PMOVSCLR_EL0, Performance Monitors Overflow Flag Status Clear register

The PMOVSCLR_EL0 characteristics are:

Purpose

Contains the state of the overflow bit for the Cycle Count Register, PMU.PMCCNTR_EL0, and each of the implemented event counters PMEVCNTR<n>. Writing to this register clears these bits.

Configuration

External register PMOVSCLR_EL0 bits [31:0] are architecturally mapped to AArch64 System register PMOVSCLR_EL0[31:0] when FEAT_PMUv3_EXT32 is implemented, FEAT_PMUv3p9 is not implemented and FEAT_PMUv3_ICNTR is not implemented.

External register PMOVSCLR_EL0 bits [31:0] are architecturally mapped to AArch64 System register PMOVSSET_EL0[31:0] when FEAT_PMUv3_EXT32 is implemented, FEAT_PMUv3p9 is not implemented and FEAT_PMUv3_ICNTR is not implemented.

External register PMOVSCLR_EL0 bits [63:0] are architecturally mapped to AArch64 System register PMOVSCLR_EL0[63:0] when FEAT_PMUv3_EXT64 is implemented, or FEAT_PMUv3p9 is implemented or FEAT_PMUv3_ICNTR is implemented.

External register PMOVSCLR_EL0 bits [63:0] are architecturally mapped to AArch64 System register PMOVSSET_EL0[63:0] when FEAT_PMUv3_EXT64 is implemented, or FEAT_PMUv3p9 is implemented or FEAT_PMUv3_ICNTR is implemented.

External register PMOVSCLR_EL0 bits [31:0] are architecturally mapped to AArch32 System register PMOVSSET[31:0].

External register PMOVSCLR_EL0 bits [31:0] are architecturally mapped to AArch32 System register PMOVSR[31:0].

This register is present only when FEAT_PMUv3_EXT is implemented. Otherwise, direct accesses to PMOVSCLR_EL0 are res0.

PMOVSCLR_EL0 is in the Core power domain.

Attributes

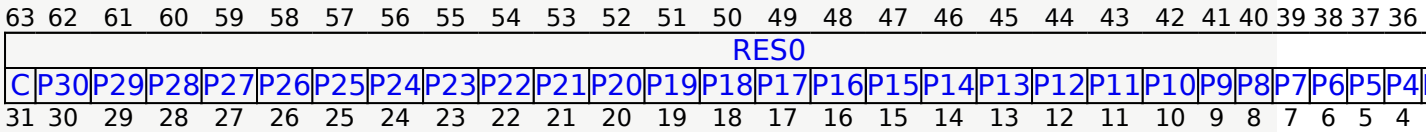
PMOVSCLR_EL0 is a:

- 64-bit register when FEAT_PMUv3_EXT64 is implemented, or FEAT_PMUv3p9 is implemented or FEAT_PMUv3_ICNTR is implemented
- 32-bit register otherwise

This register is part of the PMU block.

Field descriptions

When FEAT_PMUv3_EXT64 is implemented, or FEAT_PMUv3p9 is implemented or FEAT_PMUv3_ICNTR is implemented:



Bits [63:33]

Reserved, res0.

F0, bit [32]
When FEAT_PMUv3_ICNTR is implemented:

Unsigned overflow flag for PMU.PMICNTR_EL0 clear. On writes, allows software to clear the unsigned overflow flag for PMU.PMICNTR_EL0 to 0. On reads, returns the unsigned overflow flag for PMU.PMICNTR_EL0.

F0	Meaning
0b0	PMU.PMICNTR_EL0 has not overflowed.
0b1	PMU.PMICNTR_EL0 has overflowed.

The reset behavior of this field is:

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

Accessing this field has the following behavior:

- When SoftwareLockStatus(), access to this field is **RO**.
- Otherwise, access to this field is **W1C**.

Otherwise:

Reserved, res0.

C, bit [31]

Unsigned overflow flag for PMU.PMCCNTR_EL0 clear. On writes, allows software to clear the unsigned overflow flag for PMU.PMCCNTR_EL0 to 0. On reads, returns the unsigned overflow flag for PMU.PMCCNTR_EL0 overflow status.

C	Meaning
0b0	PMU.PMCCNTR_EL0 has not overflowed.
0b1	PMU.PMCCNTR_EL0 has overflowed.

PMU.PMCR_EL0.LC controls whether an overflow is detected from unsigned overflow of PMU.PMCCNTR_EL0[31:0] or unsigned overflow of PMU.PMCCNTR_EL0[63:0].

The reset behavior of this field is:

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

Accessing this field has the following behavior:

- When SoftwareLockStatus(), access to this field is **RO**.
- Otherwise, access to this field is **W1C**.

