RMR_EL3, Reset Management Register (EL3)

The RMR_EL3 characteristics are:

Purpose

If EL3 is the implemented and this register is implemented:

- A write to the register at EL3 can request a Warm reset.
- If EL3 can use all Execution states, this register specifies the Execution state that the PE boots into on a Warm reset.

Configuration

AArch64 System register RMR_EL3 bits [31:0] are architecturally mapped to AArch32 System register RMR[31:0] when EL3 is implemented.

This register is present only when EL3 is implemented. Otherwise, direct accesses to RMR_EL3 are undefined.

When EL3 is implemented:

- If EL3 can use all Execution states then this register must be implemented.
- If EL3 cannot use AArch32, then it is implementation defined whether the register is implemented.

Otherwise, direct accesses to RMR_EL3 are undefined.

Attributes

RMR_EL3 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																															

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
RES0																														RR	AA64

Bits [63:2]

Reserved, res0.

RR, bit [1]

Reset Request. Setting this bit to 1 requests a Warm reset.

The reset behavior of this field is:

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

AA64, bit [0]
When EL3 is capable of using AArch32:

When EL3 can use AArch32, determines which Execution state the PE boots into after a Warm reset:

AA64	Meaning
`0b0`	AArch32.
`0b1`	AArch64.

On coming out of the Warm reset, execution starts at the implementation defined reset vector address of the specified Execution state.

If EL3 can only use AArch64 state, this bit is RAO/WI.

When implemented as a RW field, this field resets to 1 on a Cold reset.

Otherwise:

Reserved, RAO/WI.

Accessing RMR_EL3

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

MRS <Xt>, RMR_EL3

op0	op1	CRn	CRm	op2
0b11	0b110	0b1100	0b0000	0b010

```
if PSTATE.EL == EL3 && IsHighestEL(EL3) then
    X[t, 64] = RMR_EL3;
else
    UNDEFINED;
```

MSR RMR_EL3, <Xt>

op0	op1	CRn	CRm	op2
0b11	0b110	0b1100	0b0000	0b010

```
if PSTATE.EL == EL3 && IsHighestEL(EL3) then
    RMR_EL3 = X[t, 64];
else
    UNDEFINED;
```

28/03/2023 16:01; 72747e43966d6b97dcbd230a1b3f0421d1ea3d94