TRCVMIDCCTLR1, Virtual Context Identifier Comparator Control Register 1

The TRCVMIDCCTLR1 characteristics are:

Purpose

Virtual Context Identifier Comparator mask values for the TRCVMIDCVR<n> registers, where n=4-7.

Configuration

AArch64 System register TRCVMIDCCTLR1 bits [31:0] are architecturally mapped to External register TRCVMIDCCTLR1[31:0].

This register is present only when FEAT_ETE is implemented, FEAT_TRC_SR is implemented, UInt(TRCIDR4.NUMVMIDC) > 0x4 and UInt(TRCIDR2.VMIDSIZE) > 0. Otherwise, direct accesses to TRCVMIDCCTLR1 are undefined.

Attributes

TRCVMIDCCTLR1 is a 64-bit register.

Field descriptions

63	62	61	60	59	58	57	56	55	54
COMP7[7]	COMP7[6]	COMP7[5]	COMP7[4]	COMP7[3]	COMP7[2]	COMP7[1]	COMP7[0]	COMP6[7]	COMP6[6]
31	30	29	28	27	26	25	24	23	22

Bits [63:32]

Reserved, res0.

COMP7[<m>], bit [m+24], for m = 7 to 0
When UInt(TRCIDR4.NUMVMIDC) > 7:

TRCVMIDCVR7 mask control. Specifies the mask value that the trace unit applies to TRCVMIDCVR7. Each bit in this field corresponds to a byte in TRCVMIDCVR7.

COMP7[<m>]	Meaning

0b0	The trace unit includes TRCVMIDCVR7[(mÃ—8+7):(mÃ—8)] when it performs the Virtual context identifier comparison.
0b1	The trace unit ignores TRCVMIDCVR7[(mÃ—8+7):(mÃ—8)] when it performs the Virtual context identifier comparison.

This bit is res0 if m >= TRCIDR2.VMIDSIZE.

The reset behavior of this field is:

- On a Trace unit reset, this field resets to an architecturally unknown value.

Otherwise:

Reserved, res0.

COMP6[<m>], bit [m+16], for m = 7 to 0
When UInt(TRCIDR4.NUMVMIDC) > 6:

TRCVMIDCVR6 mask control. Specifies the mask value that the trace unit applies to TRCVMIDCVR6. Each bit in this field corresponds to a byte in TRCVMIDCVR6.

COMP6[<m>]	Meaning
0b0	The trace unit includes TRCVMIDCVR6[(mÃ—8+7):(mÃ—8)] when it performs the Virtual context identifier comparison.
0b1	The trace unit ignores TRCVMIDCVR6[(mÃ—8+7):(mÃ—8)] when it performs the Virtual context identifier comparison.

This bit is res0 if m >= TRCIDR2.VMIDSIZE.

The reset behavior of this field is:

- On a Trace unit reset, this field resets to an architecturally unknown value.

Otherwise:

Reserved, res0.

COMP5[<m>], bit [m+8], for m = 7 to 0
When UInt(TRCIDR4.NUMVMIDC) > 5:

TRCVMIDCVR5 mask control. Specifies the mask value that the trace unit applies to TRCVMIDCVR5. Each bit in this field corresponds to a byte in TRCVMIDCVR5.

COMP5[<m>]	Meaning
0b0	The trace unit includes TRCVMIDCVR5[(mÃ—8+7):(mÃ—8)] when it performs the Virtual context identifier comparison.
0b1	The trace unit ignores TRCVMIDCVR5[(mÃ—8+7):(mÃ—8)] when it performs the Virtual context identifier comparison.

This bit is res0 if m >= TRCIDR2.VMIDSIZE.

The reset behavior of this field is:

- On a Trace unit reset, this field resets to an architecturally unknown value.

Otherwise:

Reserved, res0.

COMP4[<m>], bit [m], for m = 7 to 0
When UInt(TRCIDR4.NUMVMIDC) > 4:

TRCVMIDCVR4 mask control. Specifies the mask value that the trace unit applies to TRCVMIDCVR4. Each bit in this field corresponds to a byte in TRCVMIDCVR4.

COMP4[<m>]	Meaning
0b0	The trace unit includes TRCVMIDCVR4[(mÃ—8+7):(mÃ—8)] when it performs the Virtual context identifier comparison.

0b1	The trace unit ignores TRCVMIDCVR4[(mÃ—8+7):(mÃ—8)] when it performs the Virtual context identifier comparison.

This bit is res0 if m >= TRCIDR2.VMIDSIZE.

The reset behavior of this field is:

- On a Trace unit reset, this field resets to an architecturally unknown value.

Otherwise:

Reserved, res0.

Accessing TRCVMIDCCTLR1

If software uses the TRCVMIDCVR<n> registers, where n=4-7, then it must program this register.

If software sets a mask bit to 1 then it must program the relevant byte in TRCVMIDCVR<n> to `0x00`.

