Arm SystemReady and the UEFI Firmware Ecosystem

UEFI 2021 Virtual Plugfest
January 26, 2021
Dong Wei (Arm)
Samer El-Haj-Mahmoud (Arm)
Dong Wei is an Arm Fellow and is responsible for the Arm SystemReady program and other related standards. He is the Chief Executive of the UEFI Forum and a Board member of PCI-SIG and CXL Consortium.

Samer El-Haj-Mahmoud is a Senior Principal Architect at Arm, working on Arm SystemReady and firmware architecture. He contributes to industry standards such as UEFI, ACPI, CXL, and DMTF Redfish as well as the TianoCore open-source firmware project.
Agenda

• Arm SystemReady
• Arm UEFI Firmware Ecosystem
• Devices Showcase
Arm SystemReady

Vision: “Software Can Just Work on Arm-based Devices”
## Arm SystemReady

### Making all Arm-based infrastructure consistent

<table>
<thead>
<tr>
<th>Hardware Requirements</th>
<th>Firmware Requirements</th>
<th>Certification</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>BSA - Base System Architecture</strong></td>
<td><strong>BBR – Base Boot Requirements</strong></td>
<td><strong>ACS - Architectural Compliance Suites</strong></td>
</tr>
</tbody>
</table>
| • Documents minimal set of CPU and System architecture necessary for an OS to boot and run. Includes aspects such as PCIe integration etc. | • Expand to include common firmware interfaces, but recognize that different software stacks will require different recipes | • WIP, Restructured for SystemReady.  
• Existing ACS v2.5 used for now, with new versions available in the future |
| • Add segment-specific xBSA hardware requirements (e.g. SBSA for servers) | | |

---

For more information, visit: 

https://developer.arm.com/architectures/system-architectures/arm-systemready

www.uefi.org
BBR Interfaces

- PSCI, SMCCC (Common)
- UEFI (for SBBR recipe)
- ACPI (for SBBR recipe)
- Exceptions (if needed)
- SMBIOS
- DeviceTree (reference DT Spec)

BBR Recipes Tailored to Various OSes

SBBR
- Same requirements as current SBBR “Servers” specification
- PSCI, SMCCC, UEFI, ACPI, SMBIOS
- More complete OS support

EBBR
- PSCI, SMCCC, UEFI

LBBR
- PSCI, SMCCC, LinuxBoot, ACPI, SMBIOS

EBBR Spec

- UEFI Requirements for embedded systems
- Subset of SBBR. BBR spec refers to EBBR spec as needed (for EBBR recipe)
- Includes reduced UEFI requirements for embedded systems
- Open community spec development
  [https://github.com/ARM-software/ebbr](https://github.com/ARM-software/ebbr)
- Join the discussion on Linaro Boot Architecture mailing list

https://developer.arm.com/documentation/den0044/latest
www.uefi.org
BBSR (Base Boot Security Requirements)

• Additional requirements for security interfaces for UEFI (SBBR or EBBR) based systems
  – UEFI Authenticated Variables
  – UEFI Secure Boot
  – UEFI secure firmware update using Capsule Updates
  – TPMs and Measured Boot

• Additional SystemReady “Security Option” Certification

https://developer.arm.com/documentation/den0107/latest
System Firmware Landscape

Vertical

Horizontal

Vertical

trustedfirmware.org

Edge Node

Cloud & Datacenters

www.uefi.org
## One program, Multiple Bands

<table>
<thead>
<tr>
<th>Certification</th>
<th>Description</th>
<th>Specifications</th>
</tr>
</thead>
<tbody>
<tr>
<td>SystemReady SR</td>
<td>ServerReady</td>
<td>BSA</td>
</tr>
<tr>
<td>SystemReady LS</td>
<td>LinuxBoot Server Ready</td>
<td>BSA</td>
</tr>
<tr>
<td>SystemReady ES</td>
<td>Embedded Server Ready</td>
<td>BSA</td>
</tr>
<tr>
<td>SystemReady IR</td>
<td>IOT Ready</td>
<td>BSA</td>
</tr>
<tr>
<td>Security</td>
<td>Security Option</td>
<td>BSA</td>
</tr>
</tbody>
</table>

