LD — *Load register* — **LD**

P851M
P852M
P856M
P857M

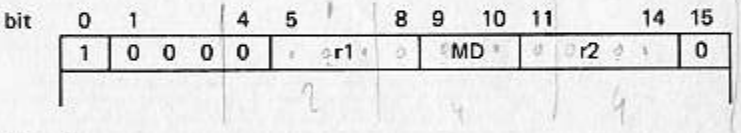
Syntax: [label]␣ LD [*]␣ r1, m [,r2]

The contents of the register specified by r1 are replaced by the contents of the effective memory address. This effective memory address can be found as follows:

Type	*Function*		*MD*	*Syntax*
T4	(m)	→ r1	10	LD r1, m
T5	(m + (r2))	→ r1	10	LD r1, m, r2
T6	((m))	→ r1	11	LD* r1, m
T7	((m + (r2)))	→ r1	11	LD* r1, m, r2

Condition register:

CR = 0 if (r1) = 0
1 if (r1) > 0
2 if (r1) < 0



Remark:
Restricted to system mode if r1 = A15.

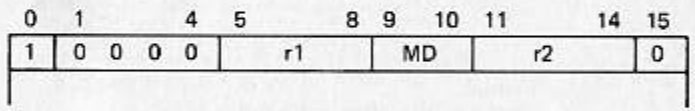
LDR — *Load register/register* — **LDR** — P851M P852M P856M P857M

Syntax: [label]␣ LDR [*]␣ r1, r2

The 16 bits of the register specified by r1 are replaced either by the contents of the register specified by r2 (direct addressing) or by the contents of the effective memory address which can be found in the register specified by r2 (indirect addressing). In the last addressing mode, if r2 specifies the A15 register, the latter is assumed to be the stack. In this case, the pointer is updated (i.e. incremented by one word to point to the latest entry) before the transfer of data occurs.

Type	*Function*		*MD*	*Syntax*
T1	(r2)	→ r1	00	LDR r1, r2
T3A	((r2))	→ r1	01	LDR* r1, r2
T3B	(A15) + 2 → A15, ((A15))	→ r1	01	LDR* r1, A15

Condition register:
CR = 0 if (r1) = 0
1 if (r1) > 0
2 if (r1) < 0



Remark:
Restricted to system mode if r1 = A15 or if type 3B.

LDK LDKL — *Load constant* — **LDK LDKL** — P851M P852M P856M P857M

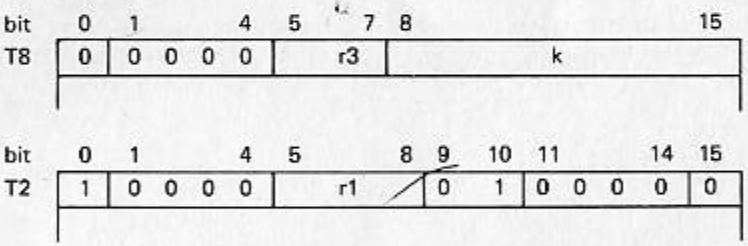
Syntax:
[label]␣ LDK␣ r3, k — T8
[label]␣ LDKL␣ r1, lk — T2

T8 The positive constant k is loaded into bits 8 through 15 of the register specified in r3. The bits 0 through 7 are reset to zero.

T2 The positive or negative constant, which can be found in the word following the instruction, replaces the contents of the register specified by r1.

Type	*Function*	*Syntax*
T8	$k \rightarrow r3_{8-15}$ $0 \rightarrow r3_{0-7}$	LDK r3, k
T2	lk → r1	LDKL r1, lk

Condition register:
T8 Unchanged
T2 CR = 0 if lk = 0
1 if lk > 0
2 if lk < 0



Remark:
Restricted to system mode if r1 = A15.

