

NAME**simd** - SIMD enhancements**DESCRIPTION**

On some architectures, the FreeBSD *libc* provides enhanced implementations of commonly used functions, replacing the architecture-independent implementations used otherwise. Depending on architecture and function, an enhanced implementation of a function may either always be used or the *libc* detects at runtime which SIMD instruction set extensions are supported and picks the most suitable implementation automatically. On **amd64**, the environment variable ARCHLEVEL can be used to override this mechanism.

Enhanced functions are present for the following architectures:

<i>FUNCTION</i>	<i>AARCH64</i>	<i>ARM</i>	<i>AMD64</i>	<i>I386</i>	<i>PPC64</i>
bcmp			S1	S	
bcopy		S	S	S	SV
bzero		S	S	S	
div			S	S	
index	A		S1		
ldiv			S	S	
lldiv			S		
memchr	A		S1		
memcmp	A	S	S1	S	
memccpy			S1		
memcpy	S	S	S	S	SV
memmove	S	S	S	S	SV
memrchr	A		S1		
memset	A	S	S	S	
rindex	A		S1	S	
stpcpy	A		S1		
stpncpy			S1		
strcat			S1	S	
strchr	A		S1	S	
strchrnul	A		S1		
strcmp	S	S	S1	S	
strcpy	A		S1	S	S2
strncpy			S2		

strlcat			S1	
strncpy			S1	
strlen	A	S	S1	
strncat			S1	
strncmp	S	S	S1	S
strncpy			S1	S2
strnlen	A		S1	
strrchr	A		S1	S
strpbrk			S2	
strsep			S2	
strspn			S2	
swab				S
timingsafe_bcmp			S1	
timingsafe_memcmp			S	
wcschr				S
wscmp				S
wcslen				S
wmemchr				S

S: scalar (non-SIMD), **1:** amd64 baseline, **2:** x86-64-v2 or PowerPC 2.05, **3:** x86-64-v3, **4:** x86-64-v4, **V:** PowerPC VSX, **A:** Arm ASIMD (NEON).

ENVIRONMENT

ARCHLEVEL

On *amd64*, controls the level of SIMD enhancements used. If this variable is set to an architecture level from the list below and that architecture level is supported by the processor, SIMD enhancements up to ARCHLEVEL are used. If ARCHLEVEL is unset, not recognised, or not supported by the processor, the highest level of SIMD enhancements supported by the processor is used.

A suffix beginning with ‘:’ or ‘+’ in ARCHLEVEL is ignored and may be used for future extensions. The architecture level can be prefixed with a ‘!’ character to force use of the requested architecture level, even if the processor does not advertise that it is supported. This usually causes applications to crash and should only be used for testing purposes or if architecture level detection yields incorrect results.

The architecture levels follow the AMD64 SysV ABI supplement:

scalar scalar enhancements only (no SIMD)

baseline cmov, cx8, x87 FPU, fxsr, MMX, osfxsr, SSE, SSE2

x86-64-v2 cx16, lahf/sahf, popcnt, SSE3, SSSE3, SSE4.1, SSE4.2

x86-64-v3 AVX, AVX2, BMI1, BMI2, F16C, FMA, lzcnt, movbe, osxsave

x86-64-v4 AVX-512F/BW/CD/DQ/VL

DIAGNOSTICS

Illegal Instruction Printed by `sh(1)` if a command is terminated through delivery of a SIGILL signal, see `signal(3)`.

Use of an unsupported architecture level was forced by setting `ARCHLEVEL` to a string beginning with a `!` character, causing a process to crash due to use of an unsupported instruction. Unset `ARCHLEVEL`, remove the `!` prefix or select a supported architecture level.

Message may also appear for unrelated reasons.

SEE ALSO

`string(3)`, `arch(7)`

H. J. Lu, Michael Matz, Milind Girkar, Jan Hubicka, Andreas Jaeger, and Mark Mitchell, "AMD64 Architecture Processor Supplement", *System V Application Binary Interface*, May 23, 2023, Version 1.0.

HISTORY

Architecture-specific enhanced *libc* functions were added starting with FreeBSD 2.0 for **i386**, FreeBSD 6.0 for **arm**, FreeBSD 6.1 for **amd64**, FreeBSD 11.0 for **aarch64**, and FreeBSD 12.0 for **powerpc64**. SIMD-enhanced functions were first added with FreeBSD 13.0 for **powerpc64** and with FreeBSD 14.1 for **amd64**.

A **simd** manual page appeared in FreeBSD 14.1.

AUTHOR

Robert Clausecker <fuz@FreeBSD.org>

CAVEATS

Other parts of FreeBSD such as cryptographic routines in the kernel or in OpenSSL may also use SIMD enhancements. These enhancements are not subject to the `ARCHLEVEL` variable and may have their own configuration mechanism.

BUGS

Use of SIMD enhancements cannot be configured on powerpc64.