

MSR (register)

Move general-purpose register to System Register allows the PE to write an AArch64 System register from a general-purpose 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 1 0 1 0 1 0 1 0 0 | | | | | | | | | | | 0 | 1 | 0 | op1 | | CRn | | | CRm | | | op2 | | | Rt | | | | | | |
| | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |

MSR (<systemreg> | s<op0>_<op1>_<Cn>_<Cm>_<op2>), <Xt>

```
AArch64.CheckSystemAccess('1':o0, op1, CRn, CRm, op2, Rt, L);
```

```
integer t = UInt(Rt);
```

```
integer sys_op0 = 2 + UInt(o0);
```

```
integer sys_op1 = UInt(op1);
```

```
integer sys_op2 = UInt(op2);
```

```
integer sys_crn = UInt(CRn);
```

```
integer sys_crm = UInt(CRm);
```

Assembler Symbols

<systemreg> Is a System register name, encoded in the "o0:op1:CRn:CRm:op2".

The System register names are defined in '[AArch64 System Registers in the System Register XML](#)'.

<op0>

Is an unsigned immediate, encoded in "o0":

| o0 | <op0> |
|----|-------|
| 0 | 2 |
| 1 | 3 |

<op1>

Is a 3-bit unsigned immediate, in the range 0 to 7, encoded in the "op1" field.

<Cn>

Is a name 'Cn', with 'n' in the range 0 to 15, encoded in the "CRn" field.

<Cm>

Is a name 'Cm', with 'm' in the range 0 to 15, encoded in the "CRm" field.

<op2>

Is a 3-bit unsigned immediate, in the range 0 to 7, encoded in the "op2" field.

<Xt>

Is the 64-bit name of the general-purpose source register, encoded in the "Rt" field.

Operation

[AArch64.SysRegWrite](#)(sys_op0, sys_op1, sys_crn, sys_crm, sys_op2, t);

| | | | | | |
|---------------------------------------|--|--------------------------------------|--------------------------------------|---------------------------------------|-----------------------------|
| Base Instructions | SIMD&FP Instructions | SVE Instructions | SME Instructions | Index by Encoding | 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 Â© 2010-2023 Arm Limited or its affiliates. All rights reserved. This
document is Non-Confidential.