

## TRCRSCTLR<n>, Resource Selection Control Register <n>, n = 2 - 31

The TRCRSCTLR<n> characteristics are:

### Purpose

Controls the selection of the resources in the trace unit.

### Configuration

AArch64 System register TRCRSCTLR<n> bits [31:0] are architecturally mapped to External register [TRCRSCTLR<n>\[31:0\]](#).

This register is present only when FEAT\_ETE is implemented, FEAT\_TRC\_SR is implemented and  $(\text{UInt}(\text{TRCIDR4.NUMRSPAIR}) + 1) * 2 > n$ . Otherwise, direct accesses to TRCRSCTLR<n> are undefined.

Resource selector 0 always returns FALSE.

Resource selector 1 always returns TRUE.

Resource selectors are implemented in pairs. Each odd numbered resource selector is part of a pair with the even numbered resource selector that is numbered as one less than it. For example, resource selectors 2 and 3 form a pair.

### Attributes

TRCRSCTLR<n> is a 64-bit register.

### Field descriptions

63	62	61	60	59	58	57	56	55	54	53	52	51	50	49	48	47	46	45	44	43	42	41	40	39	38	37	36	35	34	33	32
RES0																															
RES0										PAIRINV		INV		GROUP								SELECT									
31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0

### Bits [63:22]

Reserved, res0.

**PAIRINV, bit [21]****When  $n \bmod 2 == 0$ :**

Controls whether the combined result from a resource selector pair is inverted.

<b>PAIRINV</b>	<b>Meaning</b>
0b0	Do not invert the combined output of the 2 resource selectors.
0b1	Invert the combined output of the 2 resource selectors.

If:

- A is the register TRCRSCTLR<n>.
- B is the register TRCRSCTLR<n+1>.

Then the combined output of the 2 resource selectors A and B depends on the value of (A.PAIRINV, A.INV, B.INV) as follows:

- 0b000 -> A and B.
- 0b001 -> Reserved.
- 0b010 -> not(A) and B.
- 0b011 -> not(A) and not(B).
- 0b100 -> not(A) or not(B).
- 0b101 -> not(A) or B.
- 0b110 -> Reserved.
- 0b111 -> A or B.

The reset behavior of this field is:

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

**Otherwise:**

Reserved, res0.

**INV, bit [20]**

Controls whether the resource, that TRCRSCTLR<n>.GROUP and TRCRSCTLR<n>.SELECT selects, is inverted.

<b>INV</b>	<b>Meaning</b>
0b0	Do not invert the output of this selector.
0b1	Invert the output of this selector.

The reset behavior of this field is:

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

#### **GROUP, bits [19:16]**

Selects a group of resources.

<b>GROUP</b>	<b>Meaning</b>	<b>SELECT</b>
0b0000	External Input Selectors.	<a href="#">SELECT encoding for External Input Selectors</a>
0b0001	PE Comparator Inputs.	<a href="#">SELECT encoding for PE Comparator Inputs</a>
0b0010	Counters and Sequencer.	<a href="#">SELECT encoding for Counters and Sequencer</a>
0b0011	Single-shot Comparator Controls.	<a href="#">SELECT encoding for Single-shot Comparator Controls</a>
0b0100	Single Address Comparators.	<a href="#">SELECT encoding for Single Address Comparators</a>
0b0101	Address Range Comparators.	<a href="#">SELECT encoding for Address Range Comparators</a>
0b0110	Context Identifier Comparators.	<a href="#">SELECT encoding for Context Identifier Comparators</a>
0b0111	Virtual Context Identifier Comparators.	<a href="#">SELECT encoding for Virtual Context Identifier Comparators</a>

All other values are reserved.

The reset behavior of this field is:

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

## **SELECT, bits [15:0]**

Resource Specific Controls. Contains the controls specific to the resource group selected by GROUP, described in the following sections.

### **SELECT encoding for External Input Selectors**

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
RES0												EXTIN[3]	EXTIN[2]	EXTIN[1]	EXTIN[0]

#### **Bits [15:4]**

Reserved, res0.

#### **EXTIN[<m>], bit [m], for m = 3 to 0**

Selects one or more External Inputs.

EXTIN[<m>]	Meaning
0b0	Ignore EXTIN <m>.
0b1	Select EXTIN <m>.

