AmpereOne® AC04 and AC04_1 Developer Errata

July 15, 2025

Introduction

This document contains errata that affect designs using the Ampere Computing® AmpereOne® AC04 and AC04_1 devices.

Table 1 describes how to determine the mapping between “Affected SoC Version Name” of an erratum to an explicit ID visible to software.

Table 1: List of AmpereOne AC04 and AC04_1 SoCs

AFFECTED SoC VERSION NAME	SMCCC SOC ID (SOC VERSION | SOC_REVISION)	MIDR_EL1 (PARTNUM | VARIANT | REVISION)
AmpereOne AC04 A0	jep106:0a16:0004 | 0x*****000	0xac4 | 0 | 0
AmpereOne AC04_1 A0	jep106:0a16:0004 | 0x*****100	0xac4 | 0 | 0

Note:

1. SOC ID can be viewed with the appropriate OS software tool or commands. For example, Linux OS distributions support the following console shell command: `cat /sys/bus/soc/devices/soc0/soc_id`
2. MIDR_EL1 can be viewed with the appropriate OS software tool or commands. For example, Linux OS distributions support the following console shell command: `lscpu`

Each erratum includes an overview, a description of the system impact, and a description of possible workaround(s).

Refer to *Table 3, “Errata Summary”, on page 3* for the list of errata.

Errata are organized by the item designator in an alphabetical order. The item designator consists of an acronym for the affected functional unit and a numeric value. Numeric values are assigned to all errata.

Note: *Unless otherwise indicated, the errata listed in this document apply to both the AC04 and AC04_1 devices.*

List of Functional Unit Acronyms

• CPU	AmpereOne AC04 or AmpereOne AC04_1 Processing Element (PE) within the Processor Complex (PCP)
• DEBUG	Debug Access Port (DAP) and other Debug Components
• MCU	Memory Controller Unit
• MESH	Coherent Mesh Network
• PCIe	PCI Express Controller
• SoC	System-on-Chip

Category Definitions

Errata are classified according to system impact and the availability of a workaround.

1. Major impact, no workaround is available. An issue is said to have a major impact if it results in a system crash, a hard failure, an unrecoverable soft failure, significant performance degradation, or the storage of incorrect data.
2. Major impact, workaround is impractical to implement. A substantial risk of encountering the same or additional issues, including performance issues, exist after the workaround is implemented.
3. Major impact, workaround available. Application of the workaround either eliminates the issue, or reduces it to a minor impact issue, or results in significant performance degradation.

4. Minor impact, no workaround is available. Minor impact issues result in slight to moderate performance degradation, or are a functional variance from specification.
5. Minor impact, workaround is available. Minor impact issues result in slight to moderate performance degradation, or are a functional variance from specification.
6. Design enhancement.

List of Abbreviations and Acronyms

Table 2: List of Abbreviations and Acronyms Used in the Document (Sheet 1 of 2)

TERM	DESCRIPTION
1P	Single-Socket Platform
2P	Dual-Socket Platform
ASID	Address Space Identifier
ATB	Advanced Trace Bus
BAR	Base Address Register
BHB	Branch History Buffer
BMC	Baseboard Management Controller
BTB	Branch Target Buffer
CA	Completer Abort
CCM	Core Cluster Module
CE	Corrected Error
CFI	Fault Handling Interrupt Corrected Errors
CPLD	Complex Programmable Logic Device
CTO	Completion Timeout
DC IVAC	Data or Unified Cache Line Invalidate by VA
DC ZVA	Data Cache Zero by VA
DE	Deferred Error
ECC	Error Correcting Code
EL2	Exception Level 2
ELR	Exception Link Register
ERRIIDR	Implementation Identification Register
ETB	Embedded Trace Buffer
ETM	Embedded Trace Macrocell
ETR	Embedded Trace Router

Table 2: List of Abbreviations and Acronyms Used in the Document (Sheet 2 of 2)

TERM	DESCRIPTION
FAR	Fault Address Register
FHI	Fault Handling Interrupt
GIC	Generic Interrupt Controller
HPFAR	Hypervisor IPA Fault Address Register
IPA	Intermediate Physical Address
L2C	Level 2 Cache
LDRAA/LDRAB	Load Register, with Pointer Authentication
MIDR	Main ID Register
MMIO	Memory Mapped I/O
NOP	No Operation
PAC	Pointer Authentication Code
PE	Processing Element
PIO	Programmed Input/Output
PMU	Performance Monitoring Unit
RP	Root Port
SBSA	Server Base System Architecture
SECpro	Security Processor
TLP	Transaction Layer Packet
UE	Uncorrected Error
UR	Unsupported Request
VA	Virtual Address
VMID	Virtual Machine Identifier
WFE	Wait For Event
WFI	Wait For Interrupt

