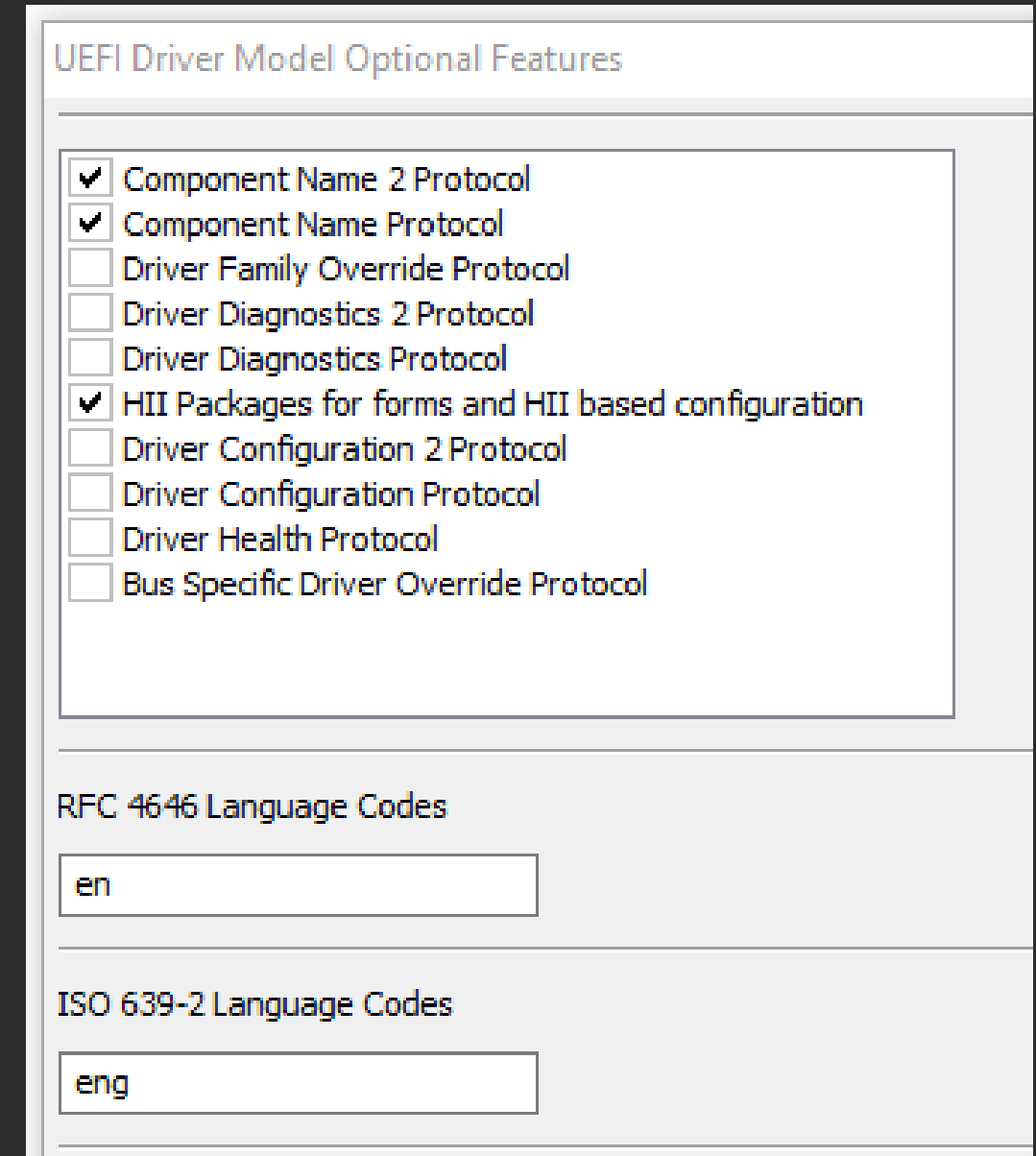
Lab 1: UEFI Driver Model Optional Features

Ensure all the forms, radio buttons, and boxes are filled in and **selected *exactly*** like the image to the right.

