HAFGRTR_EL2, Hypervisor Activity Monitors Fine-Grained Read Trap Register

The HAFGRTR_EL2 characteristics are:

Purpose

Provides controls for traps of MRS reads of Activity Monitors System registers.

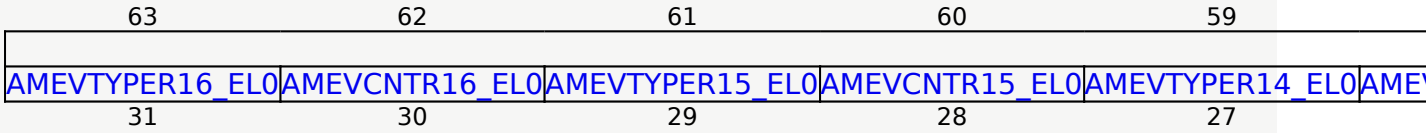
Configuration

This register is present only when FEAT_AMUv1 is implemented and FEAT_FGT is implemented. Otherwise, direct accesses to HAFGRTR_EL2 are undefined.

Attributes

HAFGRTR_EL2 is a 64-bit register.

Field descriptions



Bits [63:50]

Reserved, res0.

AMEVTYPER1<x>_EL0, bit [19+2x], for x = 15 to 0
When AMEVTYPER1<x> is implemented:

Trap MRS reads of AMEVTYPER1<x>_EL0 at EL1 and EL0 using AArch64 and MRC reads of AMEVTYPER1<x> at EL0 using AArch32 when EL1 is using AArch64 to EL2.

AMEVTYPER1<x>_EL0	Meaning

0b0	MRS reads of AMEVTYPER1<x>_EL0 at EL1 and EL0 using AArch64 and MRC reads of AMEVTYPER1<x> at EL0 using AArch32 are not trapped by this mechanism.
0b1	If EL2 is implemented and enabled in the current Security state, HCR_EL2. {E2H, TGE} != {1, 1}, EL1 is using AArch64, and either EL3 is not implemented or SCR_EL3.FGTEn == 1, then, unless the read generates a higher priority exception: • MRS reads of AMEVTYPER1<x>_EL0 at EL1 and EL0 using AArch64 are trapped to EL2 and reported with EC syndrome value 0x18. • MRC reads of AMEVTYPER1<x> at EL0 using AArch32 are trapped to EL2 and reported with EC syndrome value 0x03.

The reset behavior of this field is:

- On a Warm reset, in a system where the PE resets into EL2, this field resets to 0.

Otherwise:

Reserved, res0.

AMEVCNTR1<x>_EL0, bit [18+2x], for x = 15 to 0
When AMEVCNTR1<x> is implemented:

Trap MRS reads of AMEVCNTR1<x>_EL0 at EL1 and EL0 using AArch64 and MRC reads of AMEVCNTR1<x> at EL0 using AArch32 when EL1 is using AArch64 to EL2.

AMEVCNTR1<x>_EL0	Meaning

0b0	MRS reads of AMEVCNTR1<x>_EL0 at EL1 and EL0 using AArch64 and MRC reads of AMEVCNTR1<x> at EL0 using AArch32 are not trapped by this mechanism.
0b1	If EL2 is implemented and enabled in the current Security state, HCR_EL2. {E2H, TGE} != {1, 1}, EL1 is using AArch64, and either EL3 is not implemented or SCR_EL3.FGTEn == 1, then, unless the read generates a higher priority exception: • MRS reads of AMEVCNTR1<x>_EL0 at EL1 and EL0 using AArch64 are trapped to EL2 and reported with EC syndrome value 0x18. • MRC reads of AMEVCNTR1<x> at EL0 using AArch32 are trapped to EL2 and reported with EC syndrome value 0x03.

The reset behavior of this field is:

- On a Warm reset, in a system where the PE resets into EL2, this field resets to 0.

Otherwise:

Reserved, res0.

AMCNTEN<x>, bit [17x], for x = 1 to 0

Trap MRS reads and MRC reads of multiple System registers.

Enables a trap to EL2 the following operations:

- At EL1 and EL0 using AArch64: MRS reads of AMCNTENCLR<x>_EL0 and AMCNTENSET<x>_EL0.

- At EL0 using AArch32 when EL1 is using AArch64: MRC reads of AMCNTENCLR<x> and AMCNTENSET<x>.