Errata Summary

Table 3: Errata Summary (Sheet 1 of 2)

ERRATA NO.	CATEGORY	DESCRIPTION	PAGE
AC04_CPU_1	4	L1D_CACHE_INVAL PMU overcounts in some situations.	5

Table 3: Errata Summary (Sheet 2 of 2)

ERRATA NO.	CATEGORY	DESCRIPTION	PAGE
AC04_CPU_10	3	Certain bits in the Virtualization Translation Control Register and Translation Control Registers do not follow RES0 semantics.	*6*
AC04_CPU_14	3	Timer CVAL programming of a delta greater than 2^{63} will result in incorrect behavior.	*7*
AC04_CPU_19	5	Software reads of ICC_PMR_EL1 return incorrect value in some situations.	*8*
AC04_CPU_21	4	LD_RETIRED performance monitor event does not count atomic instructions.	*9*
AC04_CPU_23	3	Failure to synchronize writes to HCR_EL2 may corrupt address translations.	*10*
AC04_CPU_25	4	CPU can hang when two store instructions, separated by a PSB instruction, attempt to combine at retirement and the first store is profiled.	*11*
AC04_DEBUG_8	4	Cycle counts generated while Timestamping is enabled in instruction trace are incorrect.	*12*
AC04_DEBUG_9	4	TRCVICTLR.SSSTATUS might incorrectly fail to update when ViewInst EL filtering using TRCVICTLR.EXLEVEL* is configured.	*13*
AC04_DEBUG_10	5	PMCR_EL0.X does not control the export of PMU events to trace unit.	*14*
AC04_DEBUG_15	4	Applying masking (TRCCIDCCTLR.COMP0 and TRCVMIDCCTLR.COMP0) to context ID or VMID comparators (TRCCIDCVR0 and TRCVMIDCVR0) may yield incorrect results.	*15*
AC04_DEBUG_16	5	Performance Monitors Control Register (PMCR) EL0.X field does not control the export of Performance Monitoring Unit (PMU) events to the trace unit.	*16*
AC04_DEBUG_17	4	TRFCR_EL2.CX incorrectly controls Context ID tracing and Context ID comparators in SelfHostedTrace mode when HCR_EL2.E2H=1.	*17*
AC04_MCU_2	3	Non-cacheable writes to DDR memory with SKME enabled may result in corruption.	*18*
AC04_MESH_1	5	Incorrect number of children reported in the mesh crosspoint connected to HN-P nodes.	*19*
AC04_PCIE_8	3	PCIe devices will overwrite Memory Tagging Extension (MTE) allocation tag bits on DMA writes.	*20*

AC04_CPU_1: L1D_CACHE_INVAL PMU overcounts in some situations.

Functional Unit: CPU

Category: 4

Affected Version(s): AmpereOne AC04 A0, AmpereOne AC04_1 A0

Fixed Version(s): Open

Overview:

L1D_CACHE_INVAL PMU counter may overcount L1D cache invalidations by an amount anywhere from 0 up to L2C_DATA_REFILL + L2D_CACHE_INVAL.

Impact:

L1D_CACHE_INVAL PMU may be unreliable for performance tuning.

Workaround:

None.

AC04_CPU_10: Certain bits in the Virtualization Translation Control Register and Translation Control Registers do not follow RES0 semantics.

Functional Unit: CPU

Category: 3

Affected Version(s): AmpereOne AC04 A0, AmpereOne AC04_1 A0

Fixed Version(s): Open

Overview:

ID_AA64_MMFR1_EL1.HAFDBS will report a value of 0b0000 indicating that the hardware update of the access flag and dirty state are not supported. With FEAT_HAFDBS not supported, the register bits in the Virtualization Translation Control Register (VTCR_EL2) and Translation Control Registers (TCR_EL1, TCR_EL2) for enabling/disabling hardware management of access flag and dirty state – specifically VTCR_EL2.{HA, HD}, TCR_EL1.{HA, HD} and TCR_EL2.{HA, HD}, respectively must follow the RES0 semantics. These bits do not follow the RES0 semantics.

