BRBINF<n>_EL1, Branch Record Buffer Information Register <n>, n = 0 - 31

The BRBINF<n>_EL1 characteristics are:

Purpose

The information for Branch record n + (BRBFCR_EL1.BANK Ã— 32).

Configuration

This register is present only when FEAT_BRBE is implemented. Otherwise, direct accesses to BRBINF<n>_EL1 are undefined.

Attributes

BRBINF<n>_EL1 is a 64-bit register.

Field descriptions

<table>
<tr><td>6362616059585756555453525150</td><td>49</td><td>48</td><td>47</td><td>46</td><td>454443424140</td><td>3938</td><td>37</td><td>363534</td><td>33 32</td></tr>
<tr><td colspan="4">RES0</td><td>CCU</td><td colspan="5">CC</td></tr>
<tr><td>RES0</td><td>LASTFAILED</td><td>T</td><td colspan="2">RES0</td><td>TYPE</td><td>EL</td><td>MPRED</td><td>RES0</td><td>VALID</td></tr>
<tr><td>31302928272625242322212019 18</td><td>17</td><td>16</td><td>15</td><td>14</td><td>13121110 9 8</td><td>7 6</td><td>5</td><td>4 3 2</td><td>1 0</td></tr>
</table>

Bits [63:47]

Reserved, res0.

CCU, bit [46]

The number of PE clock cycles since the last Branch record entry is unknown.

CCU	Meaning
`0b0`	Indicates that the number of PE clock cycles since the last Branch record is indicated by BRBINF<n>_EL1.CC.
`0b1`	Indicates that the number of PE clock cycles since the last Branch record is unknown.

The value in this field is only valid when BRBINF<n>_EL1.VALID != `0b00`.

The reset behavior of this field is:

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

Accessing this field has the following behavior:

- When BRBINF<n>_EL1.VALID == 0b00, access to this field is **RES0**.
- Otherwise, access to this field is **RO**.

CC, bits [45:32]

The number of PE clock cycles since the last Branch record entry.

The format of this field uses a mantissa and exponent to express the cycle count value, as follows:

- CC bits[7:0] indicate the mantissa M.
- CC bits[13:8] indicate the exponent E.

The cycle count is expressed using the following function:

if IsZero(E) then UInt(M) else UInt('1':M:Zeros(UInt(E)-1))

If required, the cycle count is rounded to a multiple of $2^{(E-1)}$ towards zero before being encoded.

A value of all ones in both the mantissa and exponent indicates the cycle count value exceeded the size of the cycle counter.

The value in this field is only valid when BRBINF<n>_EL1.VALID != `0b00`.

The reset behavior of this field is:

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

Accessing this field has the following behavior:

- Access is **RES0** if any of the following are true:
    - BRBINF<n>_EL1.CCU == 1
    - BRBINF<n>_EL1.VALID == 0b00
- Otherwise, access to this field is **RO**.

Bits [31:18]

Reserved, res0.

LASTFAILED, bit [17]

When FEAT_TME is implemented:

Indicates transaction failure or cancellation.

LASTFAILED	Meaning
0b0	Indicates that no transactions in a non-prohibited region have failed or been canceled between the previous Branch record and this Branch record.
0b1	Indicates that at least one transaction in a non-prohibited region has failed or been canceled between the previous Branch record and this Branch record.

The value in this field is only valid when BRBINF<n>_EL1.VALID != 0b00.

The reset behavior of this field is:

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

Accessing this field has the following behavior:

- When BRBINF<n>_EL1.VALID == 0b00, access to this field is **RES0**.
- Otherwise, access to this field is **RO**.

Otherwise:

Reserved, res0.

T, bit [16]

When FEAT_TME is implemented:

Transactional state.

T	Meaning
0b0	The branch or exception was not executed in Transactional state.
0b1	The branch or exception was executed in Transactional state.

The value in this field is only valid when BRBINF<n>_EL1.VALID == 0b10 or BRBINF<n>_EL1.VALID == 0b11.

