GICD_NSACR<n>, Non-secure Access Control Registers, n = 0 - 63

The GICD_NSACR<n> characteristics are:

Purpose

Enables Secure software to permit Non-secure software on a particular PE to create and control Group 0 interrupts.

Configuration

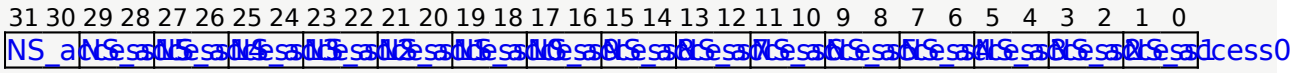
The concept of selective enabling of Non-secure access to Group 0 and Secure Group 1 interrupts applies to SGIs and SPIs.

GICD_NSACR0 is a Banked register used for SGIs. A copy is provided for every PE that has a CPU interface and that supports this feature.

Attributes

GICD_NSACR<n> is a 32-bit register.

Field descriptions



NS_access<x>, bits [2x+1:2x], for x = 15 to 0

Controls Non-secure access of the interrupt with ID 16n + x.

If the corresponding interrupt does not support configurable Non-secure access, the field is RAZ/WI.

Otherwise, the field is RW and determines the level of Non-secure control permitted if the interrupt is a Secure interrupt. If the interrupt is a Non-secure interrupt, this field is ignored.

The possible values of each 2-bit field are:

NS_access<x>	Meaning
0b00	No Non-secure access is permitted to fields associated with the corresponding interrupt.

`0b01`	Non-secure read and write access is permitted to set-pending bits in GICD_ISPENDR<n> associated with the corresponding interrupt. A Non-secure write access to GICD_SETSPI_NSR is permitted to set the pending state of the corresponding interrupt. A Non-secure write access to GICD_SGIR is permitted to generate a Secure SGI for the corresponding interrupt. An implementation might also provide read access to clear-pending bits in GICD_ICPENDR<n> associated with the corresponding interrupt.
`0b10`	As `0b01`, but adds Non-secure read and write access permission to fields associated with the corresponding interrupt in the GICD_ICPENDR<n> registers. A Non-secure write access to GICD_CLRSPI_NSR is permitted to clear the pending state of the corresponding interrupt. Also adds Non-secure read access permission to fields associated with the corresponding interrupt in the GICD_ISACTIVER<n> and GICD_ICACTIVER<n> registers.

0b11	For GICD_NSACR0 this encoding is reserved and treated as 10. For all other GICD_NSACR<n> registers this encoding is treated as 0b10, but adds Non-secure read and write access permission to GICD_ITARGETSR<n> and GICD_IROUTER<n> fields associated with the corresponding interrupt.

The reset behavior of this field is:

- On a GIC reset, this field resets to 0.

For interrupt ID m, when DIV and MOD are the integer division and modulo operations:

- The corresponding GICD_NSACR<n> number, n, is given by n = m DIV 16.
- The offset of the required GICD_NSACR<n> register is (0xE00 + (4*n)).

Note

Because each field in this register comprises two bits, GICD_NSACR0 controls access rights to SGI registers, GICD_NSACR1 controls access to PPI registers (and is always RAZ/WI), and all other GICD_NSACR<n> registers control access to SPI registers.

For compatibility with GICv2, writes to GICD_NSACR0 for a particular PE must be coordinated within the Distributor and must update GICR_NSACR for the Redistributor associated with that PE.

Accessing GICD_NSACR<n>

These registers are always used when affinity routing is not enabled. When affinity routing is enabled for the Secure state, GICD_NSACR0 is res0 and GICR_NSACR provides equivalent functionality for SGIs.

These registers do not support PPIs, therefore GICD_NSACR1 is RAZ/WI.

GICD_NSACR<n> can be accessed through the memory-mapped interfaces:

Component	Frame	Offset	Instance
GIC Distributor	Dist_base	`0x0E00` + (4 * n)	GICD_NSACR<n>

This interface is accessible as follows:

- When GICD_CTLR.DS == 1, accesses to this register are **RAZ/WI**.
- When GICD_CTLR.DS == 0 and an access is Secure, accesses to this register are **RW**.
- When GICD_CTLR.DS == 0 and an access is Non-secure, accesses to this register are **RAZ/WI**.
- When GICD_CTLR.DS == 0, FEAT_RME is implemented and an access is Root, accesses to this register are **RW**.
- When GICD_CTLR.DS == 0, FEAT_RME is implemented and an access is Realm, accesses to this register are **RAZ/WI**.

28/03/2023 16:02; 72747e43966d6b97dcbd230a1b3f0421d1ea3d94