- ✓ "Component Name 2 Protocol"
- ✓ "Component Name Protocol"
- ✓ "HII Packages for Forms . . ."

Click

Next >>



The screenshot shows a window titled "UEFI Driver Model Optional Features". It contains a list of optional features with checkboxes. The following features are checked: "Component Name 2 Protocol", "Component Name Protocol", and "HII Packages for forms and HII based configuration". The other features are unchecked: "Driver Family Override Protocol", "Driver Diagnostics 2 Protocol", "Driver Diagnostics Protocol", "Driver Configuration 2 Protocol", "Driver Configuration Protocol", "Driver Health Protocol", and "Bus Specific Driver Override Protocol". Below the list, there are two sections for language codes. The "RFC 4646 Language Codes" section has a text box containing "en". The "ISO 639-2 Language Codes" section has a text box containing "eng".

| Feature | Selected |
|--|----------|
| Component Name 2 Protocol | ✓ |
| Component Name Protocol | ✓ |
| Driver Family Override Protocol | |
| Driver Diagnostics 2 Protocol | |
| Driver Diagnostics Protocol | |
| HII Packages for forms and HII based configuration | ✓ |
| Driver Configuration 2 Protocol | |
| Driver Configuration Protocol | |
| Driver Health Protocol | |
| Bus Specific Driver Override Protocol | |

RFC 4646 Language Codes

en

ISO 639-2 Language Codes

eng

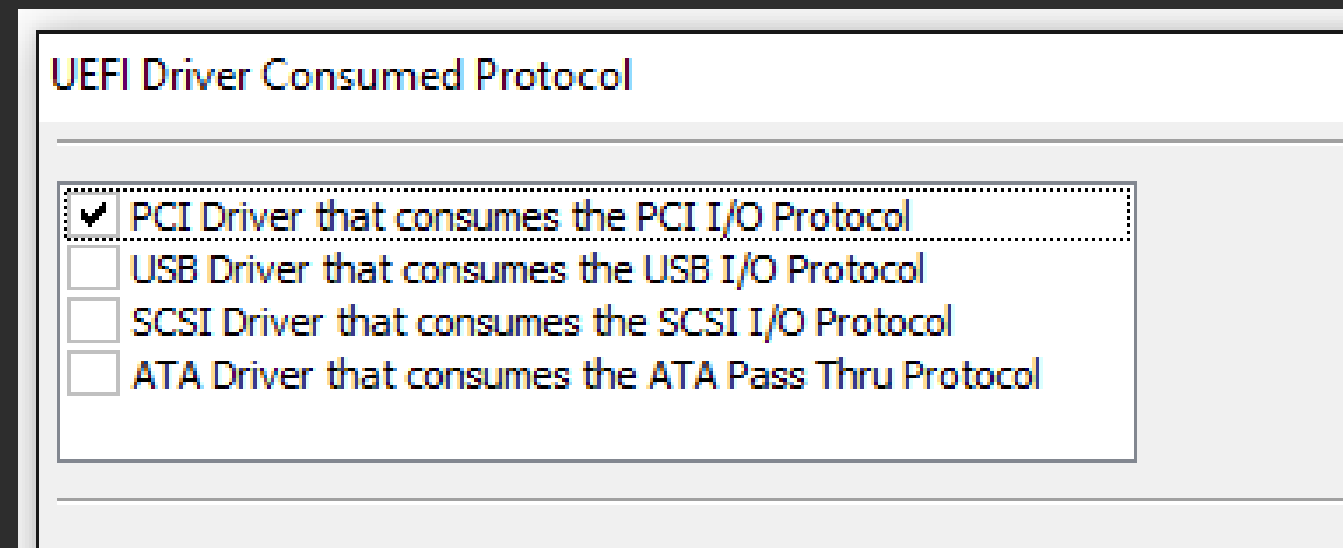
Lab 1: UEFI Driver Consumed Protocol

Select

✓ “PCI Driver that consumes the PCI I/O Protocol”