https://developer.arm.com/architectures/system-architectures/arm-systemready
One program, Multiple Bands

arm SystemReady

SR
CERTIFIED V2.0

arm SystemReady

LS
CERTIFIED V1.0

arm SystemReady

ES
CERTIFIED V1.0

arm SystemReady

IR
CERTIFIED V1.0

arm SystemReady

CERTIFIED V1.0

www.uefi.org
Firmware Interfaces for SR, ES, IR

- OSes
  - SR (SBBR)
  - ES (SBBR)
  - IR (EBBR + DT)

<table>
<thead>
<tr>
<th>DT</th>
<th>UEFI</th>
<th>ACPI</th>
<th>SMBIOS</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>U-boot</td>
<td>EDK2</td>
<td></td>
</tr>
</tbody>
</table>

PSCI, SMCCC

Trusted Firmware

www.uefi.org
<table>
<thead>
<tr>
<th></th>
<th>LS (LinuxBoot Server Ready)</th>
<th>IR (IoT Ready)</th>
<th>ES (Embedded Server Ready)</th>
<th>SR (ServerReady)</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Firmware Spec</strong></td>
<td>ACPI + SMBIOS</td>
<td>UEFI + Devicetree</td>
<td>UEFI + ACPI + SMBIOS</td>
<td>UEFI + ACPI + SMBIOS</td>
</tr>
<tr>
<td><strong>Platform Hardware</strong></td>
<td>64-bit Arm</td>
<td>32-bit/64-bit Arm</td>
<td>64-bit Arm</td>
<td>64-bit Arm</td>
</tr>
<tr>
<td><strong>OS/Hypervisor</strong></td>
<td>Linux</td>
<td>Linux, etc.</td>
<td>Windows IoT Enterprise, VMware ESXi, RHEL, SLES, Ubuntu, CentOS, Fedora, openSUSE, Debian, FreeBSD, NetBSD</td>
<td>Generic, off-the-shelf</td>
</tr>
<tr>
<td><strong>OS Distro (examples)</strong></td>
<td>Linux</td>
<td>Fedora, openSUSE, Ubuntu, Debian</td>
<td>Under investigation: OpenWRT, QNX, VxWorks, Integrity, Yocto, Wind River, Mentor</td>
<td>VMware ESXi, Windows Client/Server, RHEL, SLES, Ubuntu, CentOS, Fedora, openSUSE, Debian, FreeBSD, NetBSD</td>
</tr>
<tr>
<td><strong>Hardware Compliance Levels</strong></td>
<td>BSA+SBSA</td>
<td>BSA</td>
<td>BSA</td>
<td>BSA+SBSA</td>
</tr>
<tr>
<td><strong>Levels</strong></td>
<td>Levels 3 through 6</td>
<td>+ No BSA requirements for 32-bit + waivers for existing HW initially</td>
<td>+ waivers for existing HW initially</td>
<td>Levels 3 through 6</td>
</tr>
<tr>
<td><strong>BBR Recipe</strong></td>
<td>LBBR</td>
<td>EBBR</td>
<td>SBBR</td>
<td>SBBR</td>
</tr>
<tr>
<td><strong>Certification</strong></td>
<td>Arm SystemReady LS + System Compatibility List</td>
<td>Arm SystemReady IR + System Certification List</td>
<td>Arm SystemReady ES + System Certification List</td>
<td>Arm SystemReady SR + System Certification List</td>
</tr>
</tbody>
</table>

Can support UEFI SecureBoot and Secure Firmware Update via UEFI Capsule Service across (BBSR)
Architectural Compliance Suite (ACS)

