## <Load instruction>

LD.W

Load word

Load of word data

[Instruction format]

- (1) LD.W disp16 [reg1], reg2
- (2) LD.W disp23 [reg1], reg3

[Operation]

(1) adr ← GR [reg1] + sign-extend (disp16)

GR [reg2] ← Load-memory (adr, Word)

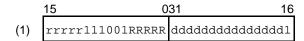
(2)  $adr \leftarrow GR [reg1] + sign-extend (disp23)$ 

GR [reg3] ← Load-memory (adr, Word)

[Format]

- (1) Format VII
- (2) Format XIV

### [Opcode]



Where dddddddddddddd is the higher 15 bits of disp16.

	15	131 16	47 32
(2)	00000111100RRRRR	wwwwwdddddd01001	DDDDDDDDDDDDDDD

Where RRRRR = reg1, wwwww = reg3.

dddddd is the lower side bits 6 to 1 of disp23.

DDDDDDDDDDDDDD is the higher 16 bits of disp23.

[Flags]

CY --

OV -

S

Z --

SAT --

#### [Description]

- (1) Adds the word data of general-purpose register reg1 to the 16-bit displacement data, sign-extended to word length, to generate a 32-bit address. Word data is read from this 32-bit address, and stored in general-purpose register reg2.
- (2) Adds the word data of general-purpose register reg1 to the 23-bit displacement data, sign-extended to word length, to generate a 32-bit address. Word data is read from this address, and stored in general-purpose register reg3.

#### <Store instruction>

ST.W

Store word

Storage of word data

[Instruction format]

- (1) ST.W reg2, disp16 [reg1]
- (2) ST.W reg3, disp23 [reg1]

[Operation]

(1) adr ← GR [reg1] + sign-extend (disp16)

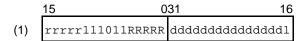
Store-memory (adr, GR [reg2], Word)

(2) adr ← GR [reg1] + sign-extend (disp23)
Store-memory (adr, GR [reg3], Word)

[Format]

- (1) Format VII
- (2) Format XIV

#### [Opcode]



Where dddddddddddddd is the higher 15 bits of disp16.

	15	31 16	47 32
(2)	00000111100RRRRR	wwwwwdddddd01111	DDDDDDDDDDDDDDD

Where RRRRR = reg1, wwwww = reg3.

dddddd is the lower side bits 6 to 1 of disp23.

DDDDDDDDDDDDDD is the higher 16 bits of disp23.

[Flags]

CY --

OV --

S

Z --

SAT --

# [Description]

- (1) Adds the data of general-purpose register reg1 to the 16-bit displacement data, sign-extended to word length, to generate a 32-bit address and stores the word data of general-purpose register reg2 to the generated 32-bit address.
- (2) Adds the data of general-purpose register reg1 to the 23-bit displacement data, sign-extended to word length, to generate a 32-bit address and stores the lowest word data of general-purpose register reg3 to the generated 32-bit address.