Arm64 Linux Kernel architecture update

Matteo Carlini
Director, Software Technology Management
Arm – Open Source Software
Table of Arm Architecture features vs Kernel Versions

Check out the following table of Arm architecture features against the Linux Kernel version they got merged in.

<table>
<thead>
<tr>
<th>Feature</th>
<th>Kernel Version</th>
<th>Notes</th>
</tr>
</thead>
<tbody>
<tr>
<td>ARMv8.1 Features</td>
<td></td>
<td></td>
</tr>
<tr>
<td>ARMv8.1-LSE</td>
<td>4.3</td>
<td>Kernel atomics use and HWCAP</td>
</tr>
<tr>
<td>ARMv8.1-RODM</td>
<td>4.11</td>
<td>HWCAP</td>
</tr>
<tr>
<td>ARMv8.1-HPD</td>
<td>N/A</td>
<td></td>
</tr>
<tr>
<td>ARMv8.1-VHE</td>
<td>4.6&lt;br&gt;4.17 - KVM</td>
<td>Virtualization Host Extensions</td>
</tr>
<tr>
<td>ARMv8.1-TTHM</td>
<td>4.3&lt;br&gt;4.7 - KVM</td>
<td>Access flag &amp; Dirty Blt Management (ARM64_HW_AFDBM)</td>
</tr>
<tr>
<td>ARMv8.1-PAN</td>
<td>4.3</td>
<td></td>
</tr>
</tbody>
</table>
A-class architecture kernel enablement recap

- v8.0
  - Enablement complete
  - Enablement ongoing
  - Enablement TBD
  - N/A – no Kernel impact

- v8.1
  - LSE
  - PMU
  - RDMA
  - VMID16
  - LOR
  - VHE
  - PAN
  - TTHM

- v8.2
  - A84ISA
  - TT2UX
  - AASHP
  - TTPSHA
  - FHM
  - IESB
  - SM
  - SHA
  - DPDostPr
cd
  - RAS
  - Debug
  - VPIPPT
  - JCPNP
  - LPA
  - LVA

- v8.3
  - CompNum
  - JSconv
  - PMU
  - RCPC
  - CCIDX
  - NV
  - TTCNP
  - UAO
  - ATS1E1
  - LP16

- v8.4
  - PAUTH
  - Debug
  - Trade
  - S-EL2
  - TTST
  - RAS
  - DFE
  - TTRem

- v8.5
  - EVT
  - G TG
  - SpecRest
  - PredInv
  - BT
  - MemTag
  - CMODX
  - FRINT
  - CondM

- v8.6
  - MTPMU
  - ECV
  - CTSS
  - FGT
  - AMU
  - NW
  - MPAM
  - F64MM
  - F32MM
  - F16MM
  - BF16

New Features
- Morello
- SVE2
- TME

Enablement status:
- Enablement complete
- Enablement ongoing
- Enablement TBD
- N/A – no Kernel impact
Armv8.2 enablement leftovers

- **SPE in KVM Guests:**
  - Prototype code available in developer's branch
  - Picked up by KVM arm64 community
  - To be reposted soon

- **RAS:**
  - Support for firmware-first RAS approach completed since **v4.16** (**v5.0** for arm64 SDEI wire-up into APEI)
  - Kernel first RAS approach ongoing by community (Arm Error Source Table (AEST) support – spec in Beta, EAC soon)

- **Perf tool improvements**
  - Support for SPE events **ongoing**
Armv8.3 & Armv8.6 Pointer Authentication

- Armv8.3 Pointer Authentication:
  - v5.0: User-space support enablement
  - v5.1: ptrace regsets for Pointer Authentication key management
  - v5.2: KVM support
  - v5.7 (queued): In-kernel support for function return address protection
  - Crash-utility (process kernel crash dumps) ongoing

- Armv8.6 Pointer Authentication enhancements (on-list):
  - Enhanced PAC generation algorithm
  - Generate fault when authenticate instruction fails
Armv8.3 & Armv8.4 Nested Virtualization

- Nested Virtualization extensions:
  - Development carried over by the KVM arm64 community
  - Arm Kernel team to test and review patches
  - Armv8.3 + Armv8.4 support posted on-list (huge v2 patchset – 94 patches!)
  - Includes:
    - Armv8.4-TTL (Translation Table Level): reduce the cost of TLB invalidation
    - Armv8.5-GTG (Guest translation granule size): advertise the supported Stage-2 page size to hypervisors
Armv8.4 MPAM

- Working on enabling multi-arch support to the resctrl interface

- Latest snapshot rebased against 5.6-rc1 (huge patch-set → 130+ patches!)
  - [http://www.linux-arm.org/git?p=linux-jm.git;a=shortlog;h=refs/heads/mpam/snapshot/feb](http://www.linux-arm.org/git?p=linux-jm.git;a=shortlog;h=refs/heads/mpam/snapshot/feb)

