OPEN POSSIBILITIES.

FSP and MinPlatform for Sapphire Rapids Intel® Xeon® Scalable Processors
FSP and MinPlatform for Sapphire Rapids Intel® Xeon® Scalable Processors

Nate DeSimone, Firmware Engineer, Intel
Isaac Oram, Principal Engineer, Intel

OPEN POSSIBILITIES.
Agenda

• FSP & MinPlatform Overview
• Recent Development Progress
• Plans for Sapphire Rapids
• Project Opportunities
• Call to Action
FSP & MinPlatform Overview

- Intel® FSP is a silicon initialization binary
  - Easily integrable into diverse OSF solutions.

- MinPlatform is a sub-project of TianoCore with 2 primary goals:
  1. Improve the architecture, modularity, and consistency of UEFI Firmware
  2. Run open source EDK II on real systems
OSF on Intel Xeon

• Intel provides industry leading support for diverse OSes (RHEL, Debian, Windows, VMware…) on Xeon.
  o Now… that extends to OSF:

• Intel makes diverse OSF possible with:
  1. Publicly redistributable FSP binaries
  2. MinPlatform as an example OSF that can be referenced/modified

Targeting a best-in-class OSF experience!
Better Together

• FSP + MinPlatform creates an OSF solution that meets the OCP Accepted Checklist requirements.

• FSP & MinPlatform derive from the same codebase and fit together exactly.

FSP + MinPlatform = Fastest TTM for OSF on Xeon
MinPlatform board APIs offer a similar experience to the coreboot mainboard APIs:

<table>
<thead>
<tr>
<th>coreboot</th>
<th>MinPlatform</th>
</tr>
</thead>
<tbody>
<tr>
<td>bootblock_mainboard_init()</td>
<td>BoardInitBeforeMemoryInit()</td>
</tr>
<tr>
<td>mainboard_init()</td>
<td>BoardInitBeforeSiliconInit()</td>
</tr>
<tr>
<td>mainboard_memory_init_params()</td>
<td>SiliconPolicyUpdatePreMem()</td>
</tr>
<tr>
<td>mainboard_silicon_init_params()</td>
<td>SiliconPolicyUpdatePostMem()</td>
</tr>
</tbody>
</table>
Recent Progress

• Ice Lake and Cooper Lake MinPlatform is now open-source

• Includes publicly redistributable FSP binaries

https://github.com/tianocore/edk2-platforms/tree/master/Platform/Intel/WhitleyOpenBoardPkg
Since OCP Tech Week 2020, 5 new boards have been liberated!

<table>
<thead>
<tr>
<th>Machine Name</th>
<th>Chipset</th>
<th>Build Target Name</th>
</tr>
</thead>
<tbody>
<tr>
<td>Intel Wilson City RVP</td>
<td>IceLake-SP</td>
<td>WilsonCityRvp</td>
</tr>
<tr>
<td>Intel Cooper City RVP</td>
<td>Cooper Lake</td>
<td>CooperCityRvp</td>
</tr>
<tr>
<td>Facebook Tioga Pass</td>
<td>Purley-R</td>
<td>BoardTiogaPass</td>
</tr>
<tr>
<td>Intel TGL-U DDR4 RVP</td>
<td>Tiger Lake</td>
<td>TigerlakeURvp</td>
</tr>
<tr>
<td>Acer Aspire VN7-572G</td>
<td>Sky Lake</td>
<td>AspireVn7Dash572G</td>
</tr>
</tbody>
</table>
Recent Progress (Cont.)

- Native support for LinuxBoot is now upstream
- 64-bit handoff via the kernel’s EFI stub boot protocol

https://github.com/tianocore/edk2-platforms/tree/master/Platform/Intel/PurleyOpenBoardPkg/Features/LinuxBoot
Planned LinuxBoot Improvements

• Move from PurleyOpenBoardPkg to a unified LinuxBootFeaturePkg.

• Support kernel 5.7+ arch-agnostic EFI initrd load.  
  https://lore.kernel.org/linux-efi/20200207202637.GA3464906@rani.riverdale.lan/T/#m4a25eb33112fab7a22faa0fd65d4d663209af32f

• LinuxBoot customized/optimized BdsDxe.

• **Open Question to Community:** What should we do about the kernel command line? We currently have:
  ```
  char CmdLine[] = " ";
  ```
Plans for 4\textsuperscript{th} Gen Xeon\textregistered{} Scalable

- Improved support for FSP API mode
- More open-source advanced features for MinPlatform available out-of-box
- Gradual progress on evolving traditional UEFI BIOS towards the lightweight and open MinPlatform design
Project Opportunities

- Unified LinuxBoot Advanced Feature
- Phase Agnostic Serial Logging (PEI/DXE/SMM)
- Binary silicon init policy abstraction
- Universal Payload use
Call to Action

1. Use FSP + MinPlatform to build OSF for your upcoming OCP design.
   • We are happy to help merge new board ports upstream to TianoCore!
   • Its open source – contributions from other silicon vendors are welcome!

2. Collaborate with us on Project Opportunities!
   • LinuxBoot Advanced Feature, Phase Agnostic Logging, Universal Payload, etc.

3. Join the TianoCore Development Mailing list.
   • https://edk2.groups.io/g/devel

4. Check out the MinPlatform training material.
   • https://github.com/tianocore-training/Presentation_FW/blob/main/FW/Presentations/_D_05_EDK_II_Open_Source_MinPlatform_pres.pdf
Thank you!