ACS for SystemReady SR
ACS v3.0 available
Tests for SBBR + SBSA compliance

ACS for SystemReady ES and IR
ACS Development WIP (ETA: Q2 CY2021)
Use ACS v2.5 w/ SBSA Level 3 to certify ES now

https://github.com/arm-software/arm-enterprise-acs

https://github.com/ARM-software/arm-systemready
BBSR ACS

• Extend ACS test suites to cover BBSR security requirements
  – **UEFI Authenticated Variables**: Leverage existing [SCT](#) and [FWTS](#) test cases
  – **UEFI Secure Boot**: New [SCT](#) and manual tests to ensure correct behavior of LoadImage() and SetVariable() with PK/KEK/db/dbx/SecureBoot/SetupMode
  – **Capsule Updates**: New manual and automated tests, leveraging UEFI tool [CapsuleApp](#), Linux [fwupdmgr](#), [FWTS esrtdump](#)
  – **TPMs and Measured Boot**: Test TCG2 UEFI Protocol ([SCT](#)), and Linux TPM2 support (using [tpm2-tools](#), and [FWTS tpmevlogdump](#))
Arm Aarch64 UEFI Driver

• Arm based BBR compliant systems require UEFI drivers to be in UEFI AArch64 native format
• Some ecosystem partners already providing AArch64 binaries
• Call to more to make available for support on Arm SystemReady compliant systems

Arm UEFI Firmware Ecosystem
Arm and TianoCore

- Open-source community project with implementations of UEFI standards: UEFI, PI, ACPI, SMBIOS, UEFI Shell, etc.
  - Including Arm SBBR specification
- Growing Arm community (maintainers, contributors)
  - Complete and partial Arm64 platforms, silicon drivers, libraries, and support code
- [https://github.com/tianocore/](https://github.com/tianocore/) : edk2, edk2-platforms, edk2-non-osi, uefi-sct (test suite)
Arm and U-Boot

• “Universal Bootloader” open-source firmware, with support for multiple architectures (including Arm/Arm64)
  – https://github.com/u-boot/u-boot
• Portable, easy to port/debug. Many (100s) boards up-streamed.
• Suitable for embedded / edge devices (predominantly vertically integrated ecosystem)
U-Boot and UEFI

- U-Boot implements a **UEFI layer** that follows the **EBBR specification**, allowing standard OS bootloader (like GRUB) to load and boot a standard OS
- UEFI compliance testing using UEFI SCT (**Results**) and FWTS (**Results**) show very good progress towards complete EBBR compliance
  - Most boot and runtime services, some UEFI protocols
  - Support for booting UEFI Shell and Linux standard UEFI boot loaders (Grub, etc.)
- UEFI Secure Boot and secure Capsule Updates has been recently added to U-Boot
- Reference presentation in **OSFC 2020 by Heinrich Schuchardt**
Arm and LinuxBoot

- LinuxBoot is an alternative firmware stack (used by hyperscale datacenters) that relies on the Linux kernel as the Normal World firmware component.
- Re-uses existing Linux drivers code (without the need to write DXE/UEFI drivers)
- On Arm64 systems, LinuxBoot could be loaded directly from TF-A
- [https://linuxboot.org/](https://linuxboot.org/)
- [https://github.com/linuxboot/linuxboot](https://github.com/linuxboot/linuxboot)
- Google/Facebook leading ongoing work to implement UEFI ABI on top of LinuxBoot.
  - Join the discussions on [OSFC slack server #efi-boot-support channel](https://osfc.slack.com/archives/efi-boot-support)
  - Current proposal relies on UefiPayloadPkg from EDK2
  - Alternative is to implement UEFI ABI directly in Linux, just like the U-Boot approach
SystemReady Devices Showcase
Ampere Altra Mt Jade