The reset behavior of this field is:

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

Accessing this field has the following behavior:

- Access is **RES0** if any of the following are true:
  - BRBINF<n>_EL1.VALID == 0b00
  - BRBINF<n>_EL1.VALID == 0b01
- Otherwise, access to this field is **RO**.

Otherwise:

Reserved, res0.

Bits [15:14]

Reserved, res0.

TYPE, bits [13:8]

Branch type.

TYPE	Meaning
0b000000	Unconditional direct branch, excluding Branch with link.
0b000001	Indirect branch, excluding Branch with link, Return from subroutine, and Exception return.
0b000010	Direct Branch with link.
0b000011	Indirect Branch with link.
0b000101	Return from subroutine.
0b000111	Exception return.
0b001000	Conditional direct branch.
0b100001	Debug halt.
0b100010	Call.
0b100011	Trap.
0b100100	SError.
0b100110	Instruction debug.
0b100111	Data debug.
0b101010	Alignment.
0b101011	Inst Fault.

0b101100	Data Fault.
0b101110	IRQ.
0b101111	FIQ.
0b111001	Debug State Exit.

All other values are reserved.

The value in this field is only valid when BRBINF<n>_EL1.VALID != 0b00.

The reset behavior of this field is:

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

Accessing this field has the following behavior:

- When BRBINF<n>_EL1.VALID == 0b00, access to this field is **RES0**.
- Otherwise, access to this field is **RO**.

EL, bits [7:6]

The Exception Level at the target address.

EL	Meaning	Applies when
0b00	EL0.	
0b01	EL1.	
0b10	EL2.	
0b11	EL3.	When FEAT_BRBEv1p1 is implemented

The value in this field is only valid when BRBINF<n>_EL1.VALID == 0b11 or BRBINF<n>_EL1.VALID == 0b01.

The reset behavior of this field is:

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

Accessing this field has the following behavior:

- Access is **RES0** if any of the following are true:
    - BRBINF<n>_EL1.VALID == 0b00
    - BRBINF<n>_EL1.VALID == 0b10
- Otherwise, access to this field is **RO**.

MPRED, bit [5]

Branch mispredict.

MPRED	Meaning
0b0	Branch was correctly predicted or the result of the prediction was not captured.
0b1	Branch was incorrectly predicted.

The value in this field is only valid when BRBINF<n>_EL1.VALID == 0b11 or BRBINF<n>_EL1.VALID == 0b10.

The reset behavior of this field is:

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

Accessing this field has the following behavior:

- Access is **RES0** if any of the following are true:
    - BRBINF<n>_EL1.VALID == 0b00
    - BRBINF<n>_EL1.VALID == 0b01
    - BRBINF<n>_EL1.TYPE[5] == 1
- Otherwise, access to this field is **RO**.

Bits [4:2]

Reserved, res0.

VALID, bits [1:0]

The Branch record is valid.

VALID	Meaning
0b00	This Branch record is not valid. The values of following fields are not valid: • BRBTGT<n>_EL1.ADDRESS. • BRBSRC<n>_EL1.ADDRESS. • BRBINF<n>_EL1.MPRED. • BRBINF<n>_EL1.LASTFAILED. • BRBINF<n>_EL1.T. • BRBINF<n>_EL1.EL. • BRBINF<n>_EL1.TYPE. • BRBINF<n>_EL1.CC. • BRBINF<n>_EL1.CCU.

`0b01`	This Branch record is valid. The values of following fields are not valid: • BRBSRC<n>_EL1.ADDRESS. • BRBINF<n>_EL1.T. • BRBINF<n>_EL1.MPRED.
`0b10`	This Branch record is valid. The values of following fields are not valid: • BRBTGT<n>_EL1.ADDRESS. • BRBINF<n>_EL1.EL.
`0b11`	This Branch record is valid.

The reset behavior of this field is:

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

Accessing BRBINF<n>_EL1

BRBINF<n>_EL1 reads-as-zero if n + (BRBFCR_EL1.BANK Ã— 32) >= BRBIDR0_EL1.NUMREC.