Impact:

Setting VTCR_EL2.{HA, HD}, TCR_EL1.{HA, HD} or TCR_EL2.{HA, HD} can lead to unpredictable behavior.

Workaround:

System software/virtualization system software must not set the TCR_EL1.{HA, HD}, TCR_EL2.{HA, HD} or VTCR_EL2.{HA, HD} bits.

AC04_CPU_14: Timer CVAL programming of a delta greater than 2^{63} will result in incorrect behavior.

Functional Unit: CPU

Category: 3

Affected Version(s): AmpereOne AC04 A0, AmpereOne AC04_1 A0

Fixed Version(s): Open

Overview:

In scenarios where the CompareValue (*CVAL) is greater than or equal to 2^{63} (or 292 years) difference from the counter being compared against, the AmpereOne TimerConditionMet calculation will be wrong (precisely opposite of the expected behavior). This limits Timer functionality to only configure timer interrupts to be within 292 years into the future.

Impact:

There is no expected practical use case for setting a CVAL delta this large. If there is any code that attempts to disable the timer by setting a value > 292 years into the future instead of actually masking it, then it is possible that would result in the timer firing immediately.

Workaround:

Software must enforce that the CVAL value programmed does not exceed a delta of 2^{63} with the counter being compared against.

AC04_CPU_19: Software reads of ICC_PMR_EL1 return incorrect value in some situations.

Functional Unit: CPU

Category: 5

Affected Version(s): AmpereOne AC04 A0, AmpereOne AC04_1 A0

Fixed Version(s): Open

Overview:

Non-secure MRS reads of ICC_PMR_EL1 while SCR_EL3.FIQ==1, when ICC_PMR_EL1 contains the Idle priority, will return 0xF8 instead of the correct non-secure view value of 0xF0.

Impact:

There are no known software impacts. Save/restore operations will restore the correct value in ICC_PMR_EL1 because of the shifting/masking that occurs on MSR writes to that register. Sanity checks in open source software do not commonly hit this case.

Workaround:

When reading ICC_PMR_EL1 from the non-secure security state with SCR_EL3.FIQ==1, treat a read of the value 0xF8 as if it read 0xF0.

AC04_CPU_21: LD_RETIRED performance monitor event does not count atomic instructions.

Functional Unit: CPU

Category: 4

Affected Version(s): AmpereOne AC04 A0, AmpereOne AC04_1 A0

Fixed Version(s): Open

Overview:

Atomic instructions are both memory-reading and memory-writing instructions and so should be counted by both LD_RETIRED and ST_RETIRED performance monitoring events. However, LD_RETIRED does not count atomic instructions.

Impact:

The LD_RETIRED performance monitoring event will undercount by an amount equal to the number of atomic instructions.

Workaround:

None.

AC04_CPU_23: Failure to synchronize writes to HCR_EL2 may corrupt address translations.

Functional Unit: CPU

Category: 3

Affected Version(s): AmpereOne AC04 A0, AmpereOne AC04_1 A0

Fixed Version(s): Open

Overview:

If an MSR to HCR_EL2 is not explicitly synchronized by wrapping it in a DSB instruction and an ISB instruction, address translations that are active in the AmpereOne AC04 or AC04_1 CPU may become corrupted.

Impact:

Address translations may incorrectly fault or receive incorrect information.

Workaround:

Writes to HCR_EL2 should be explicitly synchronized using the following instruction sequence:

```
DSB
MSR HCR_EL2
ISB
```

AC04_CPU_25: CPU can hang when two store instructions, separated by a PSB instruction, attempt to combine at retirement and the first store is profiled.

Functional Unit: CPU

Category: 4

Affected Version(s): AmpereOne AC04 A0, AmpereOne AC04_1 A0

Fixed Version(s): Open

Overview:

A CPU can hang when two store instructions, separated by a Profile Synchronization Barrier (PSB) instruction, attempt to combine at retirement and the first store is a profiled store.

Because PSBs do not allocate into the Store Ordering Buffer (SOB), the Load-Store Unit (LSU) stalls store retirement pending the commit of the store after the PSB. However, the PSB cannot commit until the profiled store retires, which cannot happen.

Impact:

A CPU can hang under certain conditions when a profiled store preceding a PSB by any number of instructions is retiring.

A Livelock Breaker (LLB) eventually resolves the hang, but the time taken for the hang to be resolved depends upon the LLB configuration.

