UDIV

Unsigned Divide divides an unsigned integer register value by another unsigned integer register value, and writes the result to the destination register. The condition flags are not affected.

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	0	0	1	1	0	1	0	1	1	0	Rm	0	0	0	0	1	0	Rn	Rd

o1

32-bit (sf == 0)

        UDIV <Wd>, <Wn>, <Wm>

64-bit (sf == 1)

        UDIV <Xd>, <Xn>, <Xm>

```
integer d = UInt(Rd);
integer n = UInt(Rn);
integer m = UInt(Rm);
constant integer datasize = 32 << UInt(sf);
```

Assembler Symbols

<Wd>   Is the 32-bit name of the general-purpose destination register, encoded in the "Rd" field.

<Wn>   Is the 32-bit name of the first general-purpose source register, encoded in the "Rn" field.

<Wm>   Is the 32-bit name of the second general-purpose source register, encoded in the "Rm" field.

<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, encoded in the "Rn" field.

<Xm>   Is the 64-bit name of the second general-purpose source register, encoded in the "Rm" field.

Operation

```
bits(datasize) operand1 = X[n, datasize];
bits(datasize) operand2 = X[m, datasize];
integer result;

if IsZero(operand2) then
    result = 0;
```

```
else
    result = RoundTowardsZero(Real(Int(operand1, TRUE)) / Real(Int(oper

X[d, datasize] = result<datasize-1:0>;
```

Base
Instructions

SIMD&FP
Instructions

SVE
Instructions

SME
Instructions

Index by
Encoding

Sh
Pseu