SPMCR_EL0, System Performance Monitor Control Register

The SPMCR_EL0 characteristics are:

Purpose

Main control register for System PMU <s>.

Configuration

This register is present only when FEAT_SPMU is implemented. Otherwise, direct accesses to SPMCR_EL0 are undefined.

Attributes

SPMCR_EL0 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																				TRO	HDBG	FZO	NA	RES0				EX	RES0		P	E
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 [63:12]

Reserved, res0.

TRO, bit [11]
When SPMCFGR_EL1.TRO == 1:

Trace enable. For more information on this field, see 'CoreSight PMU Architecture'.

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 PSTATE.EL == EL0, access to this field is **RO**.
- Otherwise, access to this field is **RW**.

Otherwise:

Reserved, res0.

HDBG, bit [10]
When SPMCFGR_EL1.HDBG == 1:

Halt-on-debug. For more information on this field, see 'CoreSight PMU Architecture'.

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 PSTATE.EL == EL0, access to this field is **RO**.
- Otherwise, access to this field is **RW**.

Otherwise:

Reserved, res0.

FZO, bit [9]
When SPMCFGR_EL1.FZO == 1:

Freeze-on-overflow. For more information on this field, see 'CoreSight PMU Architecture'.

Note that, if implemented by a System PMU, then freeze-on-overflow affects only the counters of System PMU <s>, not other System PMUs nor the PE PMU.

The reset behavior of this field is:

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

Otherwise:

Reserved, res0.

NA, bit [8]
When SPMCFGR_EL1.NA == 1:

Not accessible. For more information on this field, see 'CoreSight PMU Architecture'.

The reset behavior of this field is:

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

Access to this field is **RO**.

Otherwise:

Reserved, res0.

Bits [7:5]

Reserved, res0.

EX, bit [4]
When SPMCFGR_EL1.EX == 1:

Export enable. For more information on this field, see 'CoreSight PMU Architecture'.

The reset behavior of this field is:

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

Otherwise:

Reserved, res0.

Bits [3:2]

Reserved, res0.

P, bit [1]

Event counter reset.

P	Meaning
0b0	Write is ignored.
0b1	Reset all event counters in System PMU <s> to zero.

Note

Resetting the event counters does not affect any overflow flags.

Access to this field is **WO/RAZ**.

E, bit [0]

Count enable. This field controls System PMU <s>.

E	Meaning
0b0	Monitor is disabled.
0b1	Monitor is enabled.

Performance monitor overflow IRQs are only signaled by System PMU <s> when this field is 1.

The reset behavior of this field is:

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

Accessing SPMCR_EL0

To access SPMCR_EL0 for System PMU <s>, set SPMSELR_EL0.SYSPMUSEL to s.

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

MRS <Xt>, SPMCR_EL0

op0	op1	CRn	CRm	op2
0b10	0b011	0b1001	0b1100	0b000

```
if PSTATE.EL == EL0 then
    if Halted() && HaveEL(EL3) && EDSCR.SDD == '1'
&& boolean IMPLEMENTATION_DEFINED "EL3 trap priority
when SDD == '1'" && MDCR_EL3.EnPM2 == '0' then
        UNDEFINED;
    elsif EL2Enabled() && HCR_EL2.<E2H,TGE> != '11'
&& IsFeatureImplemented(FEAT_FGT2) && HaveEL(EL3) &&
SCR_EL3.FGTEn2 == '0' then
        AArch64.SystemAccessTrap(EL2, 0x18);
    elsif EL2Enabled() && HCR_EL2.<E2H,TGE> != '11'
&& IsFeatureImplemented(FEAT_FGT2) &&
HDFGRTR2_EL2.nSPMCR_EL0 == '0' then
        AArch64.SystemAccessTrap(EL2, 0x18);
    elsif HaveEL(EL3) && MDCR_EL3.EnPM2 == '0' then
        if Halted() && EDSCR.SDD == '1' then
            UNDEFINED;
        else
            AArch64.SystemAccessTrap(EL3, 0x18);
    else
        X[t, 64] =
SPMCR_EL0[UInt(SPMSELR_EL0.SYSPMUSEL)];
elsif PSTATE.EL == EL1 then
```

```
    if Halted() && HaveEL(EL3) && EDSCR.SDD == '1'
&& boolean IMPLEMENTATION_DEFINED "EL3 trap priority
when SDD == '1'" && MDCR_EL3.EnPM2 == '0' then
        UNDEFINED;
    elsif EL2Enabled() &&
IsFeatureImplemented(FEAT_FGT2) && HaveEL(EL3) &&
SCR_EL3.FGTEn2 == '0' then
        AArch64.SystemAccessTrap(EL2, 0x18);
    elsif EL2Enabled() &&
IsFeatureImplemented(FEAT_FGT2) &&
HDFGRTR2_EL2.nSPMCR_EL0 == '0' then
        AArch64.SystemAccessTrap(EL2, 0x18);
    elsif HaveEL(EL3) && MDCR_EL3.EnPM2 == '0' then
        if Halted() && EDSCR.SDD == '1' then
            UNDEFINED;
        else
            AArch64.SystemAccessTrap(EL3, 0x18);
    else
        X[t, 64] =
SPMCR_EL0[UInt(SPMSELR_EL0.SYSPMUSEL)];
elsif PSTATE.EL == EL2 then
    if Halted() && HaveEL(EL3) && EDSCR.SDD == '1'
&& boolean IMPLEMENTATION_DEFINED "EL3 trap priority
when SDD == '1'" && MDCR_EL3.EnPM2 == '0' then
        UNDEFINED;
    elsif HaveEL(EL3) && MDCR_EL3.EnPM2 == '0' then
        if Halted() && EDSCR.SDD == '1' then
            UNDEFINED;
        else
            AArch64.SystemAccessTrap(EL3, 0x18);
    else
        X[t, 64] =
SPMCR_EL0[UInt(SPMSELR_EL0.SYSPMUSEL)];
elsif PSTATE.EL == EL3 then
    X[t, 64] =
SPMCR_EL0[UInt(SPMSELR_EL0.SYSPMUSEL)];
```