Workarounds:

- Disable Store Retirement Combining Stall via `LsuSpr.Imp_Lsu_Ctrl_El3.DISABLE_ST_RET_COMBINE_STALL=1`.
- Insert a Data Synchronization Barrier (DSB) instruction after each PSB instruction.

AC04_DEBUG_8: Cycle counts generated while Timestamping is enabled in instruction trace are incorrect.

Functional Unit: DEBUG

Category: 4

Affected Version(s): AmpereOne AC04 A0, AmpereOne AC04_1 A0

Fixed Version(s): Open

Overview:

When CycleCounting (TRCCONFIGR.CCI) and Timestamp tracing (TRCCONFIGR.TS) are enabled together in instruction trace, cycle counts' output with the Timestamp Elements is incorrect relative to the previous Timestamp Element. Additionally, CycleCount Elements might not be generated at all.

Impact:

A trace analyzer that expects cycle counts associated with Timestamp Elements to be relative to the previous CycleCount Element will face issues interpreting the trace output. Cycle counts will be relative to the previous Timestamp Element or CycleCount Element in trace. The absence of CycleCount Elements might lead to incorrect instruction trace interpretation.

This may result in debug tools that collect and parse trace data to have a corrupted or incorrect trace decoded output.

Workaround:

Turning off Timestamping (TRCCONFIGR.TS=0) will result in accurate generation of CycleCount elements.

AC04_DEBUG_9: TRCVICTLR.SSSTATUS might incorrectly fail to update when ViewInst EL filtering using TRCVICTLR.EXLEVEL* is configured.

Functional Unit: DEBUG

Category: 4

Affected Version(s): AmpereOne AC04 A0, AmpereOne AC04_1 A0

Fixed Version(s): Open

Overview:

ViewInst Start/Stop function (TRCVICTLR.SSSTATUS) might be incorrect if the start/stop point addresses (enabled by TRCVISSCTLR) execute in an exception level (EL) configured by TRCVICTLR.EXLEVEL* to filter instruction trace.

Impact:

An out-of-date ViewInst Start/Stop function might mean some desired address regions do not get traced once execution exits out of the filtered EL.

Workaround:

Users configuring the debug trace configuration must avoid configuring the start/stop PCs in a filtered EL.

AC04_DEBUG_10: PMCR_EL0.X does not control the export of PMU events to trace unit.

Functional Unit: DEBUG

Category: 5

Affected Version(s): AmpereOne AC04 A0, AmpereOne AC04_1 A0

Fixed Version(s): Open

Overview:

PMCR_EL0.X does not control the export of PMU events over to external sources, such as the trace unit.

Impact:

Incorrect trace state might be reported or additional trace might be generated when PMU events are configured as external inputs to trace unit while PMU counting and tracing are both allowed.

Workaround:

PMU events can be deselected as external inputs to trace when export of PMU events is intended to be disabled. Alternatively, PMU counters can be disabled while tracing to prevent any unintentional export of PMU events.

AC04_DEBUG_15: Applying masking (TRCCIDCCTLR.COMP0 and TRCVMIDCCTLR.COMP0) to context ID or VMID comparators (TRCCIDCVR0 and TRCVMIDCVR0) may yield incorrect results.

Functional Unit: DEBUG

Category: 4

Affected Version(s): AmpereOne AC04 A0, AmpereOne AC04_1 A0

Fixed Version(s): Open

Overview:

Masking configured by TRCCIDCCTLR.COMP0 and TRCVMIDCCTLR.COMP0 for Context ID and VMID matching does not apply to comparator values configured in TRCCIDCVR0 and TRCVMIDCVR0, respectively.

Impact:

Incorrect Context ID or VMID matching can result in incorrect trace resource operation and unintended trace output.

Workaround:

When TRCCIDCCTLR.COMP0 and TRCVMIDCCTLR.COMP0 are configured, users should manually mask the values configured in TRCCIDCVR0 and TRCVMIDCVR0. SelfHostedTrace already applies this as a software workaround.

AC04_DEBUG_16: Performance Monitors Control Register (PMCR) EL0.X field does not control the export of Performance Monitoring Unit (PMU) events to the trace unit.

Functional Unit: DEBUG

Category: 5

Affected Version(s): AmpereOne AC04 A0, AmpereOne AC04_1 A0

Fixed Version(s): Open

Overview:

Performance Monitors Control Register (PMCR) EL0.X field does not control the export of PMU events to external sources, such as the trace unit.

Impact:

Incorrect trace state may be reported, or additional trace data may be generated, when PMU events are configured as external inputs to trace unit while PMU counting and tracing are both allowed.

