FMOV (vector, immediate)

Floating-point move immediate (vector). This instruction copies an immediate floating-point constant into every element of the SIMD&FP destination register.

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.

It has encodings from 2 classes: Half-precision and Single-precision and double-precision

Half-precision
(FEAT_FP16)

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	Q	0	0	1	1	1	1	0	0	0	0	0	a	b	c	1	1	1	1	1	1	d	e	f	g	h			Rd		

        FMOV `<Vd>.<T>`, #`<imm>`

```
if !IsFeatureImplemented(FEAT_FP16) then UNDEFINED;

integer rd = UInt(Rd);

constant integer datasize = 64 << UInt(Q);
bits(datasize) imm;

bits(8) imm8 = a:b:c:d:e:f:g:h;
bits(16) imm16 = imm8<7>:NOT(imm8<6>):Replicate(imm8<6>, 2):imm8<5:0>:Z

imm = Replicate(imm16, datasize DIV 16);
```

Single-precision and double-precision

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	Q	op	0	1	1	1	1	0	0	0	0	0	a	b	c	1	1	1	1	0	1	d	e	f	g	h			Rd		

cmode

Single-precision (op == 0)

        FMOV `<Vd>.<T>`, #`<imm>`

Double-precision (Q == 1 && op == 1)

        FMOV `<Vd>.2D`, #`<imm>`

```
integer rd = UInt(Rd);

constant integer datasize = 64 << UInt(Q);
bits(datasize) imm;
bits(64) imm64;
```

```
if cmode:op == '11111' then
    // FMOV Dn,#imm is in main FP instruction set
    if Q == '0' then UNDEFINED;

imm64 = AdvSIMDExpandImm(op, cmode, a:b:c:d:e:f:g:h);
imm = Replicate(imm64, datasize DIV 64);
```

Assembler Symbols

<Vd>　　　　　Is the name of the SIMD&FP destination register, encoded in the "Rd" field.

<T>

　　　　　For the half-precision variant: is an arrangement specifier, encoded in "Q":

Q	<T>
0	4H
1	8H

　　　　　For the single-precision variant: is an arrangement specifier, encoded in "Q":

Q	<T>
0	2S
1	4S

<imm>　　　　Is a signed floating-point constant with 3-bit exponent and normalized 4 bits of precision, encoded in "a:b:c:d:e:f:g:h". For details of the range of constants available and the encoding of <imm>, see *Modified immediate constants in A64 floating-point instructions*.

Operation

```
CheckFPAdvSIMDEnabled64();

V[rd, datasize] = imm;
```