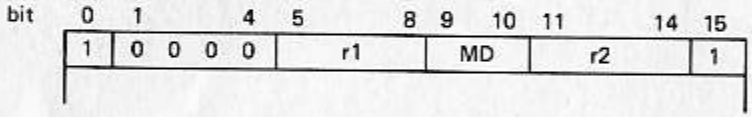
ST — *Store register* — **ST** — P851M P852M P856M P857M

Syntax: [label]␣ ST [*]␣ r1, m [, r2]

The 16 bits of the register specified by r1 replace the contents of the effective memory address.

Type	*Function*	*MD*	*Syntax*
T4	(r1) → m	10	ST r1, m
T5	(r1) → m + (r2)	10	ST r1, m, r2
T6	(r1) → (m)	11	ST* r1, m
T7	(r1) → (m + (r2))	11	ST* r1, m, r2

Condition register: Unchanged



Remark:
Restricted to system mode if r1 = A15.

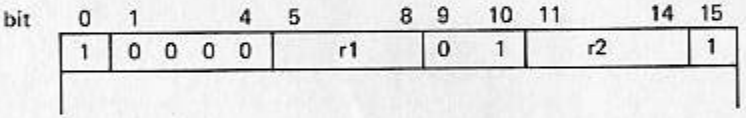
STR — *Store register/register* — **STR** — P851M P852M P856M P857M

Syntax: [label]␣ STR ␣ r1, r2

The 16 bits of the register specified by r1 replace the contents of the memory address indicated in the register specified by r2 (indirect addressing). If A15 (stack pointer) is specified by r2 it is updated.

Type	*Function*	*Syntax*
T3A	(r1) → (r2)	STR r1, r2
T3B	(r1) → (A15), (A15) − 2 → A15	STR r1, A15

Condition register: Unchanged



Remark:
* An interrupt 'stack overflow' is generated when, for T3B type, the address reached by the pointer = </100. Bit 13 is set to 1 in PSW.
* Restricted to system mode if r1 = A15 or if type 3B.

ML *Multiple load* **ML** P851M P852M P856M P857M (softw. sim

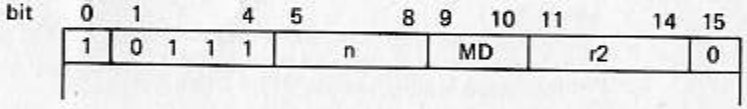
Syntax: [label]␣ ML [*]␣ n, m [, r2]

The contents of n consecutive registers (the first one being A1) are replaced by the contents of n consecutive memory locations (the first location is indicated by the effective memory address).

Type	*Function*			*MD*	*Syntax*
T4	(m) ... (m + n)	→	A1... An	10	ML n, m
T5	(m + (r2)) ... (m + (r2) + n)	→	A1... An	10	ML n, m, r2
T6	((m)) ... ((m) + n)	→	A1... An	11	ML* n, m
T7	((m + (r2))) ... ((m + (r2)) + n)	→	A1... An	11	ML* n, m, r2

n = number of registers (1 through 15)

Condition register:
CR = 0 if (A1) = 0
1 if (A1) > 0
2 if (A1) < 0



Remark:
Restricted to system mode if n = 15.

MLK *Multiple load constant* **MLK** P851M P852M P856M P857M (softw. sim

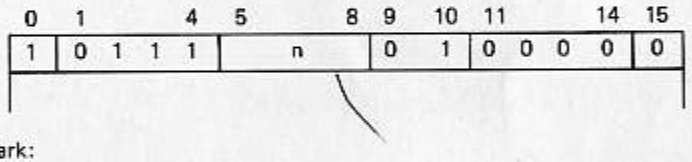
Syntax: [label] ␣ MLK ␣ n

The contents of n successive registers are replaced by n values which must be given immediately after this instruction by means of a data statement. If n = 0 the instruction is trapped.

Type	*Function*	*Syntax*
T2	Ik1, Ik2 ,..., Ikn → A1, A2 ,..., An	MLK n DATA x,...,xn

n = number of registers (1 through 15)

Condition register:
CR = 0 if (A1) = 0
1 if (A1) > 0
2 if (A1) < 0



Remark:
Restricted to system mode if n = 15.