MSR SPMCR_EL0, <Xt>

op0	op1	CRn	CRm	op2
0b10	0b011	0b1001	0b1100	0b000

```
if PSTATE.EL == EL0 then
    if Halted() && HaveEL(EL3) && EDSCR.SDD == '1'
&& boolean IMPLEMENTATION_DEFINED "EL3 trap priority
when SDD == '1'" && MDCR_EL3.EnPM2 == '0' then
        UNDEFINED;
    elsif EL2Enabled() && HCR_EL2.<E2H,TGE> != '11'
&& IsFeatureImplemented(FEAT_FGT2) && HaveEL(EL3) &&
SCR_EL3.FGTEn2 == '0' then
        AArch64.SystemAccessTrap(EL2, 0x18);
    elsif EL2Enabled() && HCR_EL2.<E2H,TGE> != '11'
&& IsFeatureImplemented(FEAT_FGT2) &&
```

```
HDFGWTR2_EL2.nSPMCR_EL0 == '0' then
        AArch64.SystemAccessTrap(EL2, 0x18);
    elsif HaveEL(EL3) && MDCR_EL3.EnPM2 == '0' then
        if Halted() && EDSCR.SDD == '1' then
            UNDEFINED;
        else
            AArch64.SystemAccessTrap(EL3, 0x18);
    else
        SPMCR_EL0[UInt(SPMSELR_EL0.SYSPMUSEL)] =
X[t, 64];
elsif PSTATE.EL == EL1 then
    if Halted() && HaveEL(EL3) && EDSCR.SDD == '1'
&& boolean IMPLEMENTATION_DEFINED "EL3 trap priority
when SDD == '1'" && MDCR_EL3.EnPM2 == '0' then
        UNDEFINED;
    elsif EL2Enabled() &&
IsFeatureImplemented(FEAT_FGT2) && HaveEL(EL3) &&
SCR_EL3.FGTEn2 == '0' then
        AArch64.SystemAccessTrap(EL2, 0x18);
    elsif EL2Enabled() &&
IsFeatureImplemented(FEAT_FGT2) &&
HDFGWTR2_EL2.nSPMCR_EL0 == '0' then
        AArch64.SystemAccessTrap(EL2, 0x18);
    elsif HaveEL(EL3) && MDCR_EL3.EnPM2 == '0' then
        if Halted() && EDSCR.SDD == '1' then
            UNDEFINED;
        else
            AArch64.SystemAccessTrap(EL3, 0x18);
    else
        SPMCR_EL0[UInt(SPMSELR_EL0.SYSPMUSEL)] =
X[t, 64];
elsif PSTATE.EL == EL2 then
    if Halted() && HaveEL(EL3) && EDSCR.SDD == '1'
&& boolean IMPLEMENTATION_DEFINED "EL3 trap priority
when SDD == '1'" && MDCR_EL3.EnPM2 == '0' then
        UNDEFINED;
    elsif HaveEL(EL3) && MDCR_EL3.EnPM2 == '0' then
        if Halted() && EDSCR.SDD == '1' then
            UNDEFINED;
        else
            AArch64.SystemAccessTrap(EL3, 0x18);
    else
        SPMCR_EL0[UInt(SPMSELR_EL0.SYSPMUSEL)] =
X[t, 64];
elsif PSTATE.EL == EL3 then
    SPMCR_EL0[UInt(SPMSELR_EL0.SYSPMUSEL)] = X[t,
64];
```