TH32 Instruction Set

(Version 0.09.0, 30 Mar. 1998)

1 Arithmetics

instruction	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
ADD	1	1	dest				0	0	1	0	0	0	0	0	0	0	operand1					operand2										
ADDC	1	1	dest				0	0	1	1	0	0	0	0	0	0	operand1					operand2										
SUB	1	1	dest				0	0	0	0	0	0	0	0	0	0	operand1					operand2										
SUBB	1	1	dest				0	0	0	1	0	0	0	0	0	0	operand1					operand2										
CMP	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	operand1					operand2										
AND¶	1	1	dest				0	1	0	0	0	0	0	0	0	0	operand1					operand2										
OR¶	1	1	dest				0	1	0	1	0	0	0	0	0	0	operand1					operand2										
XOR¶	1	1	dest				0	1	1	0	0	0	0	0	0	0	operand1					operand2										
TEST¶	0	0	0	0	0	0	0	1	0	0	0	0	0	0	0	0	operand1					operand2										
MOV¶	1	1	dest				0	1	0	1	0	0	0	0	0	0	operand					1	0	0	0	0	0	0	0	0	0	0
NOT¶	1	1	dest				0	1	1	0	0	0	0	0	0	0	operand					1	1	1	1	1	1	1	1	1	1	1
UNIHL¶	1	1	dest				0	1	1	1	0	0	0	0	0	0	operand1					operand2										
UNIHH¶	1	1	dest				0	1	1	1	0	1	0	0	0	0	operand1					operand2										
MOVHL	0	1	dest				0	1	1	1	0	1	0	0	0	0	0	0	0	0	0	operand										
SER1¶	1	1	dest				1	0	0	0	0	0	0	0	0	0	operand1					operand2										
SHR1¶	1	1	dest				1	0	1	0	0	0	0	0	0	0	0	0	0	0	0	operand										
SER8¶	1	1	dest				1	0	0	1	0	0	0	0	0	0	operand1					operand2										
SHR8¶	1	1	dest				1	0	0	1	0	0	0	0	0	0	operand					1	0	0	0	0	0	0	0	0	0	0

SER1 $d := [o1{:}o2] \to 1$

SER8 $d := [o2{:}o1] \to 8$

UNIHy $d_H := o1_H, d_L := o2_y$

MOVHL, MOVLH hazard can be removed using UNIHH, UNILL in some case.

2 Memory

instruction	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
STORE	0	0	0	0	0	0	1	1	0	1	0	0	0	0	0	0	ope1(data)					ope2(addr)										
ADDR	0	0	0	0	0	0	1	1	1	0	1	0	0	0	0	0	0	0	0	0	0	operand										
MUL20	0	0	0	0	0	0	1	1	1	1	0	0	1	1	0	0	operand1					operand2										
MUL21	0	0	0	0	0	0	1	1	1	1	0	0	1	0	0	0	operand1					operand2										
MUL22	0	0	0	0	0	0	1	1	1	1	0	1	1	1	0	0	operand1					operand2										
MUL23	0	0	0	0	0	0	1	1	1	1	0	1	1	0	0	0	operand1					operand2										
MUL30	0	0	0	0	0	0	1	1	1	1	0	0	0	1	0	0	operand1					operand2										
MUL31	0	0	0	0	0	0	1	1	1	1	0	0	0	0	0	0	operand1					operand2										
MUL32	0	0	0	0	0	0	1	1	1	1	0	1	0	1	0	0	operand1					operand2										
MUL33	0	0	0	0	0	0	1	1	1	1	0	1	0	0	0	0	operand1					operand2										
LOAD	1	1	dest				1	1	1	0	0	0	0	0	1	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
LOADH0	1	1	dest				1	1	1	0	0	0	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
LOADH	1	0	dest				1	1	1	0	0	0	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
LOADL0	1	1	dest				1	1	1	0	0	0	0	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
LOADL	0	1	dest				1	1	1	0	0	0	0	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