This bit is res0 if m >= [TRCIDR5](#).NUMEXTINSEL.

The reset behavior of this field is:

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

### **SELECT encoding for PE Comparator Inputs**

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
RES0								PECOMP[7]	PECOMP[6]	PECOMP[5]	PECOMP[4]	PECOMP[3]	PECOMP[2]	PECOMP[1]	PECOMP[0]

#### **Bits [15:8]**

Reserved, res0.

#### **PECOMP[<m>], bit [m], for m = 7 to 0**

Selects one or more PE Comparator Inputs.

PECOMP[<m>]	Meaning
0b0	Ignore PE Comparator Input <m>.
0b1	Select PE Comparator Input <m>.

This bit is res0 if  $m \geq \text{TRCIDR4.NUMPC}$ .

The reset behavior of this field is:

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

## SELECT encoding for Counters and Sequencer

15141312111098	7	6	5	4	3
RES0	SEQUENCER[3]	SEQUENCER[2]	SEQUENCER[1]	SEQUENCER[0]	COUNTERS[3]

### Bits [15:8]

Reserved, res0.

### SEQUENCER[<m>], bit [m+4], for $m = 3$ to 0

Sequencer states.

SEQUENCER[<m>]	Meaning
0b0	Ignore Sequencer state <m>.
0b1	Select Sequencer state <m>.

This bit is res0 if  $m \geq \text{TRCIDR5.NUMSEQSTATE}$ .

The reset behavior of this field is:

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

### COUNTERS[<m>], bit [m], for $m = 3$ to 0

Counters resources at zero.

COUNTERS[<m>]	Meaning
---------------	---------

0b0	Ignore Counter <m>.
0b1	Select Counter <m> is zero.

This bit is res0 if m >= [TRCIDR5](#).NUMCNTR.

The reset behavior of this field is:

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

## SELECT encoding for Single-shot Comparator Controls

15141312111098	7	6	5	4
RES0	SINGLE_SHOT[7]	SINGLE_SHOT[6]	SINGLE_SHOT[5]	SINGLE_SHOT[4]

### Bits [15:8]

Reserved, res0.

### SINGLE\_SHOT[<m>], bit [m], for m = 7 to 0

Selects one or more Single-shot Comparator Controls.

SINGLE_SHOT[<m>]	Meaning
0b0	Ignore Single-shot Comparator Control <m>.
0b1	Select Single-shot Comparator Control <m>.

This bit is res0 if m >= [TRCIDR4](#).NUMSSCC.

The reset behavior of this field is:

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

## SELECT encoding for Single Address Comparators

15	14	13	12	11	10	9	8	7	6	5	4
SAC[15]	SAC[14]	SAC[13]	SAC[12]	SAC[11]	SAC[10]	SAC[9]	SAC[8]	SAC[7]	SAC[6]	SAC[5]	SAC[4]

### SAC[<m>], bit [m], for m = 15 to 0

Selects one or more Single Address Comparators.

SAC[<m>]	Meaning
0b0	Ignore Single Address Comparator <m>.
0b1	Select Single Address Comparator <m>.

This bit is res0 if  $m \geq 2 \text{ } \tilde{\text{A}} \text{---}$  [TRCIDR4](#).NUMACPAIRS.

The reset behavior of this field is:

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

## SELECT encoding for Address Range Comparators

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
RES0								ARC[7]	ARC[6]	ARC[5]	ARC[4]	ARC[3]	ARC[2]	ARC[1]	ARC[0]

### Bits [15:8]

Reserved, res0.

### ARC[<m>], bit [m], for m = 7 to 0

Selects one or more Address Range Comparators.

ARC[<m>]	Meaning
0b0	Ignore Address Range Comparator <m>.
0b1	Select Address Range Comparator <m>.

This bit is res0 if  $m \geq$  [TRCIDR4](#).NUMACPAIRS.

The reset behavior of this field is:

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

## SELECT encoding for Context Identifier Comparators

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
RES0								CID[7]	CID[6]	CID[5]	CID[4]	CID[3]	CID[2]	CID[1]	CID[0]

### Bits [15:8]

Reserved, res0.

### CID[<m>], bit [m], for m = 7 to 0

Selects one or more Context Identifier Comparators.

CID[<m>]	Meaning
0b0	Ignore Context Identifier Comparator <m>.
0b1	Select Context Identifier Comparator <m>.

