HSTR_EL2, Hypervisor System Trap Register

The HSTR_EL2 characteristics are:

Purpose

Controls trapping to EL2 of EL1 or lower AArch32 accesses to the System register in the coproc == `0b1111` encoding space, by the CRn value used to access the register using MCR or MRC instruction. When the register is accessible using an MCRR or MRRC instruction, this is the CRm value used to access the register.

Configuration

AArch64 System register HSTR_EL2 bits [31:0] are architecturally mapped to AArch32 System register HSTR[31:0].

If EL2 is not implemented, this register is res0 from EL3.

This register has no effect if EL2 is not enabled in the current Security state.

Attributes

HSTR_EL2 is a 64-bit register.

Field descriptions

When AArch32 is supported:

63626160595857565554535251504948	47	46	45	44	43	42	41	40	39	38	37	36	35	34	33	32
RES0																
RES0	T15	RES0	T13	T12	T11	T10	T9	T8	T7	T6	T5	RES0	T3	T2	T1	T0
31302928272625242322212019181716	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0

Bits [63:16, 14, 4]

Reserved, res0.

T<n>, bit [n], for n = 15, 13 to 5, 3 to 0

The remaining fields control whether EL0 and EL1 accesses, using MCR, MRC, MCRR, and MRRC instructions, to the System registers in the coproc == `0b1111` encoding space, are trapped to EL2 as follows:

- MCR or MRC accesses to these registers that are trapped to EL2 are reported using EC syndrome value `0x03`, unless the access is undefined.
- MCRR or MRRC accesses to these registers that are trapped to EL2 are reported using EC syndrome value `0x04`, unless the access is undefined.

T<n>	**Meaning**
`0b0`	This control has no effect on EL0 or EL1 accesses to System registers.

`0b1`	System registers in the coproc == `0b1111` encoding space and CRn == <n> or CRm == <n> where T<n> is the name of this field, are trapped as follows: • An EL1 MCR or MRC access is trapped to EL2. • An EL0 MCR or MRC access is trapped to EL2, if the access is not undefined when the value of this field is 0. • An EL1 MCRR or MRRC access is trapped to EL2. • An EL0 MCRR or MRRC access is trapped to EL2, if the access is not undefined when the value of this field is 0. It is implementation defined whether an EL0 access using AArch32 is trapped to EL2, or is undefined. If the access is undefined, and generates an exception that is taken to EL1 or EL2 using AArch64, this is reported with EC syndrome value `0x00`. **Note** Arm expects that trapping to EL2 of EL0 accesses to these registers is unusual and used only when the hypervisor must virtualize EL0 operation. Arm recommends that, whenever possible, EL0 accesses to these registers behave as they would if the implementation did not include EL2. This means that, if the architecture does not support the EL0 access, then the register access instruction is

For example, when HSTR_EL2.T7 is 1, for instructions executed at EL1:

- An MCR or MRC instruction with coproc set to `0b1111` and <CRn> set to c7 is trapped to EL2.
- An MCRR or MRRC instruction with coproc set to `0b1111` and <CRm> set to c7 is trapped to EL2.

When FEAT_VHE is implemented, and the value of HCR_EL2.{E2H, TGE} is {1, 1}, this field behaves as 0 for all purposes other than a direct read of the value of this bit.

The reset behavior of this field is:

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

Otherwise:

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
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 [63:0]

Reserved, res0.

Accessing HSTR_EL2

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

MRS <Xt>, HSTR_EL2

op0	op1	CRn	CRm	op2
0b11	0b100	0b0001	0b0001	0b011

```
if PSTATE.EL == EL0 then
    UNDEFINED;
elsif PSTATE.EL == EL1 then
    if EL2Enabled() && HCR_EL2.<NV2,NV> == '11' then
        X[t, 64] = NVMem[0x080];
    elsif EL2Enabled() && HCR_EL2.NV == '1' then
        AArch64.SystemAccessTrap(EL2, 0x18);
    else
        UNDEFINED;
elsif PSTATE.EL == EL2 then
    X[t, 64] = HSTR_EL2;
elsif PSTATE.EL == EL3 then
```

```
    X[t, 64] = HSTR_EL2;
```

MSR HSTR_EL2, <Xt>

op0	op1	CRn	CRm	op2
0b11	0b100	0b0001	0b0001	0b011

```
if PSTATE.EL == EL0 then
    UNDEFINED;
elsif PSTATE.EL == EL1 then
    if EL2Enabled() && HCR_EL2.<NV2,NV> == '11' then
        NVMem[0x080] = X[t, 64];
    elsif EL2Enabled() && HCR_EL2.NV == '1' then
        AArch64.SystemAccessTrap(EL2, 0x18);
    else
        UNDEFINED;
elsif PSTATE.EL == EL2 then
    HSTR_EL2 = X[t, 64];
elsif PSTATE.EL == EL3 then
    HSTR_EL2 = X[t, 64];
```

28/03/2023 16:01; 72747e43966d6b97dcbd230a1b3f0421d1ea3d94