

BFCVT

Multi-vector floating-point convert from single-precision to packed BFloat16 format

Convert to BFloat16 from single-precision, each element of the two source vectors, and place the results in the half-width destination elements.

This instruction follows SME2 floating-point numerical behaviors corresponding to instructions that place their results in one or more SVE Z vectors.

This instruction is unpredicated.

SME2

(FEAT_SME2)

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	0	0	1	0	1	1	0	0	0	0	0	1	1	1	0	0	0	Zn			0	Zd					
																								N							

BFCVT <Zd>.H, { <Zn1>.S--<Zn2>.S }

```
if !HaveSME2() then UNDEFINED;
integer n = UInt(Zn:'0');
integer d = UInt(Zd);
```

Assembler Symbols

- <Zd> Is the name of the destination scalable vector register, encoded in the "Zd" field.
- <Zn1> Is the name of the first scalable vector register of a multi-vector sequence, encoded as "Zn" times 2.
- <Zn2> Is the name of the second scalable vector register of a multi-vector sequence, encoded as "Zn" times 2 plus 1.

Operation

```
CheckStreamingSVEEnabled();
constant integer VL = CurrentVL;
constant integer elements = VL DIV 32;
bits(VL) result;

bits(VL) operand1 = Z[n+0, VL];
bits(VL) operand2 = Z[n+1, VL];
for e = 0 to elements-1
    bits(32) element1 = Elem[operand1, e, 32];
    bits(32) element2 = Elem[operand2, e, 32];
    bits(16) res1 = FPConvertBF(element1, FPCR[]);
    bits(16) res2 = FPConvertBF(element2, FPCR[]);
    Elem[result, e, 16] = res1;
```

```
Elem[result, elements+e, 16] = res2;  
Z[d, VL] = result;
```

[Base
Instructions](#)

[SIMD&FP
Instructions](#)

[SVE
Instructions](#)

[SME
Instructions](#)

[Index by
Encoding](#)

[Sh
Pseudocode](#)

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

Copyright Â© 2010-2023 Arm Limited or its affiliates. All rights reserved. This
document is Non-Confidential.