# TLBI ALLE1IS, TLBI ALLE1ISNXS, TLB Invalidate All, EL1, Inner Shareable

The TLBI ALLE1IS, TLBI ALLE1ISNXS characteristics are:

## Purpose

Invalidates cached copies of translation table entries from TLBs that meet all the following requirements:

- The entry is a stage 1 or stage 2 translation table entry, from any level of the translation table walk.
- If FEAT_RME is implemented, one of the following applies:
  - SCR_EL3.{NSE, NS} is {0, 0} and the entry would be required to translate an address using the Secure EL1&0 translation regime.
  - SCR_EL3.{NSE, NS} is {0, 1} and the entry would be required to translate an address using the Non-secure EL1&0 translation regime.
  - SCR_EL3.{NSE, NS} is {1, 1} and the entry would be required to translate an address using the Realm EL1&0 translation regime.
- If FEAT_RME is not implemented, one of the following applies:
  - SCR_EL3.NS is 0 and the entry would be required to translate an address using the Secure EL1&0 translation regime.
  - SCR_EL3.NS is 1 and the entry would be required to translate an address using the Non-secure EL1&0 translation regime.

The invalidation applies to entries with any VMID.

The invalidation applies to all PEs in the same Inner Shareable shareability domain as the PE that executes this System instruction.

---

**Note**

For the EL1&0 translation regimes, the invalidation applies to both global entries and non-global entries with any ASID.

---

If FEAT_XS is implemented, the nXS variant of this System instruction is defined.

Both variants perform the same invalidation, but the TLBI System instruction without the nXS qualifier waits for all memory accesses using in-scope old translation information to complete before it is considered complete.

The TLBI System instruction with the nXS qualifier is considered complete when the subset of these memory accesses with XS attribute set to 0 are complete.

## Configuration

There are no configuration notes.

## Attributes

TLBI ALLE1IS, TLBI ALLE1ISNXS is a 64-bit System instruction.

## Field descriptions

This instruction has no applicable fields.

The value in the register specified by <Xt> is ignored.

## Executing TLBI ALLE1IS, TLBI ALLE1ISNXS

The Rt field should be set to `0b11111`. If the Rt field is not set to `0b11111`, it is constrained unpredictable whether:

- The instruction is undefined.
- The instruction behaves as if the Rt field is set to `0b11111`.

Accesses to this instruction use the following encodings in the System instruction encoding space:

```
TLBI ALLE1IS{, <Xt>}
```

| op0 | op1 | CRn | CRm | op2 |
|---|---|---|---|---|
| 0b01 | 0b100 | 0b1000 | 0b0011 | 0b100 |

```
if PSTATE.EL == EL0 then
    UNDEFINED;
elsif PSTATE.EL == EL1 then
    if EL2Enabled() && HCR_EL2.NV == '1' then
        AArch64.SystemAccessTrap(EL2, 0x18);
```

```
    else
        UNDEFINED;
elsif PSTATE.EL == EL2 then
    AArch64.TLBI_ALL(SecurityStateAtEL(EL1),
Regime_EL10, Shareability_ISH, TLBI_AllAttr);
elsif PSTATE.EL == EL3 then
    AArch64.TLBI_ALL(SecurityStateAtEL(EL1),
Regime_EL10, Shareability_ISH, TLBI_AllAttr);
```

## TLBI ALLE1ISNXS{, <Xt>}

| op0 | op1 | CRn | CRm | op2 |
|---|---|---|---|---|
| 0b01 | 0b100 | 0b1001 | 0b0011 | 0b100 |

```
if !IsFeatureImplemented(FEAT_XS) then
    UNDEFINED;
elsif PSTATE.EL == EL0 then
    UNDEFINED;
elsif PSTATE.EL == EL1 then
    if EL2Enabled() && HCR_EL2.NV == '1' then
        AArch64.SystemAccessTrap(EL2, 0x18);
    else
        UNDEFINED;
elsif PSTATE.EL == EL2 then
    AArch64.TLBI_ALL(SecurityStateAtEL(EL1),
Regime_EL10, Shareability_ISH, TLBI_ExcludeXS);
elsif PSTATE.EL == EL3 then
    AArch64.TLBI_ALL(SecurityStateAtEL(EL1),
Regime_EL10, Shareability_ISH, TLBI_ExcludeXS);
```



28/03/2023 16:02; 72747e43966d6b97dcbd230a1b3f0421d1ea3d94