MOV (vector to tile, single)

Move vector register to ZA tile slice

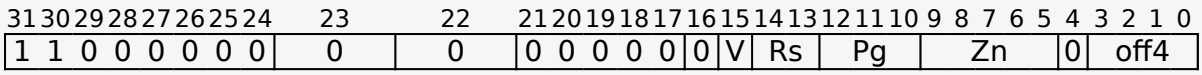
The instruction operates on individual horizontal or vertical slices within a named ZA tile of the specified element size. The slice number within the tile is selected by the sum of the slice index register and immediate offset, modulo the number of such elements in a vector. The immediate offset is in the range 0 to the number of elements in a 128-bit vector segment minus 1. Inactive elements in the destination slice remain unmodified.

This is an alias of MOVA (vector to tile, single). This means:

- The encodings in this description are named to match the encodings of MOVA (vector to tile, single).
- The description of MOVA (vector to tile, single) gives the operational pseudocode, any constrained unpredictable behavior, and any operational information for this instruction.

It has encodings from 5 classes: 8-bit , 16-bit , 32-bit , 64-bit and 128-bit

8-bit



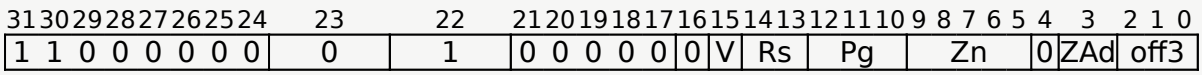

```
MOV ZA0<HV>.B[<Ws>, <offs>], <Pg>/M, <Zn>.B
```

is equivalent to

```
MOVA ZA0<HV>.B[<Ws>, <offs>], <Pg>/M, <Zn>.B
```

and is always the preferred disassembly.

16-bit



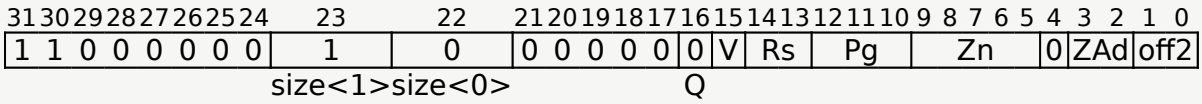
```
MOV <ZAd><HV>.H[<Ws>, <offs>], <Pg>/M, <Zn>.H
```

is equivalent to

```
MOVA <ZAd><HV>.H[<Ws>, <offs>], <Pg>/M, <Zn>.H
```

and is always the preferred disassembly.

32-bit



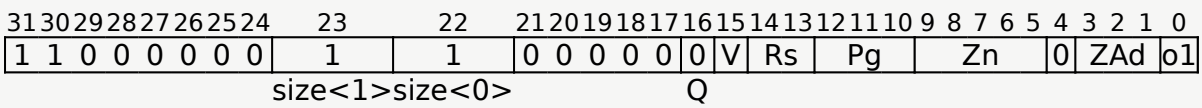

```
MOV <ZAd><HV>.S[<Ws>, <offs>], <Pg>/M, <Zn>.S
```

is equivalent to

```
MOVA <ZAd><HV>.S[<Ws>, <offs>], <Pg>/M, <Zn>.S
```

and is always the preferred disassembly.

64-bit



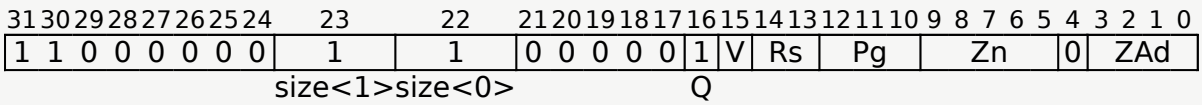

```
MOV <ZAd><HV>.D[<Ws>, <offs>], <Pg>/M, <Zn>.D
```

is equivalent to

```
MOVA <ZAd><HV>.D[<Ws>, <offs>], <Pg>/M, <Zn>.D
```

and is always the preferred disassembly.

128-bit




```
MOV <ZAd><HV>.Q[<Ws>, <offs>], <Pg>/M, <Zn>.Q
```

is equivalent to

```
MOVA <ZAd><HV>.Q[<Ws>, <offs>], <Pg>/M, <Zn>.Q
```

and is always the preferred disassembly.

Assembler Symbols

<ZAd>	For the 16-bit variant: is the name of the ZA tile ZA0-ZA1 to be accessed, encoded in the "ZAd" field.

For the 32-bit variant: is the name of the ZA tile ZA0-ZA3 to be accessed, encoded in the "ZAd" field.

For the 64-bit variant: is the name of the ZA tile ZA0-ZA7 to be accessed, encoded in the "ZAd" field.

For the 128-bit variant: is the name of the ZA tile ZA0-ZA15 to be accessed, encoded in the "ZAd" field.

<HV>

Is the horizontal or vertical slice indicator, encoded in “V”:

V	<HV>
0	H
1	V

<Ws>

Is the 32-bit name of the slice index register W12-W15, encoded in the "Rs" field.

<offs>

For the 8-bit variant: is the slice index offset, in the range 0 to 15, encoded in the "off4" field.

For the 16-bit variant: is the slice index offset, in the range 0 to 7, encoded in the "off3" field.

For the 32-bit variant: is the slice index offset, in the range 0 to 3, encoded in the "off2" field.

For the 64-bit variant: is the slice index offset, in the range 0 to 1, encoded in the "o1" field.

For the 128-bit variant: is the slice index offset 0.

<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.

Operation

The description of MOVA (vector to tile, single) gives the operational pseudocode for this instruction.

Operational information

If FEAT_SVE2 is implemented or FEAT_SME is implemented, then 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.

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