RCWSCLR, RCWSCLRA, RCWSCLRL, RCWSCLRAL

Read Check Write Software atomic bit Clear on doubleword in memory atomically loads a 64-bit doubleword from memory, performs a bitwise AND with the complement of the value held in a register on it, and conditionally stores the result back to memory. Storing of the result back to memory is conditional on RCW Checks and RCWS Checks. The value initially loaded from memory is returned in the destination register. This instruction updates the condition flags based on the result of the update of memory.

- RCWSCLRA and RCWSCLRAL load from memory with acquire semantics.
- RCWSCLRL and RCWSCLRAL store to memory with release semantics.
- RCWSCLR has neither acquire nor release semantics.

Integer
(FEAT_THE)

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	1	1	1	1	0	0	0	A	R	1	Rs					1	0	0	1	0	0	Rn					Rt				
	S															o3	opc														

RCWSCLR (A == 0 && R == 0)

```
RCWSCLR <Xs>, <Xt>, [<Xn|SP>]
```

RCWSCLRA (A == 1 && R == 0)

```
RCWSCLRA <Xs>, <Xt>, [<Xn|SP>]
```

RCWSCLRAL (A == 1 && R == 1)

```
RCWSCLRAL <Xs>, <Xt>, [<Xn|SP>]
```

RCWSCLRL (A == 0 && R == 1)

```
RCWSCLRL <Xs>, <Xt>, [<Xn|SP>]
```

```
if !IsFeatureImplemented(FEAT_THE) then UNDEFINED;
integer t = UInt(Rt);
integer n = UInt(Rn);
integer s = UInt(Rs);

boolean acquire = A == '1' && Rt != '11111';
boolean release = R == '1';
boolean tagchecked = n != 31;
```

Assembler Symbols

<Xs> Is the 64-bit name of the general-purpose register to be stored, encoded in the "Rs" field.

<Xt> Is the 64-bit name of the general-purpose register to be loaded, encoded in the "Rt" field.

<Xn|SP> Is the 64-bit name of the general-purpose base register or stack pointer, encoded in the "Rn" field.

Operation

```
if IsD128Enabled(PSTATE.EL) then UNDEFINED;
bits(64) address;
bits(64) newdata = X[s, 64];
bits(64) readdata;
bits(4) nzcv;

AccessDescriptor accdesc = CreateAccDescRCW(MemAtomicOp_BIC, TRUE, acqu

if n == 31 then
    CheckSPAlignment();
    address = SP[];
else
    address = X[n, 64];

bits(64) compdata = bits(64) UNKNOWN;     // Irrelevant when not executi
(nzcv, readdata) = MemAtomicRCW(address, compdata, newdata, accdesc);

PSTATE.<N,Z,C,V> = nzcv;
X[t, 64] = readdata;    // Return the old value when t!=31
```

Operational information

If PSTATE.DIT is 1, the timing of this instruction is insensitive to the value of the data being loaded or stored.

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