If any bit is 1 and the relevant byte in TRCVMIDCVR<n> is not `0x00`, the behavior of the Virtual Context Identifier Comparator is constrained unpredictable. In this scenario the comparator might match unexpectedly or might not match.

Writes are constrained unpredictable if the trace unit is not in the Idle state.

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

MRS <Xt>, TRCVMIDCCTLR1

op0	op1	CRn	CRm	op2
0b10	0b001	0b0011	0b0011	0b010

```
if PSTATE.EL == EL0 then
    UNDEFINED;
elsif PSTATE.EL == EL1 then
    if Halted() && HaveEL(EL3) && EDSCR.SDD == '1'
&& boolean IMPLEMENTATION_DEFINED "EL3 trap priority
when SDD == '1'" && CPTR_EL3.TTA == '1' then
```

```
        UNDEFINED;
    elsif CPACR_EL1.TTA == '1' then
        AArch64.SystemAccessTrap(EL1, 0x18);
    elsif EL2Enabled() && CPTR_EL2.TTA == '1' then
        AArch64.SystemAccessTrap(EL2, 0x18);
    elsif EL2Enabled() &&
IsFeatureImplemented(FEAT_FGT) && (!HaveEL(EL3) ||
SCR_EL3.FGTEn == '1') && HDFGRTR_EL2.TRC == '1' then
        AArch64.SystemAccessTrap(EL2, 0x18);
    elsif HaveEL(EL3) && CPTR_EL3.TTA == '1' then
        if Halted() && EDSCR.SDD == '1' then
            UNDEFINED;
        else
            AArch64.SystemAccessTrap(EL3, 0x18);
    else
        X[t, 64] = TRCVMIDCCTLR1;
elsif PSTATE.EL == EL2 then
    if Halted() && HaveEL(EL3) && EDSCR.SDD == '1'
&& boolean IMPLEMENTATION_DEFINED "EL3 trap priority
when SDD == '1'" && CPTR_EL3.TTA == '1' then
        UNDEFINED;
    elsif CPTR_EL2.TTA == '1' then
        AArch64.SystemAccessTrap(EL2, 0x18);
    elsif HaveEL(EL3) && CPTR_EL3.TTA == '1' then
        if Halted() && EDSCR.SDD == '1' then
            UNDEFINED;
        else
            AArch64.SystemAccessTrap(EL3, 0x18);
    else
        X[t, 64] = TRCVMIDCCTLR1;
elsif PSTATE.EL == EL3 then
    if CPTR_EL3.TTA == '1' then
        AArch64.SystemAccessTrap(EL3, 0x18);
    else
        X[t, 64] = TRCVMIDCCTLR1;
```

MSR TRCVMIDCCTLR1, <Xt>

op0	op1	CRn	CRm	op2
0b10	0b001	0b0011	0b0011	0b010

```
if PSTATE.EL == EL0 then
    UNDEFINED;
elsif PSTATE.EL == EL1 then
    if Halted() && HaveEL(EL3) && EDSCR.SDD == '1'
&& boolean IMPLEMENTATION_DEFINED "EL3 trap priority
when SDD == '1'" && CPTR_EL3.TTA == '1' then
        UNDEFINED;
    elsif CPACR_EL1.TTA == '1' then
        AArch64.SystemAccessTrap(EL1, 0x18);
    elsif EL2Enabled() && CPTR_EL2.TTA == '1' then
        AArch64.SystemAccessTrap(EL2, 0x18);
    elsif EL2Enabled() &&
```

```
IsFeatureImplemented(FEAT_FGT) && (!HaveEL(EL3) ||
SCR_EL3.FGTEn == '1') && HDFGWTR_EL2.TRC == '1' then
        AArch64.SystemAccessTrap(EL2, 0x18);
    elsif HaveEL(EL3) && CPTR_EL3.TTA == '1' then
        if Halted() && EDSCR.SDD == '1' then
            UNDEFINED;
        else
            AArch64.SystemAccessTrap(EL3, 0x18);
    else
        TRCVMIDCCTLR1 = X[t, 64];
elsif PSTATE.EL == EL2 then
    if Halted() && HaveEL(EL3) && EDSCR.SDD == '1'
&& boolean IMPLEMENTATION_DEFINED "EL3 trap priority
when SDD == '1'" && CPTR_EL3.TTA == '1' then
        UNDEFINED;
    elsif CPTR_EL2.TTA == '1' then
        AArch64.SystemAccessTrap(EL2, 0x18);
    elsif HaveEL(EL3) && CPTR_EL3.TTA == '1' then
        if Halted() && EDSCR.SDD == '1' then
            UNDEFINED;
        else
            AArch64.SystemAccessTrap(EL3, 0x18);
    else
        TRCVMIDCCTLR1 = X[t, 64];
elsif PSTATE.EL == EL3 then
    if CPTR_EL3.TTA == '1' then
        AArch64.SystemAccessTrap(EL3, 0x18);
    else
        TRCVMIDCCTLR1 = X[t, 64];
```

28/03/2023 16:02; 72747e43966d6b97dcbd230a1b3f0421d1ea3d94