MLR *Multiple load/register* **MLR** P851M P852M P856M P857M (softw. sim)

Syntax: [label] ␣ MLR ␣ n, r2

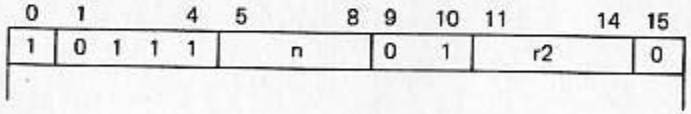
The contents of n consecutive registers (the first one being A1), are replaced by the contents of n consecutive memory locations. The first address of those locations is indicated by the contents of r2.
If r2 is the stackpointer A15, the system stackpointer is updated.

Type	*Function*		*Syntax*
T3A	((r2))	→ A1	MLR n, r2
	((r2) +2)	→ A2	
	–		
	–		
	((r2) +2n – 2)	→ An	
T3B	(A15) + 2n	→ A15	MLR n, A15
	((A15))	→ A1	
	((A15) – 2)	→ A2	
	–		
	–		
	((A15) – 2n + 2)	→ An	

n = number of registers (1 through 15)

Condition register:

CR = 0 if (A1) = 0
1 if (A1) > 0
2 if (A1) < 0



Remark:
* Restricted to system mode if n = 15 or if r2 = A15
* If 3B type, the contents must be even (P851M).

MS *Multiple store* **MS** P851M P852M P856M P857M (softw. sim)

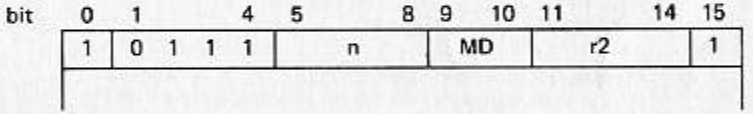
Syntax: [label] ␣ MS [*] ␣ n, m [, r2]

The contents of n consecutive memory locations (the first one is given by the effective memory address) are replaced by the contents of n consecutive registers.

Type	*Function*	*MD*	*Syntax*
T4	A1 . . An → m, . . , m + n	10	MS n, m
T5	A1 . . An → m + (r2), . . , m + (r2) + n	10	MS n, m, r2
T6	A1 . . An → (m), . . , (m) + n	11	MS* n, m
T7	A1 . . An → (m + (r2)), . . , (m + (r2)) + n	11	MS* n, m, r2

n = number of registers (1 through 15)

Condition register: Unchanged



Remark:
Restricted to system mode if n = 15.

MSR *Multiple store register* **MSR** P851M P852M P856M P857M (softw. sim

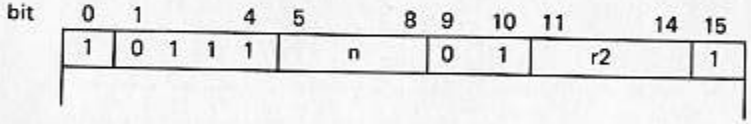
Syntax: [label] ␣ MSR ␣ n, r2

The contents of n consecutive registers (the first one is register A1) replace the contents of n consecutive memory locations. The first address of those locations is specified in r2. If r2 = the system stackpointer A15, the stack-pointer is updated by the contents of n registers.

Type	*Function*		*Syntax*
T3A	(A1)	→ (r2)	MSR n, r2
	(A2)	→ (r2) + 2	
	–		
	–		
	(An)	→ (r2) + 2n – 2	
T3B	(A1)	→ (A15)	MSR n, A15
	(A2)	→ (A15) – 2	
	–		
	–		
	(An)	→ (A15) – 2n + 2	
	(A15) – 2n	→ (A15)	

n = number of registers (1 through 15)

Condition register: Unchanged



Remark:
* An interrupt 'stack overflow' is generated when, for type T3B, the address reached by the pointer = </100. Bit 13 in PSW is set to 1.
* Restricted to system mode when n = 15 or r2 = A15.
* If 3B type, the A15 contents must be even (P851M).