This bit is res0 if m >= [TRCIDR4](#).NUMCIDC.

The reset behavior of this field is:

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

## SELECT encoding for Virtual Context Identifier Comparators

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
RES0								VMID[7]	VMID[6]	VMID[5]	VMID[4]	VMID[3]	VMID[2]	VMID[1]	VMID[0]

### Bits [15:8]

Reserved, res0.

### VMID[<m>], bit [m], for m = 7 to 0

Selects one or more Virtual Context Identifier Comparators.

VMID[<m>]	Meaning
0b0	Ignore Virtual Context Identifier Comparator <m>.



0b1	Select Virtual Context Identifier Comparator <m>.
-----	--

---

This bit is res0 if m >= [TRCIDR4](#).NUMVMIDC.

The reset behavior of this field is:

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

## Accessing TRCRSCTLR<n>

Must be programmed if any of the following are true:

- [TRCCNTCTLR<a>](#).RLDEVENT.TYPE == 0 and [TRCCNTCTLR<a>](#).RLDEVENT.SEL == n.
- [TRCCNTCTLR<a>](#).RLDEVENT.TYPE == 1 and [TRCCNTCTLR<a>](#).RLDEVENT.SEL == n/2.
- [TRCCNTCTLR<a>](#).CNTEVENT.TYPE == 0 and [TRCCNTCTLR<a>](#).CNTEVENT.SEL == n.
- [TRCCNTCTLR<a>](#).CNTEVENT.TYPE == 1 and [TRCCNTCTLR<a>](#).CNTEVENT.SEL == n/2.
- [TRCEVENTCTL0R](#).EVENT0.TYPE == 0 and [TRCEVENTCTL0R](#).EVENT0.SEL == n.
- [TRCEVENTCTL0R](#).EVENT0.TYPE == 1 and [TRCEVENTCTL0R](#).EVENT0.SEL == n/2.
- [TRCEVENTCTL0R](#).EVENT1.TYPE == 0 and [TRCEVENTCTL0R](#).EVENT1.SEL == n.
- [TRCEVENTCTL0R](#).EVENT1.TYPE == 1 and [TRCEVENTCTL0R](#).EVENT1.SEL == n/2.
- [TRCEVENTCTL0R](#).EVENT2.TYPE == 0 and [TRCEVENTCTL0R](#).EVENT2.SEL == n.
- [TRCEVENTCTL0R](#).EVENT2.TYPE == 1 and [TRCEVENTCTL0R](#).EVENT2.SEL == n/2.
- [TRCEVENTCTL0R](#).EVENT3.TYPE == 0 and [TRCEVENTCTL0R](#).EVENT3.SEL == n.
- [TRCEVENTCTL0R](#).EVENT3.TYPE == 1 and [TRCEVENTCTL0R](#).EVENT3.SEL == n/2.
- [TRCSEQEVR<a>](#).B.TYPE == 0 and [TRCSEQEVR<a>](#).B.SEL = n.
- [TRCSEQEVR<a>](#).B.TYPE == 1 and [TRCSEQEVR<a>](#).B.SEL = n/2.
- [TRCSEQEVR<a>](#).F.TYPE == 0 and [TRCSEQEVR<a>](#).F.SEL = n.
- [TRCSEQEVR<a>](#).F.TYPE == 1 and [TRCSEQEVR<a>](#).F.SEL = n/2.
- [TRCSEQRSTEVR](#).RST.TYPE == 0 and [TRCSEQRSTEVR](#).RST.SEL == n.
- [TRCSEQRSTEVR](#).RST.TYPE == 1 and [TRCSEQRSTEVR](#).RST.SEL == n/2.
- [TRCTSCTLR](#).EVENT.TYPE == 0 and [TRCTSCTLR](#).EVENT.SEL == n.
- [TRCTSCTLR](#).EVENT.TYPE == 1 and [TRCTSCTLR](#).EVENT.SEL == n/2.
- [TRCVICTLR](#).EVENT.TYPE == 0 and [TRCVICTLR](#).EVENT.SEL == n.

- [TRCVICTLR](#).EVENT.TYPE == 1 and [TRCVICTLR](#).EVENT.SEL == n/2.

