## **SMULH**

Signed Multiply High multiplies two 64-bit register values, and writes bits[127:64] of the 128-bit result to the 64-bit destination register.

```
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 0 0 1 1 0 1 0 0 Rm 0 (1)(1)(1)(1) Rn Rd
```

```
SMULH <Xd>, <Xn>, <Xm>
integer d = UInt(Rd);
integer n = UInt(Rn);
integer m = UInt(Rm);
```

## **Assembler Symbols**

```
<Xd> Is the 64-bit name of the general-purpose destination register, encoded in the "Rd" field.
<Xn> Is the 64-bit name of the first general-purpose source register holding the multiplicand, encoded in the "Rn" field.
<Xm> Is the 64-bit name of the second general-purpose source register holding the multiplier, encoded in the "Rm" field.
```

## **Operation**

```
bits(64) operand1 = \underline{X}[n, 64];
bits(64) operand2 = \underline{X}[m, 64];
integer result;
result = \underline{Int}(operand1, FALSE) * \underline{Int}(operand2, FALSE);
\underline{X}[d, 64] = result<127:64>;
```

## **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.

<u>Base</u>	SIMD&FP	<u>SVE</u>	<u>SME</u>	Index by
<u>Instructions</u>	<u>Instructions</u>	<u>Instructions</u>	<u>Instructions</u>	<b>Encoding</b>

Sh

Pseu

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