Workaround:

PMU events can be not selected as external inputs to trace when export of PMU events is intended to be disabled. Alternatively, PMU counters can be disabled while tracing to prevent any unintentional export of PMU events.

AC04_DEBUG_17: TRFCR_EL2.CX incorrectly controls Context ID tracing and Context ID comparators in SelfHostedTrace mode when HCR_EL2.E2H=1.

Functional Unit: DEBUG

Category: 4

Affected Version(s): AmpereOne AC04 A0, AmpereOne AC04_1 A0

Fixed Version(s): Open

Overview:

Context ID tracing (TRCCONFIGR.CID) is disabled and Context ID comparators (TRCCIDCVR) mismatch when TRFCR_EL2.CX=1 when in SelfHostedTrace mode and HCR_EL2.E2H=1. This erratum does not affect Virtual Context ID tracing (TRCCONFIGR.VMID) or comparators (TRCVMIDCVR).

Impact:

Context ID (CONTEXTIDR_EL1) is not output in trace stream in the described condition. Context ID comparators mismatch, which may lead to unintended trace output.

Workaround:

None.

AC04_MCU_2: Non-cacheable writes to DDR memory with SKME enabled may result in corruption.

Functional Unit: MCU

Category: 3

Affected Version(s): AmpereOne AC04 A0, AmpereOne AC04_1 A0

Fixed Version(s): Open

Overview:

When Single-Key-Memory-Encryption (SKME) is enabled, non-cacheable writes to encrypted DDR memory may result in SRAM parity errors at the Memory Controller Unit (MCU) or may lead to silent data corruption.

The conditions for this erratum to manifest are:

- SKME enabled in boot firmware via NVPARAM option
- A high rate of back-to-back partial (non-cacheable) writes to DDR memory

Impact:

If SKME is enabled, specialized software or tools that map DDR memory as non-cacheable may result in unreliable system functionality.

Mainstream OS and hypervisor software (such as Linux) will not be impacted due to lack of support for mapping “normal memory” as non-cacheable. Software will typically map DDR memory as cacheable and honor the recommended memory attributes provided by UEFI via the firmware memory map. While it is possible for some specialized software to map DDR memory as non-cacheable or “device” per the architecture, this would be an impractical software design choice on a fully coherent system.

Debug tools (such as OpenOCD) that attempt to trace to memory when memory encryption is enabled would be impacted if the Embedded Trace Router (ETR) is configured to write trace data with the non-cacheable attribute.

Workaround:

Always access DDR memory using cacheable transactions from software (using appropriate page table attributes in both the MMU and SMMU) and debug tools (using ETR settings).

AmpereOne reference UEFI firmware will map all DDR memory regions as cacheable, and OS software is expected to honor these memory map attributes when setting up heap memory. In addition, reference UEFI firmware will set up the ACPI IO Remapping Table (IORT) and PCIe Root Port configuration to always perform cacheable accesses as well.

AC04_MESH_1: Incorrect number of children reported in the mesh crosspoint connected to HN-P nodes.

Functional Unit: Mesh

Category: 5

Affected Version(s): AmpereOne AC04 A0, AmpereOne AC04_1 A0

Fixed Version(s): Open

Overview:

Each mesh crosspoint has a register named “por_mxp_child_info”, which includes a field named “child_count”. This field is intended to report the number of child pointers associated with this crosspoint. For crosspoints that report connected devices of “device_type” 5’b01011 (HN-P type), the child_count value will be incorrectly reported as 8. The correct value must be 2.

Impact:

During software or firmware mesh discovery software flows, this may result in software attempting to dereference the third child pointer, which returns zero. This will result in incorrect discovery behavior and may result in invalid discovery information. The invalid discovery information may result in improper behavior of software or firmware dependent on this information.

Workaround:

In conditions where:

- The device_type is HN-P, or
- The value of the child pointer is zero

Software must ignore and skip the remaining child pointers within the child_info structure.

AC04_PCIE_8: PCIe devices will overwrite Memory Tagging Extension (MTE) allocation tag bits on DMA writes.

Functional Unit: PCIe

Category: 3

Affected Version(s): AmpereOne AC04 A0, AmpereOne AC04_1 A0

Fixed Version(s): Open

Overview:

Memory tagging extension (MTE) is an Armv8 Instruction Set Architecture (ISA) extension that can be highly useful to software for automatic detection and mitigation of buffer overflow bugs, security vulnerabilities, and pointer programming errors initiated by software writes to these buffers.