LOAD LOAD instruction can be combined with MUL* and ADDR by bitwise OR

3 Branch, Set

instruction	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
SETH	1	0	dest				1	1	0	0	0	0	0	0	1	0	Imm16															
SETH0	1	1	dest				1	1	0	0	0	0	0	0	1	0	Imm16															
SETL	0	1	dest				1	1	0	0	0	0	0	0	0	1	Imm16															
SETL0	1	1	dest				1	1	0	0	0	0	0	0	0	1	Imm16															
SHL16	1	1	dest				1	1	0	0	0	0	1	0	1	0	0	0	0	0	0	operand										
SHR16	1	1	dest				1	1	0	0	0	1	1	0	0	1	0	0	0	0	0	operand										
BA	0	0	0	0	0	0	1	1	0	0	1	0	0	1	0	0	Imm16															
BA	0	0	0	0	0	0	1	1	0	0	1	0	1	1	0	0	0	0	0	0	0	operand										
BN	0	0	0	0	0	0	1	1	0	0	1	0	0	0	0	0	Imm16															
BN	0	0	0	0	0	0	1	1	0	0	1	0	1	0	0	0	0	0	0	0	0	operand										
BE	0	0	0	0	0	1	1	1	0	0	1	0	0	0	0	0	Imm16															
BE	0	0	0	0	0	1	1	1	0	0	1	0	1	0	0	0	0	0	0	0	0	operand										
BNE	0	0	0	0	0	1	1	1	0	0	1	0	0	1	0	0	Imm16															
BNE	0	0	0	0	0	1	1	1	0	0	1	0	1	1	0	0	0	0	0	0	0	operand										
BG	0	0	1	0	0	1	1	1	0	0	1	0	0	1	0	0	Imm16															
BG	0	0	1	0	0	1	1	1	0	0	1	0	1	1	0	0	0	0	0	0	0	operand										
BGE,BC	0	0	1	0	0	0	1	1	0	0	1	0	0	1	0	0	Imm16															
BGE,BC	0	0	1	0	0	0	1	1	0	0	1	0	1	1	0	0	0	0	0	0	0	operand										
BL,BNC	0	0	1	0	0	0	1	1	0	0	1	0	0	0	0	0	Imm16															
BL,BNC	0	0	1	0	0	0	1	1	0	0	1	0	1	0	0	0	0	0	0	0	0	operand										
BLE	0	0	1	0	0	1	1	1	0	0	1	0	0	0	0	0	Imm16															
BLE	0	0	1	0	0	1	1	1	0	0	1	0	1	0	0	0	0	0	0	0	0	operand										
BSG	0	0	0	1	1	1	1	1	0	0	1	0	0	1	0	0	Imm16															
BSG	0	0	0	1	1	1	1	1	0	0	1	0	1	1	0	0	0	0	0	0	0	operand										
BSGE	0	0	0	1	1	0	1	1	0	0	1	0	0	1	0	0	Imm16															
BSGE	0	0	0	1	1	0	1	1	0	0	1	0	1	1	0	0	0	0	0	0	0	operand										
BSL	0	0	0	1	1	0	1	1	0	0	1	0	0	0	0	0	Imm16															
BSL	0	0	0	1	1	0	1	1	0	0	1	0	1	0	0	0	0	0	0	0	0	operand										
BSLE	0	0	0	1	1	1	1	1	0	0	1	0	0	0	0	0	Imm16															
BSLE	0	0	0	1	1	1	1	1	0	0	1	0	1	0	0	0	0	0	0	0	0	operand										
BPOS	0	0	0	1	0	0	1	1	0	0	1	0	0	1	0	0	Imm16															
BPOS	0	0	0	1	0	0	1	1	0	0	1	0	1	1	0	0	0	0	0	0	0	operand										
BNEG	0	0	0	1	0	0	1	1	0	0	1	0	0	0	0	0	Imm16															
BNEG	0	0	0	1	0	0	1	1	0	0	1	0	1	0	0	0	0	0	0	0	0	operand										

B* branch instructions have 2 branch delay slots.

4 Remarks

All Instructions are executed with 1-clock.

All instructions have 1 result latency on result-value and no latency on flags. Specified explicitly, results of instructions maked by ¶ can be used as 2nd-operand of next instruction.

Credits