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

```
MRS <Xt>, BRBINF<m>_EL1 ; Where m = 0-31
```

op0	op1	CRn	CRm	op2
0b10	0b001	0b1000	m[3:0]	m[4]:0b00

```
integer m = UInt(op2<2>:CRm<3:0>);

if PSTATE.EL == EL0 then
    UNDEFINED;
elsif PSTATE.EL == EL1 then
    if Halted() && HaveEL(EL3) && EDSCR.SDD == '1'
&& boolean IMPLEMENTATION_DEFINED "EL3 trap priority
when SDD == '1'" && MDCR_EL3.SBRBE != '11' &&
SCR_EL3.NS == '0' then
        UNDEFINED;
    elsif Halted() && HaveEL(EL3) && EDSCR.SDD ==
'1' && boolean IMPLEMENTATION_DEFINED "EL3 trap
priority when SDD == '1'" && MDCR_EL3.SBRBE == 'x0'
&& SCR_EL3.NS == '1' then
        UNDEFINED;
    elsif EL2Enabled() &&
IsFeatureImplemented(FEAT_FGT) && (!HaveEL(EL3) ||
SCR_EL3.FGTEn == '1') && HDFGRTR_EL2.nBRBDATA == '0'
then
        AArch64.SystemAccessTrap(EL2, 0x18);
    elsif HaveEL(EL3) && MDCR_EL3.SBRBE != '11' &&
SCR_EL3.NS == '0' then
        if Halted() && EDSCR.SDD == '1' then
            UNDEFINED;
        else
            AArch64.SystemAccessTrap(EL3, 0x18);
    elsif HaveEL(EL3) && MDCR_EL3.SBRBE == 'x0' &&
SCR_EL3.NS == '1' then
        if Halted() && EDSCR.SDD == '1' then
            UNDEFINED;
        else
            AArch64.SystemAccessTrap(EL3, 0x18);
    elsif m + (UInt(BRBFCR_EL1.BANK) * 32) >=
NUM_BRBE_RECORDS then
        X[t, 64] = Zeros(64);
    else
        X[t, 64] = BRBINF_EL1[m];
elsif PSTATE.EL == EL2 then
    if Halted() && HaveEL(EL3) && EDSCR.SDD == '1'
&& boolean IMPLEMENTATION_DEFINED "EL3 trap priority
when SDD == '1'" && MDCR_EL3.SBRBE != '11' &&
SCR_EL3.NS == '0' then
        UNDEFINED;
    elsif Halted() && HaveEL(EL3) && EDSCR.SDD ==
'1' && boolean IMPLEMENTATION_DEFINED "EL3 trap
priority when SDD == '1'" && MDCR_EL3.SBRBE == 'x0'
&& SCR_EL3.NS == '1' then
        UNDEFINED;
    elsif HaveEL(EL3) && MDCR_EL3.SBRBE != '11' &&
SCR_EL3.NS == '0' then
        if Halted() && EDSCR.SDD == '1' then
            UNDEFINED;
        else
            AArch64.SystemAccessTrap(EL3, 0x18);
    elsif HaveEL(EL3) && MDCR_EL3.SBRBE == 'x0' &&
SCR_EL3.NS == '1' then
        if Halted() && EDSCR.SDD == '1' then
            UNDEFINED;
        else
            AArch64.SystemAccessTrap(EL3, 0x18);
    elsif m + (UInt(BRBFCR_EL1.BANK) * 32) >=
NUM_BRBE_RECORDS then
        X[t, 64] = Zeros(64);
    else
        X[t, 64] = BRBINF_EL1[m];
elsif PSTATE.EL == EL3 then
    if m + (UInt(BRBFCR_EL1.BANK) * 32) >=
NUM_BRBE_RECORDS then
        X[t, 64] = Zeros(64);
    else
        X[t, 64] = BRBINF_EL1[m];
```

28/03/2023 16:02; 72747e43966d6b97dcbd230a1b3f0421d1ea3d94