P<m>, bit [m], for m = 30 to 0

Unsigned overflow flag for PMEVCNTR<m>_EL0 clear. On writes, allows software to clear the unsigned overflow flag for PMEVCNTR<m>_EL0 to 0. On reads, returns the unsigned overflow flag for PMEVCNTR<m>_EL0 overflow status.

P<m>	Meaning
0b0	PMEVCNTR<m>_EL0 has not overflowed.
0b1	PMEVCNTR<m>_EL0 has overflowed.

If FEAT_PMUv3p5 is implemented, PMU.MDCR_EL2.HLP and PMU.PMCR_EL0.LP control whether an overflow is detected from unsigned overflow of PMU.PMEVCNTR<n>_EL0[31:0] or unsigned overflow of PMU.PMEVCNTR<n>_EL0[63:0].

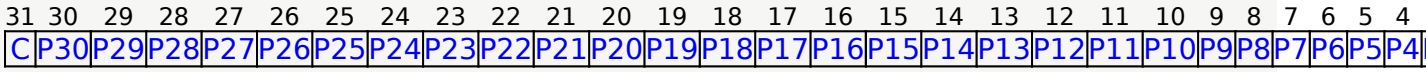
The reset behavior of this field is:

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

Accessing this field has the following behavior:

- When m >= NUM_PMU_COUNTERS, access to this field is **RAZ/WI**.
- When SoftwareLockStatus(), access to this field is **RO**.
- Otherwise, access to this field is **W1C**.

Otherwise:



C, bit [31]

Unsigned overflow flag for PMU.PMCCNTR_EL0 clear. On writes, allows software to clear the unsigned overflow flag for PMU.PMCCNTR_EL0 to 0. On reads, returns the unsigned overflow flag for PMU.PMCCNTR_EL0 overflow status.

C	Meaning
0b0	PMU.PMCCNTR_EL0 has not overflowed.
0b1	PMU.PMCCNTR_EL0 has overflowed.

PMU.PMCR_EL0.LC controls whether an overflow is detected from unsigned overflow of PMU.PMCCNTR_EL0[31:0] or unsigned overflow of PMU.PMCCNTR_EL0[63:0].

The reset behavior of this field is:

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

Accessing this field has the following behavior:

- When SoftwareLockStatus(), access to this field is **RO**.
- Otherwise, access to this field is **W1C**.

P<m>, bit [m], for m = 30 to 0

Unsigned overflow flag for PMEVCNTR<m>_EL0 clear. On writes, allows software to clear the unsigned overflow flag for PMEVCNTR<m>_EL0 to 0. On reads, returns the unsigned overflow flag for PMEVCNTR<m>_EL0 overflow status.

P<m>	Meaning

0b0	PMEVCNTR<m>_EL0 has not overflowed.
0b1	PMEVCNTR<m>_EL0 has overflowed.

If FEAT_PMUv3p5 is implemented, PMU.MDCR_EL2.HLP and PMU.PMCR_EL0.LP control whether an overflow is detected from unsigned overflow of PMU.PMEVCNTR<n>_EL0[31:0] or unsigned overflow of PMU.PMEVCNTR<n>_EL0[63:0].

The reset behavior of this field is:

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

Accessing this field has the following behavior:

- When m >= NUM_PMU_COUNTERS, access to this field is **RAZ/WI**.
- When SoftwareLockStatus(), access to this field is **RO**.
- Otherwise, access to this field is **W1C**.

Accessing PMOVSCLR_EL0

Note

SoftwareLockStatus() depends on the type of access attempted and AllowExternalPMUAccess() has a new definition from Armv8.4. Refer to the Pseudocode definitions for more information.

Accesses to this register use the following encodings:

When FEAT_PMUv3_EXT64 is implemented, or FEAT_PMUv3_ICNTR is implemented or FEAT_PMUv3p9 is implemented

[63:0] Accessible at offset 0xC80 from PMU

- When DoubleLockStatus(), or !IsCorePowered(), or OSLockStatus() or !AllowExternalPMUAccess(), accesses to this register generate an error response.
- When FEAT_PMUv3_EXT32 is implemented and SoftwareLockStatus(), accesses to this register are **RO**.
- Otherwise, accesses to this register are **RW**.

When FEAT_PMUv3_EXT32 is implemented, FEAT_PMUv3_ICNTR is not implemented and FEAT_PMUv3p9 is not implemented

[31:0] Accessible at offset 0xC80 from PMU

- When DoubleLockStatus(), or !IsCorePowered(), or OSLockStatus() or !AllowExternalPMUAccess(), accesses to this register generate an error response.
- When SoftwareLockStatus(), accesses to this register are **RO**.
- Otherwise, accesses to this register are **RW**.

28/03/2023 16:01; 72747e43966d6b97dcbd230a1b3f0421d1ea3d94