Click

Next >>



UEFI Driver Consumed Protocol

- ☒ PCI Driver that consumes the PCI I/O Protocol
- ☐ USB Driver that consumes the USB I/O Protocol
- ☐ SCSI Driver that consumes the SCSI I/O Protocol
- ☐ ATA Driver that consumes the ATA Pass Thru Protocol

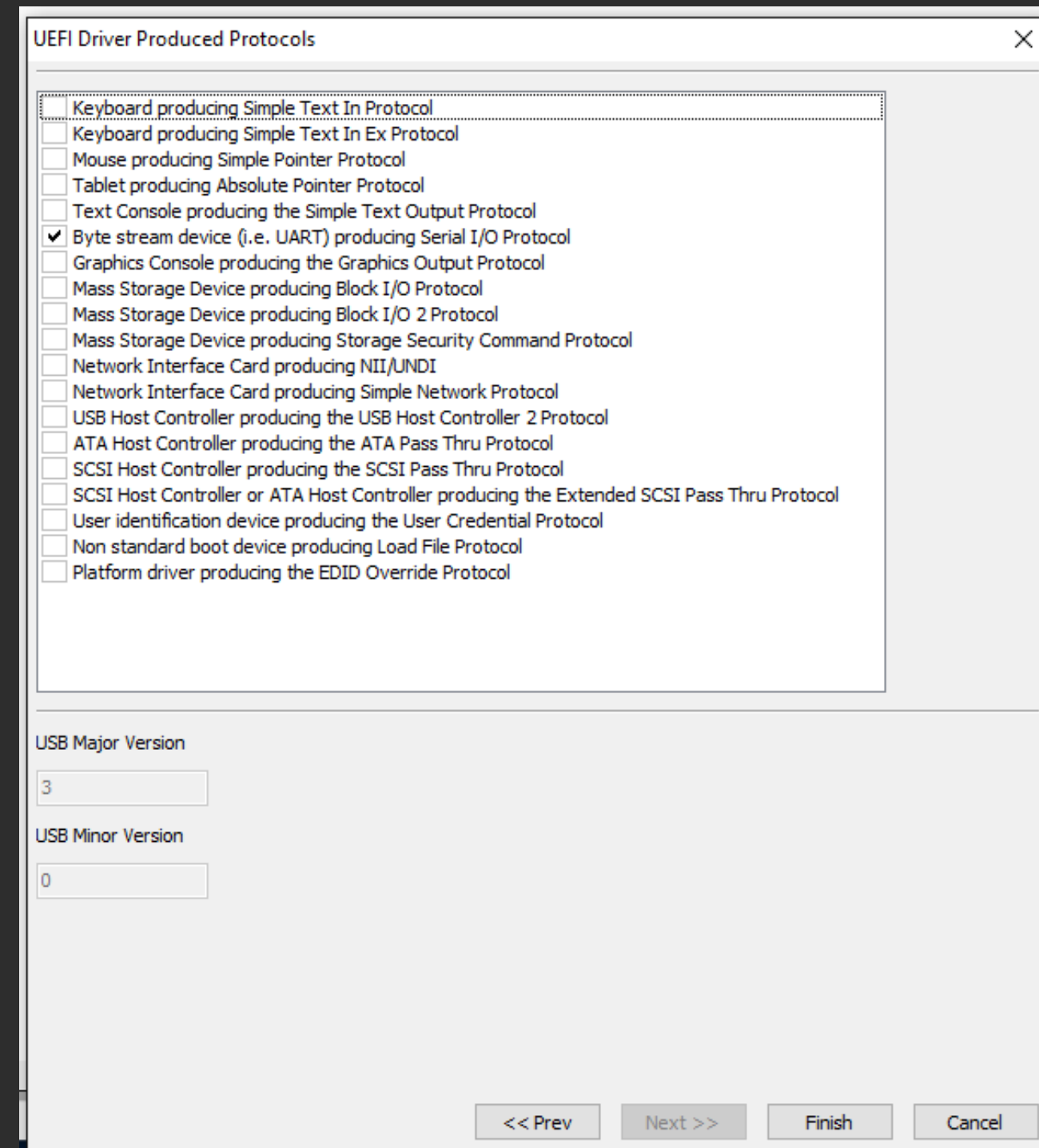
Lab1: UEFI Driver Produced Protocols

Select

- ✓ "Byte stream device (i.e.UART) producing Serial I/O Protocol"

Click

Finish



The screenshot shows a window titled "UEFI Driver Produced Protocols" with a list of protocols and their checkboxes. The "Byte stream device (i.e. UART) producing Serial I/O Protocol" is checked. Below the list are input fields for "USB Major Version" (3) and "USB Minor Version" (0). At the bottom are navigation buttons: "<< Prev", "Next >>", "Finish", and "Cancel".

| Protocol | Selected |
|--|-------------------------------------|
| Keyboard producing Simple Text In Protocol | <input checked="" type="checkbox"/> |
| Keyboard producing Simple Text In Ex Protocol | <input type="checkbox"/> |
| Mouse producing Simple Pointer Protocol | <input type="checkbox"/> |
| Tablet producing Absolute Pointer Protocol | <input type="checkbox"/> |
| Text Console producing the Simple Text Output Protocol | <input type="checkbox"/> |
