LD1Q

Gather load quadwords

Gather load of quadwords to active elements of a vector register from memory addresses generated by a vector base plus a 64-bit unscaled scalar register offset. Inactive elements will not cause a read from Device memory or signal faults, and are set to zero in the destination vector.

This instruction is illegal when executed in Streaming SVE mode, unless FEAT_SME_FA64 is implemented and enabled.

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
1	1	0	0	0	1	0	0	0	0	0	Rm					1	0	1	Pg			Zn					Zt				

LD1Q { <Zt>.Q }, <Pg>/Z, [<Zn>.D{, <Xm>}]

```
if !HaveSVE2p1() then UNDEFINED;
integer t = UInt(Zt);
integer n = UInt(Zn);
integer m = UInt(Rm);
integer g = UInt(Pg);
```

Assembler Symbols

<Zt>	Is the name of the scalable vector register to be transferred, encoded in the "Zt" field.
<Pg>	Is the name of the governing scalable predicate register P0-P7, encoded in the "Pg" field.
<Zn>	Is the name of the base scalable vector register, encoded in the "Zn" field.
<Xm>	Is the optional 64-bit name of the general-purpose offset register, defaulting to XZR, encoded in the "Rm" field.

Operation

```
constant integer VL = CurrentVL;
constant integer PL = VL DIV 8;
CheckNonStreamingSVEEnabled();
constant integer elements = VL DIV 128;
bits(PL) mask = P[g, PL];
bits(VL) base;
bits(64) offset;
bits(VL) result;
boolean contiguous = FALSE;
boolean nontemporal = FALSE;
```

```
boolean tagchecked = TRUE;
AccessDescriptor accdesc = CreateAccDescSVE(MemOp_LOAD, nontemporal, co

if AnyActiveElement(mask, 128) then
    base = Z[n, VL];
    offset = X[m, 64];

for e = 0 to elements-1
    if ActivePredicateElement(mask, e, 128) then
        bits(64) addr = Elem[base, 2*e, 64] + offset;
        Elem[result, e, 128] = Mem[addr, 16, accdesc];
    else
        Elem[result, e, 128] = Zeros(128);

Z[t, VL] = result;
```

Operational information

If PSTATE.DIT is 1, the timing of this instruction is insensitive to the value of the data being loaded or stored when its governing predicate register contains the same value for each execution.

Internal version only: isa v33.64, AdvSIMD v29.12, pseudocode no_diffs_2023_09_RC2, sve v2023-06_rel ; Build timestamp: 2023-09-18T17:56