

GICD_SPENDSGIR<n>, SGI Set-Pending Registers, n = 0 - 3

The GICD_SPENDSGIR<n> characteristics are:

Purpose

Adds the pending state to an SGI.

A write to this register changes the state of an inactive SGI to pending, and the state of an active SGI to active and pending.

Configuration

Four SGI set-pending registers are implemented. Each register contains eight set-pending bits for each of four SGIs, for a total of 16 possible SGIs.

In multiprocessor implementations, each PE has a copy of these registers.

Attributes

GICD_SPENDSGIR<n> is a 32-bit register.

Field descriptions

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
SGI_set_pending_bits								SGI_set_pending_bits								SGI_set_pending_bits								SGI_set_pending_bits							

SGI_set_pending_bits<x>, bits [8x+7:8x], for x = 3 to 0

Adds the pending state to SGI number $4n + x$ for the PE corresponding to the bit number written to.

Reads and writes have the following behavior:

SGI_set_pending_bits<x>	Meaning
-------------------------	---------

0x00	If read, indicates that the SGI from the corresponding PE is not pending and is not active and pending. If written, has no effect.
0x01	If read, indicates that the SGI from the corresponding PE is pending or is active and pending. If written, adds the pending state to the SGI for the corresponding PE.

The reset behavior of this field is:

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

For SGI ID m , generated by processing element C writing to the corresponding [GICD_SGIR](#) field, where DIV and MOD are the integer division and modulo operations:

- The corresponding GICD_SPENDSGIR< n > number is given by $n = m \text{ DIV } 4$.
- The offset of the required register is $(0xF20 + (4n))$.
- The offset of the required field within the register GICD_SPENDSGIR< n > is given by $m \text{ MOD } 4$.
- The required bit in the 8-bit SGI set-pending field m is bit C .

Accessing GICD_SPENDSGIR< n >

These registers are used only when affinity routing is not enabled. When affinity routing is enabled for the Security state of an interrupt then the bit associated with SGI in that Security state is res0. An implementation is permitted to make the register RAZ/WI in this case.

A register bit that corresponds to an unimplemented SGI is RAZ/WI.

These registers are byte-accessible.

If the GIC implementation supports two Security states:

- A register bit that corresponds to a Group 0 interrupt is RAZ/WI to Non-secure accesses.
- Register bits corresponding to unimplemented PEs are RAZ/WI.

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

Component	Frame	Offset	Instance
GIC Distributor	Dist_base	0x0F20 + (4 * n)	GICD_SPENDSGIR<n>

Accesses on this interface are **RW**.