<u>SME</u>	Index by
Instructions	Encoding

Pseu

SIMD&FP Base **SVE** Instructions Instructions **Instructions**

UQRSHR (two registers)

Multi-vector unsigned saturating rounding shift right narrow by immediate

Shift right by an immediate value, the unsigned integer value in each element of the two source vectors and place the rounded results in the halfwidth destination elements. Each result element is saturated to the halfwidth N-bit element's unsigned integer range 0 to (2^{N}) -1. The immediate shift amount is an unsigned value in the range 1 to 16. 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 1 1 1 0 imm4 | 1 1 0 1 0 1
                                                       Zn
```

```
UQRSHR \langle Zd \rangle.H, { \langle Zn1 \rangle.S-\langle Zn2 \rangle.S }, #\langle const \rangle
```

```
if !HaveSME2() then UNDEFINED;
constant integer esize = 16;
integer n = UInt(Zn:'0');
integer d = UInt(Zd);
integer shift = esize - UInt(imm4);
```

Assembler Symbols

<za></za>	Is the name of the destination scalable vector register, encoded in the "Zd" field.
<zn1></zn1>	Is the name of the first scalable vector register of a multivector sequence, encoded as "Zn" times 2.
<zn2></zn2>	Is the name of the second scalable vector register of a multi-vector sequence, encoded as "Zn" times 2 plus 1.
<const></const>	Is the immediate shift amount, in the range 1 to 16, encoded in the "imm4" field.

Operation

```
CheckStreamingSVEEnabled();
constant integer VL = CurrentVL;
constant integer elements = VL DIV (2 * esize);
bits(VL) result;
for r = 0 to 1
    bits (VL) operand = \mathbb{Z}[n+r, VL];
```

```
for e = 0 to elements-1
   bits(2 * esize) element = Elem[operand, e, 2 * esize];
   integer res = (UInt(element) + (1 << (shift-1))) >> shift;
   Elem[result, r*elements + e, esize] = UnsignedSat(res, esize);
Z[d, VL] = result;
```

Sh

Pseu

BaseSIMD&FPSVESMEIndex byInstructionsInstructionsInstructionsInstructions

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.