| Byte stream device (i.e. UART) producing Serial I/O Protocol | <input checked="" type="checkbox"/> |
| Graphics Console producing the Graphics Output Protocol | <input type="checkbox"/> |
| Mass Storage Device producing Block I/O Protocol | <input type="checkbox"/> |
| Mass Storage Device producing Block I/O 2 Protocol | <input type="checkbox"/> |
| Mass Storage Device producing Storage Security Command Protocol | <input type="checkbox"/> |
| Network Interface Card producing NII/UNDI | <input type="checkbox"/> |
| Network Interface Card producing Simple Network Protocol | <input type="checkbox"/> |
| USB Host Controller producing the USB Host Controller 2 Protocol | <input type="checkbox"/> |
| ATA Host Controller producing the ATA Pass Thru Protocol | <input type="checkbox"/> |
| SCSI Host Controller producing the SCSI Pass Thru Protocol | <input type="checkbox"/> |
| SCSI Host Controller or ATA Host Controller producing the Extended SCSI Pass Thru Protocol | <input type="checkbox"/> |
| User identification device producing the User Credential Protocol | <input type="checkbox"/> |
| Non standard boot device producing Load File Protocol | <input type="checkbox"/> |
| Platform driver producing the EDID Override Protocol | <input type="checkbox"/> |

