## STSETH, STSETLH

Atomic bit set on halfword in memory, without return, atomically loads a 16-bit halfword from memory, performs a bitwise OR with the value held in a register on it, and stores the result back to memory.

- `STSETH` does not have release semantics.
- `STSETLH` stores to memory with release semantics, as described in *Load-Acquire, Store-Release*.

For information about memory accesses, see *Load/Store addressing modes*.

This is an alias of LDSETH, LDSETAH, LDSETALH, LDSETLH. This means:

- The encodings in this description are named to match the encodings of LDSETH, LDSETAH, LDSETALH, LDSETLH.
- The description of LDSETH, LDSETAH, LDSETALH, LDSETLH gives the operational pseudocode, any constrained unpredictable behavior, and any operational information for this instruction.

**Integer**
**(FEAT_LSE)**

| 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 | 0 | R | 1 | Rs | 0 | 0 1 1 | 0 0 | Rn | 1 1 1 1 1 |
| size | | | | A | | | | opc | | | | Rt |

**No memory ordering (R == 0)**

```
STSETH <Ws>, [<Xn|SP>]
```

is equivalent to

```
LDSETH <Ws>, WZR, [<Xn|SP>]
```

and is always the preferred disassembly.

**Release (R == 1)**

```
STSETLH <Ws>, [<Xn|SP>]
```

is equivalent to

```
LDSETLH <Ws>, WZR, [<Xn|SP>]
```

and is always the preferred disassembly.

**Assembler Symbols**

| | |
|---|---|
| <Ws> | Is the 32-bit name of the general-purpose register holding the data value to be operated on with the contents of the memory location, encoded in the "Rs" field. |
| <Xn\|SP> | Is the 64-bit name of the general-purpose base register or stack pointer, encoded in the "Rn" field. |

**Operation**

The description of LDSETH, LDSETAH, LDSETALH, LDSETLH gives the operational pseudocode for this instruction.

**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