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

Sh

Pseu

Base Instructions

SIMD&FP **Instructions**

SVE Instructions

EOR (immediate)

Bitwise Exclusive-OR (immediate) performs a bitwise exclusive-OR of a register value and an immediate value, and writes the result to the 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
sf 1 0 1 0 0 1 0 0 N
                             immr
                                            imms
                                                            Rn
                                                                        Rd
   opc
```

```
32-bit (sf == 0 \&\& N == 0)
```

```
EOR <Wd WSP>, <Wn>, #<imm>
```

64-bit (sf == 1)

```
EOR \langle Xd | SP \rangle, \langle Xn \rangle, \#\langle imm \rangle
integer d = UInt(Rd);
integer n = UInt(Rn);
constant integer datasize = 32 << UInt(sf);</pre>
bits(datasize) imm;
if sf == '0' && N != '0' then UNDEFINED;
(imm, −) = DecodeBitMasks(N, imms, immr, TRUE, datasize);
```

Assembler Symbols

<Wd|WSP> Is the 32-bit name of the destination general-purpose

register or stack pointer, encoded in the "Rd" field.

<Wn> Is the 32-bit name of the general-purpose source register,

encoded in the "Rn" field.

<Xd|SP> Is the 64-bit name of the destination general-purpose

register or stack pointer, encoded in the "Rd" field.

<Xn> Is the 64-bit name of the general-purpose source register,

encoded in the "Rn" field.

<imm> For the 32-bit variant: is the bitmask immediate, encoded in

"imms:immr".

For the 64-bit variant: is the bitmask immediate, encoded in

"N:imms:immr".

Operation

```
bits(datasize) result;
bits(datasize) operand1 = X[n, datasize];
result = operand1 EOR imm;
```

```
if d == 31 then
    SP[] = ZeroExtend(result, 64);
else
    X[d, datasize] = result;
```

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.

BaseSIMD&FPSVESMEIndex byInstructionsInstructionsInstructionsInstructionsEncoding

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.

Sh Pseu