

EORQV

Bitwise exclusive OR reduction of quadword vector segments

Bitwise exclusive OR of the same element numbers from each 128-bit source vector segment, placing each result into the corresponding element number of the 128-bit SIMD&FP destination register. Inactive elements in the source vector are treated as all zeros.

SVE2

(FEAT_SVE2p1)

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
0	0	0	0	0	1	0	0	size	0	1	1	1	0	1	0	0	1	Pg							Zn					Vd	

EORQV <Vd>.<T>, <Pg>, <Zn>.<Tb>

```
if !HaveSVE2p1() && !HaveSME2p1() then UNDEFINED;
constant integer esize = 8 << UInt(size);
integer g = UInt(Pg);
integer n = UInt(Zn);
integer d = UInt(Vd);
```

Assembler Symbols

<Vd> Is the name of the destination SIMD&FP register, encoded in the "Vd" field.

<T> Is an arrangement specifier, encoded in "size":

size	<T>
00	16B
01	8H
10	4S
11	2D

<Pg> Is the name of the governing scalable predicate register P0-P7, encoded in the "Pg" field.

<Zn> Is the name of the source scalable vector register, encoded in the "Zn" field.

<Tb>

Is the size specifier, encoded in “size”:

size	<Tb>
00	B
01	H
10	S
11	D

Operation

```
ChecksVEEnabled();
constant integer VL = CurrentVL;
constant integer PL = VL DIV 8;
constant integer segments = VL DIV 128;
constant integer elemperssegment = 128 DIV esize;
bits(PL) mask = P[g, PL];
bits(VL) operand = if AnyActiveElement(mask, esize) then Z[n, VL] else
bits(128) result = Zeros(128);
bits(128) stmp = Zeros(128);

bits(esize) dtmp;

for e = 0 to elemperssegment-1
    dtmp = Zeros(esize);
    for s = 0 to segments-1
        if ActivePredicateElement(mask, s * elemperssegment + e, esize)
            stmp = Elem[operand, s, 128];
            dtmp = dtmp EOR Elem[stmp, e, esize];
            Elem[result, e, esize] = dtmp<esize-1:0>;

V[d, 128] = result;
```

Operational information

If PSTATE.DIT is 1:

- The execution time of this instruction is independent of:
 - The values of the data supplied in any of its operand registers when its governing predicate register contains the same value for each execution.
 - The values of the NZCV flags.
- The response of this instruction to asynchronous exceptions does not vary based on:
 - The values of the data supplied in any of its operand registers when its governing predicate register contains the same value for each execution.
 - The values of the NZCV flags.

[Base
Instructions](#)

[SIMD&FP
Instructions](#)

[SVE
Instructions](#)

[SME
Instructions](#)

[Index by
Encoding](#)

[Sh
Pseud](#)

