

Chapter 26

“Ztso” Standard Extension for Total Store Ordering, Version 1.0

This chapter defines the “Ztso” extension for the RISC-V Total Store Ordering (RVTSO) memory consistency model. RVTSO is defined as a delta from RVWMO, which is defined in Chapter 17.1.

The Ztso extension is meant to facilitate the porting of code originally written for architectures with TSO memory models, such as x86 or some versions of SPARC. It also supports implementations which inherently provide RVTSO behavior and want to expose that fact to software.

RVTSO makes the following adjustments to RVWMO:

- All load operations behave as if they have an acquire-RCpc annotation
- All store operations behave as if they have a release-RCpc annotation.
- All AMOs behave as if they have both acquire-RCsc and release-RCsc annotations.

These rules render all PPO rules except 4–7 redundant. They also make redundant any non-I/O fences that do not have both PW and SR set. Finally, they also imply that no memory operation will be reordered past an AMO in either direction.

In the context of RVTSO, as is the case for RVWMO, the storage ordering annotations are concisely and completely defined by PPO rules 5–7. In both of these memory models, it is the [Load Value Axiom](#) that allows a hart to forward a value from its store buffer to a subsequent (in program order) load—that is to say that stores can be forwarded locally before they are visible to other harts.

Additionally, if the Ztso extension is implemented, then vector memory instructions in the V extension and Zve family of extensions follow RVTSO at the instruction level. The Ztso extension does not strengthen the ordering of intra-instruction element accesses.

In spite of the fact that Ztso adds no new instructions to the ISA, code written assuming RVTSO will not run correctly on implementations not supporting Ztso. Binaries compiled to run only under Ztso should indicate as such via a flag in the binary, so that platforms which do not implement Ztso can simply refuse to run them.