ZERO (double-vector)

Base

Instructions

Zero ZA double-vector groups

The instruction zeroes one, two, or four ZA double-vector groups. The lowest of the two consecutive vector numbers forming the double-vector group within all of, each half of, or each quarter of the ZA array are selected by the sum of the vector select register and immediate offset, modulo all, half, or quarter the number of ZA array vectors.

The vector group symbol, VGx2 or VGx4, indicates that the ZA operand consists of two or four ZA double-vector groups respectively.

It has encodings from 3 classes: $\underline{\text{One ZA double-vector}}$, $\underline{\text{Two ZA double-vectors}}$ and $\underline{\text{Four ZA double-vectors}}$

One ZA double-vector (FEAT SME2p1)

ZERO ZA.D[<Wv>, <offs1>:<offs2>]

```
if !HaveSME2p1() then UNDEFINED;
integer v = UInt('010':Rv);
integer offset = UInt(off3:'0');
constant integer ngrp = 1;
constant integer nvec = 2;
```

Two ZA double-vectors (FEAT SME2p1)

```
ZERO ZA.D[<Wv>, <offs1>:<offs2>, VGx2]
```

```
if !HaveSME2p1() then UNDEFINED;
integer v = UInt('010':Rv);
integer offset = UInt(off2:'0');
constant integer ngrp = 2;
constant integer nvec = 2;
```

Four ZA double-vectors (FEAT SME2p1)

zero za.D[<wv>, <offs1>:<offs2>, VGx4] if !HaveSME2p1() then UNDEFINED; integer v = UInt('010':Rv); integer offset = UInt(off2:'0'); constant integer ngrp = 4; constant integer nvec = 2;

Assembler Symbols

<Wv> Is the 32-bit name of the vector select register W8-W11, encoded in the "Ry" field.

<offs1> For the one ZA double-vector variant: is the vector select offset, pointing to first of two consecutive vectors, encoded as "off3" field times 2.

For the four ZA double-vectors and two ZA double-vectors variant: is the vector select offset, pointing to first of two consecutive vectors, encoded as "off2" field times 2.

<offs2> For the one ZA double-vector variant: is the vector select
 offset, pointing to last of two consecutive vectors, encoded
 as "off3" field times 2 plus 1.

For the four ZA double-vectors and two ZA double-vectors variant: is the vector select offset, pointing to last of two consecutive vectors, encoded as "off2" field times 2 plus 1.

Operation

```
CheckStreamingSVEAndZAEnabled();
constant integer VL = CurrentVL;
integer vectors = VL DIV 8;
integer vstride = vectors DIV ngrp;
bits(32) vbase = X[v, 32];
integer vec = (UInt(vbase) + offset) MOD vstride;
vec = vec - (vec MOD nvec);

for r = 0 to ngrp-1
    for i = 0 to nvec-1
        ZAvector[vec + i, VL] = Zeros(VL);
    vec = vec + vstride;
```

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 registers.
 - 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 registers.
 - The values of the NZCV flags.

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

Sh

Pseu

Copyright \hat{A} © 2010-2023 Arm Limited or its affiliates. All rights reserved. This document is Non-Confidential.