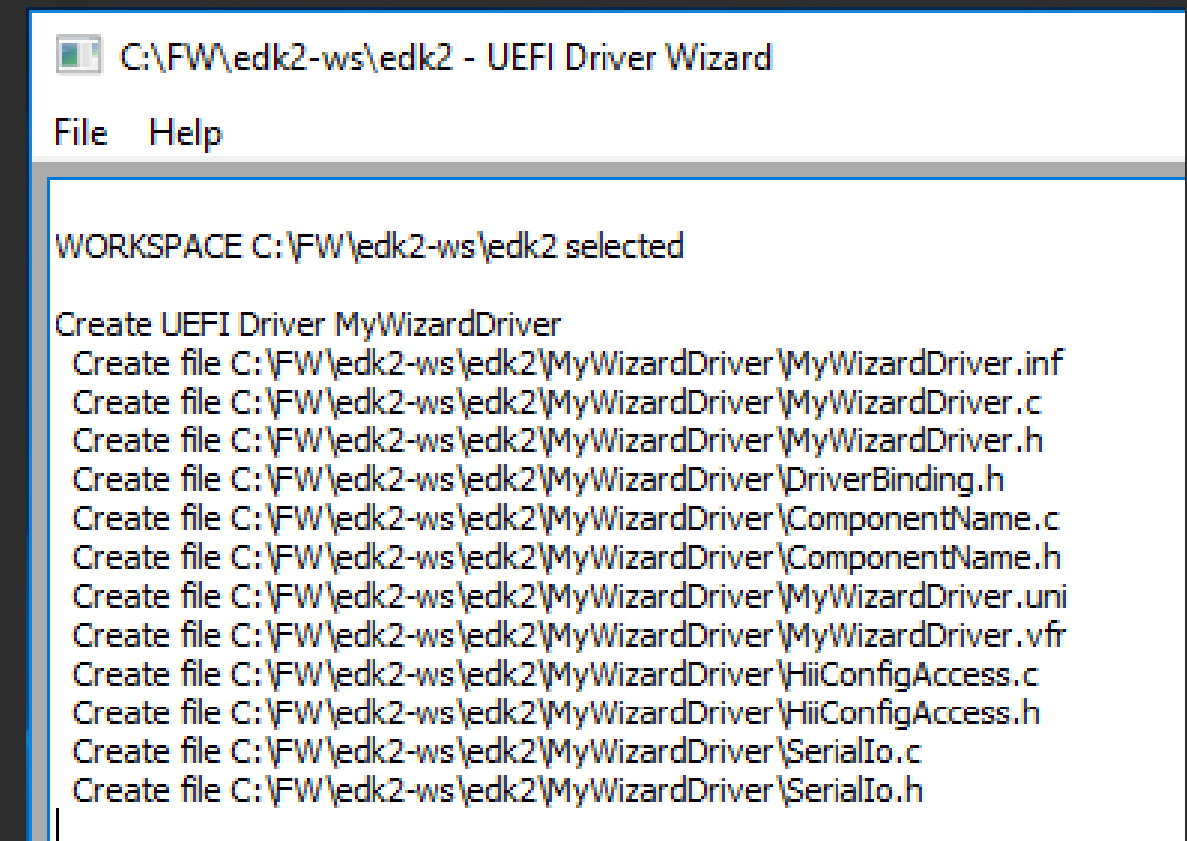
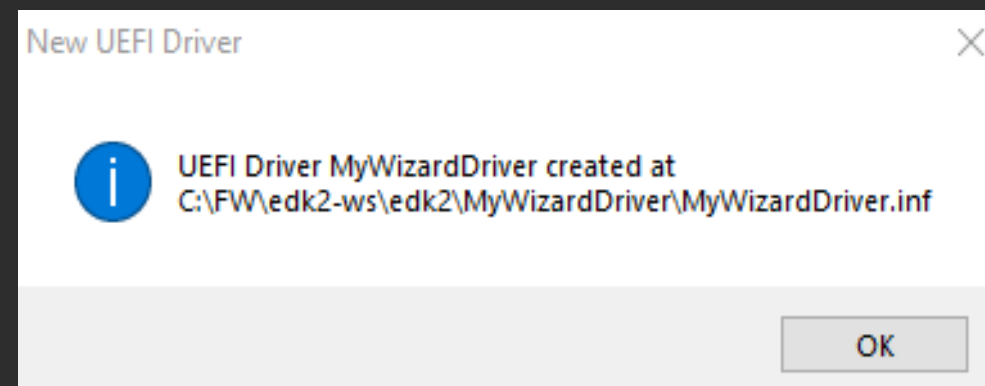
USB Major Version: 3

USB Minor Version: 0

<< Prev Next >> Finish Cancel

Lab 1: UEFI Driver Created

UEFI Driver template created



Close the UEFI Wizard

Summary

- ✿ Setup the UEFI Driver Wizard
- ✿ Create a UEFI Driver Template

Questions?



Return to Main Training Page



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



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.

Copyright (c) 2021=2022, Intel Corporation. All rights reserved.