

ICC_SGI1R_EL1, Interrupt Controller Software Generated Interrupt Group 1 Register

The ICC_SGI1R_EL1 characteristics are:

Purpose

Generates Group 1 SGIs for the current Security state.

Configuration

AArch64 System register ICC_SGI1R_EL1 performs the same function as AArch32 System register [ICC_SGI1R](#).

This register is present only when FEAT_GICv3 is implemented. Otherwise, direct accesses to ICC_SGI1R_EL1 are undefined.

Under certain conditions a write to ICC_SGI1R_EL1 can generate Group 0 interrupts, see 'Forwarding an SGI to a target PE' in ARM® Generic Interrupt Controller Architecture Specification, GIC architecture version 3.0 and version 4.0 (ARM IHI 0069).

Attributes

ICC_SGI1R_EL1 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								Aff3								RS				RES0		IRM		Aff2							
RES0				INTID				Aff1								TargetList															
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 [63:56]

Reserved, res0.

Aff3, bits [55:48]

The affinity 3 value of the affinity path of the cluster for which SGI interrupts will be generated.

If the IRM bit is 1, this field is res0.

RS, bits [47:44]

RangeSelector

Controls which group of 16 values is represented by the TargetList field.

TargetList[n] represents aff0 value $((RS * 16) + n)$.

When [ICC_CTLR_EL1](#).RSS==0, RS is res0.

When [ICC_CTLR_EL1](#).RSS==1 and [GICD_TYPER](#).RSS==0, writing this register with $RS \neq 0$ is a constrained unpredictable choice of :

- The write is ignored.
- The RS field is treated as 0.

Bits [43:41]

Reserved, res0.

IRM, bit [40]

Interrupt Routing Mode. Determines how the generated interrupts are distributed to PEs. Possible values are:

IRM	Meaning
0b0	Interrupts routed to the PEs specified by Aff3.Aff2.Aff1.<target list>.
0b1	Interrupts routed to all PEs in the system, excluding "self".

Aff2, bits [39:32]

The affinity 2 value of the affinity path of the cluster for which SGI interrupts will be generated.

If the IRM bit is 1, this field is res0.

Bits [31:28]

Reserved, res0.

INTID, bits [27:24]

The INTID of the SGI.

Aff1, bits [23:16]

The affinity 1 value of the affinity path of the cluster for which SGI interrupts will be generated.

If the IRM bit is 1, this field is res0.

TargetList, bits [15:0]

Target List. The set of PEs for which SGI interrupts will be generated. Each bit corresponds to the PE within a cluster with an Affinity 0 value equal to the bit number.

If a bit is 1 and the bit does not correspond to a valid target PE, the bit must be ignored by the Distributor. It is implementation defined whether, in such cases, a Distributor can signal a system error.

Note

If SRE is set only for Secure EL3, software executing at EL3 might use the System register interface to generate SGIs. Therefore, the Distributor must always be able to receive and acknowledge Generate SGI packets received from CPU interface regardless of the ARE settings for a Security state. However, the Distributor might discard such packets.

If the IRM bit is 1, this field is res0.

Accessing ICC_SGI1R_EL1

Note

Accesses at EL3 are treated as Secure regardless of the value of SCR_EL3.NS.

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

MSR ICC_SGI1R_EL1, <Xt>

op0	op1	CRn	CRm	op2
0b11	0b000	0b1100	0b1011	0b101

```
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' && SCR_EL3.<IRQ,FIQ> == '11' then
    UNDEFINED;
elseif ICC_SRE_EL1.SRE == '0' then
    AArch64.SystemAccessTrap(EL1, 0x18);
elseif EL2Enabled() && ICH_HCR_EL2.TC == '1' then
    AArch64.SystemAccessTrap(EL2, 0x18);
elseif EL2Enabled() && HCR_EL2.FMO == '1' then
    AArch64.SystemAccessTrap(EL2, 0x18);
elseif EL2Enabled() && HCR_EL2.IMO == '1' then
    AArch64.SystemAccessTrap(EL2, 0x18);
elseif HaveEL(EL3) && SCR_EL3.<IRQ,FIQ> == '11'
then
    if Halted() && EDSCR.SDD == '1' then
        UNDEFINED;
    else
        AArch64.SystemAccessTrap(EL3, 0x18);
    else
        ICC_SGI1R_EL1 = X[t, 64];
elseif PSTATE.EL == EL2 then
    if Halted() && HaveEL(EL3) && EDSCR.SDD == '1'
    && boolean IMPLEMENTATION_DEFINED "EL3 trap priority
when SDD == '1' && SCR_EL3.<IRQ,FIQ> == '11' then
        UNDEFINED;
    elseif ICC_SRE_EL2.SRE == '0' then
        AArch64.SystemAccessTrap(EL2, 0x18);
    elseif HaveEL(EL3) && SCR_EL3.<IRQ,FIQ> == '11'
then
        if Halted() && EDSCR.SDD == '1' then
            UNDEFINED;
        else
            AArch64.SystemAccessTrap(EL3, 0x18);
        else
            ICC_SGI1R_EL1 = X[t, 64];
elseif PSTATE.EL == EL3 then
    if ICC_SRE_EL3.SRE == '0' then
        AArch64.SystemAccessTrap(EL3, 0x18);
    else
        ICC_SGI1R_EL1 = X[t, 64];

```

[AArch32
Registers](#)

[AArch64
Registers](#)

[AArch32
Instructions](#)

[AArch64
Instructions](#)

[Index by
Encoding](#)

[External
Registers](#)

28/03/2023 16:01; 72747e43966d6b97dcbdd230a1b3f0421d1ea3d94

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