

FRINTZ (scalar)

Floating-point Round to Integral, toward Zero (scalar). This instruction rounds a floating-point value in the SIMD&FP source register to an integral floating-point value of the same size using the Round towards Zero rounding mode, and writes the result to the SIMD&FP destination register.

A zero input gives a zero result with the same sign, an infinite input gives an infinite result with the same sign, and a NaN is propagated as for normal arithmetic.

A floating-point exception can be generated by this instruction. Depending on the settings in [FPCR](#), the exception results in either a flag being set in [FPSR](#), or a synchronous exception being generated. For more information, see [Floating-point exception traps](#).

Depending on the settings in the [CPACR_EL1](#), [CPTR_EL2](#), and [CPTR_EL3](#) registers, and the current Security state and Exception level, an attempt to execute the instruction might be trapped.

| | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|-------|----|----|----|----|----|---|---|---|----|---|---|---|----|---|---|--|
| 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 | 1 | 1 | 1 | 1 | 0 | f | t | y | p | e | 1 | 0 | 0 | 1 | 0 | 1 | 1 | 1 | 0 | 0 | 0 | 0 | Rn | | | | Rd | | | |
| | | | | | | | | | | | | | | | | rmode | | | | | | | | | | | | | | | | |

Half-precision (ftype == 11) (FEAT_FP16)

```
FRINTZ <Hd>, <Hn>
```

Single-precision (ftype == 00)

```
FRINTZ <Sd>, <Sn>
```

Double-precision (ftype == 01)

```
FRINTZ <Dd>, <Dn>
```

```
if ftype == '10' || (ftype == '11' && !IsFeatureImplemented(FEAT_FP16))
integer d = UInt(Rd);
integer n = UInt(Rn);

constant integer esize = 8 << UInt(ftype EOR '10');
FPRounding rounding;
rounding = FPDecodeRounding('11');
```

Assembler Symbols

<Dd> Is the 64-bit name of the SIMD&FP destination register, encoded in the "Rd" field.

| | |
|------|--|
| <Dn> | Is the 64-bit name of the SIMD&FP source register, encoded in the "Rn" field. |
| <Hd> | Is the 16-bit name of the SIMD&FP destination register, encoded in the "Rd" field. |
| <Hn> | Is the 16-bit name of the SIMD&FP source register, encoded in the "Rn" field. |
| <Sd> | Is the 32-bit name of the SIMD&FP destination register, encoded in the "Rd" field. |
| <Sn> | Is the 32-bit name of the SIMD&FP source register, encoded in the "Rn" field. |

Operation

```

CheckFPEnabled64() ;

FPCRType fpcr = FPCR[];
boolean merge = IsMerging(fpcr);
bits(128) result = if merge then V[d, 128] else Zeros(128);
bits(esize) operand = V[n, esize];

Elem[result, 0, esize] = FPRoundInt(operand, fpcr, rounding, FALSE);
V[d, 128] = 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 © 2010-2023 Arm Limited or its affiliates. All rights reserved. This
document is Non-Confidential.