Extension Naming Convention

- All RISC-V standard extensions that are not a single capital letter start with either a S (Privileged ISA extension) or a Z (Unprivileged ISA extension).
- The second letter of the extension name specifies the category of the extension.

### S* Categories (Privileged ISA)

The second letter is the category of the extension (usually the privilege level associated with the extension).

<table>
<thead>
<tr>
<th>S* Prefix</th>
<th>Meaning</th>
</tr>
</thead>
<tbody>
<tr>
<td>Sd</td>
<td>Debug</td>
</tr>
<tr>
<td>Sh</td>
<td>Hypervisor (HS-mode and VS-mode)</td>
</tr>
<tr>
<td>Sm</td>
<td>M-mode</td>
</tr>
<tr>
<td>Ss</td>
<td>S-mode</td>
</tr>
<tr>
<td>Su</td>
<td>U-mode</td>
</tr>
<tr>
<td>Sv</td>
<td>Virtual memory</td>
</tr>
<tr>
<td>S?n</td>
<td>Level immediately less than &quot;?&quot; level</td>
</tr>
</tbody>
</table>

A new convention started with the pointer masking extension in May 2023 was to use a 3rd letter of "n" to indicate the next lower privilege level (whatever that is for a particular implementation). The extension only exerts control at the next lower level, not at all lower levels.

### S* Examples

<table>
<thead>
<tr>
<th>S* Extension</th>
<th>Meaning</th>
</tr>
</thead>
<tbody>
<tr>
<td>Shvstvecd</td>
<td>The vstvec.MODE CSR field supports direct mode</td>
</tr>
<tr>
<td>Svbare</td>
<td>The S-mode virtual-memory satp &quot;Bare&quot; mode is supported</td>
</tr>
<tr>
<td>Ss1p13</td>
<td>Privileged Architecture version 1.13 (p = period)</td>
</tr>
<tr>
<td>Sv39</td>
<td>The S-mode virtual-memory address size is 39 bits</td>
</tr>
</tbody>
</table>

### Z* Categories (Unprivileged ISA)

The second letter is often the original RISC-V capital letter extension name as listed in the RISC-V Unprivileged ISA specification. For example, A = Atomics so Za is the prefix for atomic-related extensions.

<table>
<thead>
<tr>
<th>Z* Prefix</th>
<th>Meaning</th>
<th>Z* Prefix</th>
<th>Meaning</th>
</tr>
</thead>
<tbody>
<tr>
<td>Za</td>
<td>Atomics</td>
<td>Zj</td>
<td>Java</td>
</tr>
<tr>
<td>Zb</td>
<td>Bit manipulation</td>
<td>Zk</td>
<td>Scalar crypto</td>
</tr>
<tr>
<td>Zc</td>
<td>Compressed</td>
<td>Zm</td>
<td>Multiplication and division</td>
</tr>
<tr>
<td>Zf</td>
<td>Single-precision floating-point</td>
<td>Zv</td>
<td>Vector</td>
</tr>
<tr>
<td>Zh</td>
<td>Hypervisor</td>
<td>Zvk</td>
<td>Scalar crypto</td>
</tr>
<tr>
<td>Zi</td>
<td>Base integer ISA</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

### Z* Examples

<table>
<thead>
<tr>
<th>Z* Extension</th>
<th>Meaning</th>
</tr>
</thead>
<tbody>
<tr>
<td>Zbb</td>
<td>Basic bit manipulation instructions</td>
</tr>
<tr>
<td>Zfhmin</td>
<td>Half-precision floating-point transfer and conversion instructions</td>
</tr>
<tr>
<td>Zkne</td>
<td>AES scalar encryption</td>
</tr>
<tr>
<td>Zpm</td>
<td>Pointer-masking extension for J category</td>
</tr>
<tr>
<td>Zicsr</td>
<td>CSR read/write instructions</td>
</tr>
<tr>
<td>Zicbom</td>
<td>Cache block management instructions</td>
</tr>
</tbody>
</table>

### RISC-V Reference Links

These links explain the naming convention shown above.

### RISC-V Unprivileged ISA Specification

See "ISA Extension Naming Conventions" chapter.

### Google groups post in 2017 Excerpts

The current RISC-V standard instruction-set namespace consists of 26 single-letter names, and each letter typically corresponds to a relatively large number of instructions. While this has worked well to concisely specify the initial set of extensions and to convey the concept of a modular instruction set, there is a need for a larger instruction-set namespace to accommodate future instruction-set extensions.

The proposal is to use multicharacter names beginning with Z to expand the namespace for standard extensions, for example, Zftrig to name the ftrig extension. Underscores are used in an ISA name string to separate Z names from other components, and instruction-set names are case-insensitive.

All Z instruction-set names begin with a single letter that indicates the category of the extension. All existing single-letter names retain their existing meaning, and are given alternate names in the Z namespace of Z followed by the single letter, e.g., Za = A. Each existing single letter name is now also considered to name a category, and additional instruction-set extensions can be added to the existing single-letter categories, e.g., Zftrig. The unallocated single-letter names are now reserved for future instruction-set categories, rather than a single fixed extension.
GitHub post in 2021 to RISC-V ISA manual Excerpts

The convention is that $S^*$ represents extensions to the Privileged architecture (versus the Unprivileged arch). The second letter represents categories:

- $v$ = virtual memory
- $d$ = debug
- $m$ = M-mode
- $s$ = S-mode

ISA extensions that include M-mode and S-mode functionality will be represented as two privilege-level specific extensions.