- Rework expected to continue throughout all 2020 with arm64 architectural specific enablement happening during 2021

<table>
<thead>
<tr>
<th>MPAM controls</th>
<th>Resctrl interface</th>
</tr>
</thead>
<tbody>
<tr>
<td>Cache portion bitmaps (L2 &amp; L3 cache)</td>
<td>Available</td>
</tr>
<tr>
<td>Cache capacity</td>
<td>Not Available</td>
</tr>
<tr>
<td>Memory bandwidth portion bitmap</td>
<td>Available</td>
</tr>
<tr>
<td>Memory bandwidth min/max/stride</td>
<td>Not available</td>
</tr>
<tr>
<td>Raw priority</td>
<td>Not available</td>
</tr>
</tbody>
</table>
Armv8.5 Branch Target Identification

- User-space support
  - Queued for v5.7

- In-Kernel support
  - Annotation of assembly functions in the kernel → Queued for v5.7
  - Setup guard pages – internal development ongoing, external posting soon
  - -fpatchable-function-entry support working both in GCC10 and in LLVM 10 release branch
Armv8.5 Memory Tagging (MTE)

- User-space enablement:
  - v5.4: arm64 ABI relaxation to allow passing of tagged pointers to kernel syscalls
  - Heap-tagging support ongoing on-list and on developer’s branch
  - Working in conjunction with the GNU Toolchain team to ensure proper Glibc ABI is in place to test kernel support before upstreaming

- In-Kernel support:
  - Core changes (swap & page tables support) for heap tagging ongoing
  - Stack-tagging:
    - Fine-grained stack-tagging (colouring each function) breaks single-image
    - Considerations for future debug feature (pending compiler support)
SVE & Transactional Memory (TME)

- **SVE**
  - SVE enabled (**v4.15** Kernel, **v5.2** KVM Guests)
  - SVE2 exposed via HWCAP in **v5.2**

- **TME**
  - Kernel Enablement patches [on-list](#)
  - Not immediate pull from partners/ecosystem
Upcoming Kernel releases

- **Kernel v5.6**
  - Armv8.5-E0PD (Preventing EL0 access to halves of address maps)
  - Armv8.5-RNG (Random Number Generator): runtime support
  - Armv8.6-BF16 (BFLOAT16 extension) – HWCAP
  - Armv8.6-I8MM (Int8 Matrix Multiplication) – HWCAP
  - GICv4.1 support

- **Kernel v5.7 (queued)**
  - Armv8.3 PAuth: In-kernel support for function return address protection
  - Armv8.4 Activity Monitors extensions (AMU)
  - Armv8.5 PMU 64-bit counters
  - Armv8.5 BTI user-space & new arm64 asm annotations (prep for BTI in-kernel)
  - Armv8.5-RNG (Random Number Generator): boot time support
  - Memory Hot Remove
Arm Specifications support
(https://developer.arm.com/architectures/system-architectures/software-standards)

- PSA FF-A (SPCI)
  - Arm specification reaching EAC by mid April
  - Arm Kernel team working on upstream kernel driver (on a developer’s branch)
- SCMI
  - v5.4 Fast Channels & Reset Management support
  - v5.6 support for multiple device per protocol
  - Notifications support ongoing
  - System Power protocol development ongoing
- SMCCCv1.2 support development ongoing (SOC_ID)
- True Random Number Generator (TRNG) interface
  - Arm specification will be publicly released as Beta in April
  - Standard Kernel – Firmware SMC interfaces → To be planned 2nd half of 2020
- ACPI for CMN-600 PMU: ongoing on developer’s branch
Ladies and gentlemen, now the moment you've all been waiting for, the world-famous...
Morello

https://www.cl.cam.ac.uk/research/security/ctsrd/cheri/cheri-morello.html
Kernel: Morello ABI phased approach

Software component support for the Morello architecture

- Morello is a prototype architecture (not available to license for product development)
- Support for the C64 instruction set and ABI will be maintained in forks and not upstreamed

October 2020

FVP model

Development board

October 2021

Linux Kernel (Morello ABI 128bit syscall interface)

Early limited interface
Reliant on BIONIC library “shim”

Generic Kernel ABI definition
Developed alongside library support

C Library

BIONIC Android C lib
Early “shimmed” syscall implementation
static linkage only

BIONIC and glibc development
glibc enables Distro environments

Toolchain support

LLVM toolchain
ABI compliant primary toolchain and utilities. (+ gas support for Kernel)

GNU/GCC toolchain
Evolving support

https://connect.linaro.org/resources/ltd20/ltd20-110/
Useful resources

- Open Source section on developer.arm.com

- A-Profile architecture
  - https://developer.arm.com/architectures/cpu-architecture/a-profile
  - https://community.arm.com/developer/ip-products/processors/b/processors-ip-blog/posts/arm-architecture-developments-armv8-6-a
Thank you

Accelerating deployment in the Arm Ecosystem