AMCNTEN<x>	Meaning
0b0	The operations listed above are not trapped by this mechanism.
0b1	If EL2 is implemented and enabled in the current Security state, HCR_EL2. {E2H, TGE} != {1, 1}, EL1 is using AArch64, and either EL3 is not implemented or SCR_EL3.FGTEn == 1, then, unless the read generates a higher priority exception: • MRS reads at EL1 and EL0 using AArch64 of AMCNTENCLR<x>_EL0 and AMCNTENSET<x>_EL0 are trapped to EL2 and reported with EC syndrome value 0x18. • MRC reads at EL0 using AArch32 of AMCNTENCLR<x> and AMCNTENSET<x> are trapped to EL2 and reported with EC syndrome value 0x03.

The reset behavior of this field is:

- On a Warm reset, in a system where the PE resets into EL2, this field resets to 0.

Bits [16:5]

Reserved, res0.

AMEVCNTR0<x>_EL0, bit [x+1], for x = 3 to 0

Trap MRS reads of AMEVCNTR0<x>_EL0 at EL1 and EL0 using AArch64 and MRC reads of AMEVCNTR0<x> at EL0 using AArch32 when EL1 is using AArch64 to EL2.

AMEVCNTR0<x>_EL0	Meaning

0b0	MRS reads of AMEVCNTR0<x>_EL0 at EL1 and EL0 using AArch64 and MRC reads of AMEVCNTR0<x> at EL0 using AArch32 are not trapped by this mechanism.
0b1	If EL2 is implemented and enabled in the current Security state, HCR_EL2. {E2H, TGE} != {1, 1}, EL1 is using AArch64, and either EL3 is not implemented or SCR_EL3.FGTEn == 1, then, unless the read generates a higher priority exception: • MRS reads of AMEVCNTR0<x>_EL0 at EL1 and EL0 using AArch64 are trapped to EL2 and reported with EC syndrome value 0x18. • MRC reads of AMEVCNTR0<x> at EL0 using AArch32 are trapped to EL2 and reported with EC syndrome value 0x03.

The reset behavior of this field is:

- On a Warm reset, in a system where the PE resets into EL2, this field resets to 0.

Accessing HAFGRTR_EL2

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

```
MRS <Xt>, HAFGRTR_EL2
```

op0	op1	CRn	CRm	op2
0b11	0b100	0b0011	0b0001	0b110

```
if PSTATE.EL == EL0 then
    UNDEFINED;
elsif PSTATE.EL == EL1 then
    if EL2Enabled() && HCR_EL2.<NV2,NV> == '11' then
        X[t, 64] = NVMem[0x1E8];
    elsif EL2Enabled() && HCR_EL2.NV == '1' then
        AArch64.SystemAccessTrap(EL2, 0x18);
    else
        UNDEFINED;
elsif PSTATE.EL == EL2 then
    if Halted() && HaveEL(EL3) && EDSCR.SDD == '1'
&& boolean IMPLEMENTATION_DEFINED "EL3 trap priority
when SDD == '1'" && SCR_EL3.FGTEn == '0' then
        UNDEFINED;
    elsif HaveEL(EL3) && SCR_EL3.FGTEn == '0' then
        if Halted() && EDSCR.SDD == '1' then
            UNDEFINED;
        else
            AArch64.SystemAccessTrap(EL3, 0x18);
    else
        X[t, 64] = HAFGRTR_EL2;
elsif PSTATE.EL == EL3 then
    X[t, 64] = HAFGRTR_EL2;
```

MSR HAFGRTR_EL2, <Xt>

op0	op1	CRn	CRm	op2
0b11	0b100	0b0011	0b0001	0b110

```
if PSTATE.EL == EL0 then
    UNDEFINED;
elsif PSTATE.EL == EL1 then
    if EL2Enabled() && HCR_EL2.<NV2,NV> == '11' then
        NVMem[0x1E8] = X[t, 64];
    elsif EL2Enabled() && HCR_EL2.NV == '1' then
        AArch64.SystemAccessTrap(EL2, 0x18);
    else
        UNDEFINED;
elsif PSTATE.EL == EL2 then
    if Halted() && HaveEL(EL3) && EDSCR.SDD == '1'
&& boolean IMPLEMENTATION_DEFINED "EL3 trap priority
when SDD == '1'" && SCR_EL3.FGTEn == '0' then
        UNDEFINED;
    elsif HaveEL(EL3) && SCR_EL3.FGTEn == '0' then
        if Halted() && EDSCR.SDD == '1' then
            UNDEFINED;
        else
            AArch64.SystemAccessTrap(EL3, 0x18);
    else
        HAFGRTR_EL2 = X[t, 64];
elsif PSTATE.EL == EL3 then
    HAFGRTR_EL2 = X[t, 64];
```

28/03/2023 16:01; 72747e43966d6b97dcbd230a1b3f0421d1ea3d94