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>, TRCRSCTLR<m> ; Where m = 2-31

op0	op1	CRn	CRm	op2
0b10	0b001	0b0001	m[3:0]	0b00:m[4]

```
integer m = UInt(op2<0>:CRm<3:0>);

if m >= NUM_TRACE_RESOURCE_SELECTOR_PAIRS * 2 then
    UNDEFINED;
elseif PSTATE.EL == EL0 then
    UNDEFINED;
elseif 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;
    elseif CPACR_EL1.TTA == '1' then
        AArch64.SystemAccessTrap(EL1, 0x18);
    elseif EL2Enabled() && CPTR_EL2.TTA == '1' then
        AArch64.SystemAccessTrap(EL2, 0x18);
    elseif EL2Enabled() &&
    IsFeatureImplemented(FEAT_FGT) && (!HaveEL(EL3) ||
    SCR_EL3.FGTEn == '1') && HDFGRTR_EL2.TRC == '1' then
        AArch64.SystemAccessTrap(EL2, 0x18);
    elseif HaveEL(EL3) && CPTR_EL3.TTA == '1' then
        if Halted() && EDSCR.SDD == '1' then
            UNDEFINED;
        else
            AArch64.SystemAccessTrap(EL3, 0x18);
    else
        X[t, 64] = TRCRSCTLR[m];
elseif 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;
    elseif CPTR_EL2.TTA == '1' then
        AArch64.SystemAccessTrap(EL2, 0x18);
    elseif HaveEL(EL3) && CPTR_EL3.TTA == '1' then
        if Halted() && EDSCR.SDD == '1' then
            UNDEFINED;
        else
            AArch64.SystemAccessTrap(EL3, 0x18);
    else
        X[t, 64] = TRCRSCTLR[m];
```

```

elseif PSTATE.EL == EL3 then
    if CPTR_EL3.TTA == '1' then
        AArch64.SystemAccessTrap(EL3, 0x18);
    else
        X[t, 64] = TRCRSCTLR[m];

```

MSR TRCRSCTLR<m>, <Xt> ; Where m = 2-31

op0	op1	CRn	CRm	op2
0b10	0b001	0b0001	m[3:0]	0b00:m[4]

```

integer m = UInt(op2<0>:CRm<3:0>);

if m >= NUM_TRACE_RESOURCE_SELECTOR_PAIRS * 2 then
    UNDEFINED;
elseif PSTATE.EL == EL0 then
    UNDEFINED;
elseif 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;
    elseif CPACR_EL1.TTA == '1' then
        AArch64.SystemAccessTrap(EL1, 0x18);
    elseif EL2Enabled() && CPTR_EL2.TTA == '1' then
        AArch64.SystemAccessTrap(EL2, 0x18);
    elseif EL2Enabled() &&
    IsFeatureImplemented(FEAT_FGT) && (!HaveEL(EL3) ||
    SCR_EL3.FGTEn == '1') && HDFGWTR_EL2.TRC == '1' then
        AArch64.SystemAccessTrap(EL2, 0x18);
    elseif HaveEL(EL3) && CPTR_EL3.TTA == '1' then
        if Halted() && EDSCR.SDD == '1' then
            UNDEFINED;
        else
            AArch64.SystemAccessTrap(EL3, 0x18);
    else
        TRCRSCTLR[m] = X[t, 64];
elseif 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;
    elseif CPTR_EL2.TTA == '1' then
        AArch64.SystemAccessTrap(EL2, 0x18);
    elseif HaveEL(EL3) && CPTR_EL3.TTA == '1' then
        if Halted() && EDSCR.SDD == '1' then
            UNDEFINED;
        else
            AArch64.SystemAccessTrap(EL3, 0x18);
    else
        TRCRSCTLR[m] = X[t, 64];
elseif PSTATE.EL == EL3 then
    if CPTR_EL3.TTA == '1' then

```

```
    AArch64.SystemAccessTrap(EL3, 0x18);  
else  
    TRCRSCTLR[m] = X[t, 64];
```

---

[AArch32  
Registers](#)[AArch64  
Registers](#)[AArch32  
Instructions](#)[AArch64  
Instructions](#)[Index by  
Encoding](#)[External  
Registers](#)

28/03/2023 16:02; 72747e43966d6b97dcbd230a1b3f0421d1ea3d94

Copyright Â© 2010-2023 Arm Limited or its affiliates. All rights reserved. This document is Non-Confidential.