- Arm SystemReady SR v2.0 certified
- Ampere Computing Altra Mt Jade Dual Socket Rack Server
- Choice for evaluating benefit of Arm compute in enterprise server roles. Cloud native, high performance scalable CPU
- Firmware options both open-source and commercial
  - UEFI EDK2 (upstreaming patches under review in edk2-devel)
  - OpenBMC FW (upstreaming patches under review)
  - LinuxBoot FW
- UEFI Firmware upstreaming to TianoCore WIP

https://amperecomputing.com/altra/
Raspberry Pi 4 Model B

- **Arm SystemReady ES v1.0 certified**
- Choice for evaluating benefit of Arm in an IoT, edge gateway, or low-end developer box roles
- Open-source firmware community project (leadership from Arm, VMware, Akeo Consulting, and others in the developer community)
- Open-source community:
  - [UEFI project on Github](https://github.com/uefi)
  - [UEFI EDK2 FW (upstream)](https://github.com/uefi)
  - [TF-A FW (upstream)](https://github.com/tf-arch)
  - [Discord community](https://discord.gg/UEFI)
- Porting to other flavors (CM4, RPi 400) TBD/WIP
- Reference: [UEFI Forum Webinar](https://www.uefi.org/events) (by Arm and VMware)

Reference: [https://rpi4-uefi.dev/](https://rpi4-uefi.dev/)
NXP Layerscape LX 2160A RDB

- Arm SystemReady ES v1.0 certified
- Choice for evaluating benefit of Arm in mobile edge compute, edge gateway, embedded/edge server, NFV, 5G, switching, ...
- UEFI EDK2 FW (upstream)
- UEFI EDK2 FW (NXP repo)
- TF-A FW:
- UEFI Firmware upstreaming to TianoCore WIP

https://www.nxp.com/design/qoriq-developer-resources/layerscape-lx2160a-reference-design-board:LX2160A-RDB
Solidrun HoneyComb LX2K

- Arm SystemReady Certification **In Progress**
- Based on NXP Layerscape LX2160A
- Choice for evaluating benefit of Arm in a Micro-server, Workstation, or Edge gateway role.
- UEFI EDK2 FW
- TF-A FW
- UEFI FW Build script
- Discord community
- UEFI FW upstreaming WIP

NXP LS1046A FRWY / RDB

• Arm SystemReady Certification **In Progress**
• Choice for evaluating benefit of Arm in high performance IoT, edge gateway, enterprise access point, etc...
• **UEFI EDK2 FW (upstream)**
• **UEFI EDK2 FW**
• **TF-A FW**
• UEFI Firmware upstreaming to TianoCore WIP

https://www.nxp.com/design/qoriq-developer-resources/ls1046a-freeway-board:FRWY-LS1046A
SBDA QEMU

- Virtualization environment for Armv8-A, with support for Arm SBDA specifications
  - Available as “sbsa-ref” machine
  - Supports SBDA HW such as GICv3, generic timer, watchdog, etc..
- Choice as an environment for developing firmware and testing operating systems and compliance testing
- Linaro working on completing SBDA and SBBR support and testing compliance with the ACS test suite
- Upstreamed to:
  - QEMU
  - UEFI EDK2 FW
  - TF-A FW
- Testing results: sbsa-acs and UEFI SCT tests
Marvell Octeon TX2 CN913x

- Work done by SemiHalf to for UEFI+ACPI support in EDK2
  - Firmware already available upstream
  - Boots most standard distros (Linux, ESXi, BSDs).
  - Testing with ACS test suites for more complete BSA+SBBR compliance

- EDK2 FW (upstream)
- TF-A FW (upstream)
- Reference presentation in OSFC 2020 by Marcin Wojtas
Questions?
More Questions?

Following today’s webinar, join the live, interactive WebEx Q&A for the opportunity to chat with the presenter

Visit this link to attend: [http://bit.ly/3pjpf0O](http://bit.ly/3pjpf0O)
Meeting number (access code): 126 003 8932
Meeting password: UEFIForum (83343678 from phones and video systems)
Thanks for attending the UEFI 2021 Virtual Plugfest

For more information on UEFI Forum and UEFI Specifications, visit http://www.uefi.org