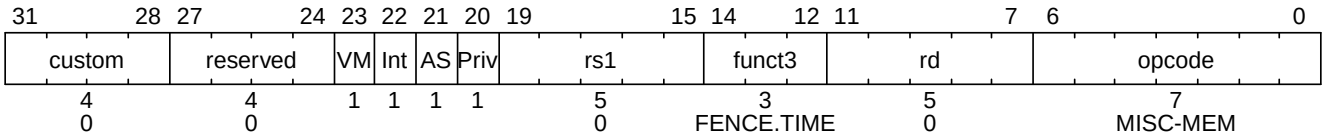
"Zifencetime" Extension for Timing Fence

FENCE.TIME is an instruction that can be used to ensure microarchitectural state isolation. FENCE.TIME is a contract between software and hardware; its semantics are intended to prevent the occurrence of covert channels.



[FENCE.TIME Instruction Encoding]

```
The timing of any instruction or sequence of instructions that execute after
FENCE.TIME must be independent of any sequence of instructions before the fence, or
equivalently independent of the microarchitectural state before the fence.
```

Flags indicating the associated security boundary may exclude this requirement for some subsets of microarchitectural state.

The `Priv` flag equal to 1 indicates that the fence is associated with a privilege-level switch. Notably, branch predictor state can be partitioned by privilege level in the implementation. In this case, a privilege-level switch would not require flushing this particular state.

The `AS` flag equal to 1 indicates a change of address space, typically associated with an ASID modification.

The `Int` flag equal to 1 indicates that the fence is applied as part of state isolation in relation to an interrupt.

The `VM` flag equal to 1 indicates a switch to another virtual machine, typically associated with a VMID modification.

NOTE	This instruction is functionally equivalent to a NOP. Adding or removing this instruction does not change the functionality of the program, but it can drastically alter performance.

WARNING	The effectiveness of FENCE.TIME depends heavily on the underlying hardware implementation, which may vary between systems. Even when FENCE.TIME is used correctly, a comprehensive security evaluation is essential to ensure that the intended microarchitectural state isolation is achieved and that all potential covert channels are effectively mitigated.