The architecture requires that when a direct-memory-access (DMA) agent writes to tagged memory, the value of the allocation tag bit be preserved. In this implementation, the allocation tag bits may be overwritten to zero on system level cache (SLC) misses.

Impact:

Memory tagging may not work reliably due to unexpected tag faults when software accesses tagged memory buffers allocated for the purpose of PCIe DMA writes.

Workaround:

There are two options to work around this issue:

1. Software should avoid mapping DMA buffers as tagged memory.
2. Prevent tag faults on allocation tag value of zero by modifying NVPARAM boot settings:
   - Set the `NVPSBOOT.FEATURE_CTRL.MTE_IGNORE_ZERO_TAG_ALLOCATION` bit as described in the *AmpereOne Family NVPARAM Specification*[1]. This bit configures the core to enable a feature to Ignore Zero Tag Allocations. When set, the CPU core ignores memory tag mismatches if the allocation tag is 0.

   Note: This workaround should be supplemented with software changes to avoid using the allocation tag value of zero. This may involve changing the default tag value to a non-zero value (such as 0xF).

References:

1. The latest versions of the *NVPARAM Specification* can be downloaded from Ampere's Customer Connect portal (*requires login credentials*) as listed below:
   - *AmpereOne AC03 and AC04 NVPARAM Specification* from any of the latest SRP 4.x tarballs
   - *AmpereOne AC04_1 NVPARAM Specification* from any of the latest SRP 5.x tarballs
2. The latest version of the *AmpereOne AC04 and AC04_1 MTE Supplement* can be downloaded from any of the latest SRP 4.x or 5.x tarballs from Ampere's Customer Connect portal.

Revision History

ISSUE	DATE	DESCRIPTION OF MODIFICATIONS
1.20	July 15, 2025	Added erratum AC04_CPU_25 on *page 11*.
		Updated erratum AC04_DEBUG_9 on *page 13*.
		Added erratum AC04_DEBUG_10 on *page 14*.
		Added erratum AC04_DEBUG_15 on *page 15*.
		Added erratum AC04_DEBUG_16 on *page 16*.
		Added erratum AC04_DEBUG_17 on *page 17*.
		Added erratum AC04_PCIE_8 on *page 20*.
1.10	January 25, 2025	Added erratum AC04_CPU_21 on *page 9*.
		Added erratum AC04_CPU_23 on *page 10*.
1.05	January 10, 2025	Added erratum AC04_DEBUG_8 on *page 12*.
		Added erratum AC04_DEBUG_9 on *page 13*.
1.00	August 27, 2024	Added erratum AC04_CPU_1 on *page 5*.
		Added erratum AC04_CPU_10 on *page 6*.
		Added erratum AC04_CPU_14 on *page 7*.
		Added erratum AC04_CPU_19 on *page 8*.
		Added erratum AC04_MCU_2 on *page 18*.
		Added erratum AC04_MESH_1 on *page 19*.

July 15, 2025

Ampere Computing reserves the right to change or discontinue this product without notice.

While the information contained herein is believed to be accurate, such information is preliminary, and should not be relied upon for accuracy or completeness, and no representations or warranties of accuracy or completeness are made.

The information contained in this document is subject to change or withdrawal at any time without notice and is being provided on an "AS IS" basis without warranty or indemnity of any kind, whether express or implied, including without limitation, the implied warranties of non-infringement, merchantability, or fitness for a particular purpose.

Any products, services, or programs discussed in this document are sold or licensed under Ampere Computing's standard terms and conditions, copies of which may be obtained from your local Ampere Computing representative. Nothing in this document shall operate as an expressed or implied license or indemnity under the intellectual property rights of Ampere Computing or third parties.

Without limiting the generality of the foregoing, any performance data contained in this document was determined in a specific or controlled environment and not submitted to any formal Ampere Computing test. Therefore, the results obtained in other operating environments may vary significantly. Under no circumstances will Ampere Computing be liable for any damages whatsoever arising out of or resulting from any use of the document or the information contained herein.

Ampere Computing
4655 Great America Parkway, Santa Clara, CA 95054
Phone: (669) 770-3700
https://www.amperecomputing.com

Ampere Computing reserves the right to make changes to its products, its datasheets, or related documentation, without notice and warrants its products solely pursuant to its terms and conditions of sale, only to substantially comply with the latest available datasheet.