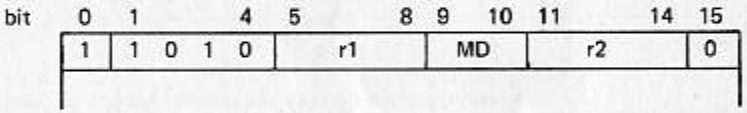
EL *Extended load (MMU option)* **EL** P857M

Syntax: [label] ␣ EL [*] ␣ r1, m [, r2]

The 16-bit contents of the effective memory address, specified in m and translated by the MMU, are loaded in register r1.

Type	*Function*		*MD*	*Syntax*
T4	(m) extended	→ r1	10	EL r1, m
T5	(m + (r2)) extended	→ r1	10	EL r1, m, r2
T6	((m)) extended	→ r1	11	EL* r1, m
T7	((m + (r2))) extended	→ r1	11	EL* r1, m, r2

Condition register:
CR = 0 if (r1) = 0
1 if (r1) > 0
2 if (r1) < 0



Remark:
This instruction may only be used in system mode.

ELR *Extended load/reg. (MMU option)* **ELR** **P857M**

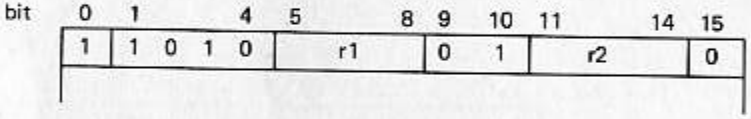
Syntax: [label] ␣ ELR ␣ r1, r2

The 16-bit contents of the effective memory address pointed to in register r2, and translated by the MMU, are loaded in register r1.

Type	*Function*
T3	((r2)) extended → r1

Condition register:

CR = 0 if (r1) = 0
1 if (r1) > 0
2 if (r1) < 0



Remark:
This instruction may only be used in system mode.

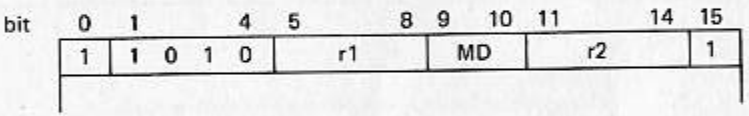
ES *Extended store (MMU option)* **ES** **P857M**

Syntax: [label] ␣ ES [*] ␣ r1, m [, r2]

The 16-bit contents of register r1 replace the contents of the effective memory address as translated by the MMU.

Type	*Function*	*MD*	*Syntax*
T4	(r1) → m, extended	10	ES r1, m
T5	(r1) → m + (r2), extended	10	ES r1, m, r2
T6	(r1) → (m), extended	11	ES* r1, m
T7	(r1) → (m + (r2)), extended	11	ES* r1, m, r2

Condition register: Unchanged



Remark:
This instruction may only be used in system mode.

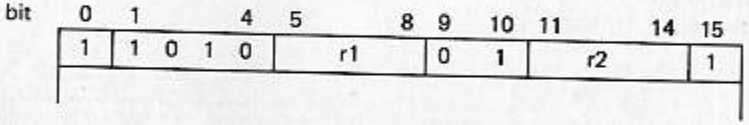
ESR *Extended store/reg. (MMU option)* **ESR** **P857M**

Syntax: [label] ␣ ESR ␣ r1, r2

This instruction replaces the contents of the memory address specified in r2, and translated by the MMU, by the 16-bit contents of register r1.

Type	*Function*
T3	(r1) → (r2) extended

Condition register: Unchanged



Remark:
This instruction may only be used in system mode.

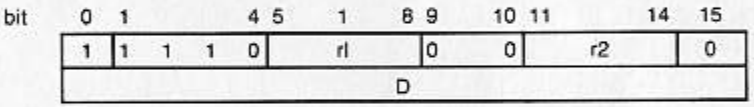
LDA *Load address* **LDA** **P853 P854 P858 P859**

Syntax: [label] LDA r1,D,r2

This instruction loads the address specified in r2, incremented by the value D from the second instruction word, into the register specified by r1.

Type	*Function*
T1	(r2) + D → r1

Condition register: Unchanged



Remark

* r1 must be ≠0
* restricted to system mode if r1 = A15