700 LAR and LSL Instructions Do Not Check Invalid Long Mode Descriptor Types

Description

The architecture specifies that the processor checks for invalid descriptor types when a Load Access Rights Byte (LAR) instruction or a Load Segment Limit (LSL) instruction is executed in long mode. An invalid descriptor type should cause the processor to clear the zero flag (ZF) and complete the instruction without modifying the destination register. However, the processor does not perform this check and loads the attribute (LAR) or segment limit (LSL) as if the descriptor type was valid.

The invalid descriptor types for LAR are 1 (available 16-bit TSS), 3 (busy 16-bit TSS), 4 (16-bit call gate) or 5 (task gate). The invalid descriptor types for a LSL instruction are types 1 (available 16-bit TSS) or 3 (busy 16-bit TSS).

Potential Effect on System

None expected, since the operating system code would typically only provide legal descriptors. However, in the case of erroneous software, the above described check would not be performed, resulting in unpredictable system failure. AMD has not observed this erratum with any commercially available software.

Suggested Workaround

None required, it is anticipated that long mode operating system code ensures that the descriptor type is legal when executing LAR and LSL instructions.

Fix Planned

No