



## Vereinfachter DLX Instruktionssatz

Abkürzungen:

$I(c)$ : Instruktion, die in Konfiguration  $c$  ausgeführt wird:  $I(c) = c.m_4(c.pc)$   
 $opc$  : Op-Code:  $opc(c) = I(c)[31:26]$

$R - Type(c) \leftrightarrow opc = 0^6$   
 $J - Type(c) \leftrightarrow opc \in \{000010, 000011, 111110, 111111\}$   
 $I - Type(c) \leftrightarrow (R - Type(c) \vee J - Type(c))$

$imm$  : immediate Konstante :

$$imm(c) = \begin{cases} I(c)[25 : 0] & : J - Type = 1 \\ I(c)[15 : 0] & : sonst \end{cases}$$

$sxtimm(c)$ : sign-extended immediate Konstante:

$$sxtimm(c) = \begin{cases} I(c)_{25}^6 I(c)[25 : 0] & : J - Type = 1 \\ I(c)_{15}^{16} I(c)[15 : 0] & : sonst \end{cases}$$

Lemma:  $[sxtimm(c)] = [imm(c)]$

$RS1$  : Register Source 1:  $RS1(c) = I(c)[25:21]$

$RS2$  : Register Source 2:  $RS2(c) = I(c)[20:16]$

$RD$  : Register Destination:

$$RD(c) = \begin{cases} I(c)[15 : 11] & : R - Type = 1 \\ I(c)[20 : 16] & : sonst \end{cases}$$

$SA$  : (Shift Amount) oder special purpose register address

$f_u$  : Function Code :  $f_u(c) = I(c)[5:0]$

$aluf$  : ALU funktion, steuert ALU Operation :

$$aluf(c) = \begin{cases} f_u(c)[3 : 0] & : R - Type = 1 \\ I(c)[29 : 26] & : sonst \end{cases}$$

In der folgenden Übersicht aller Instruktionen schreiben wir nicht jedesmal die Abhängigkeit von der Konfiguration  $c$  und anstatt  $gpr(RD) = x$ , schreiben wir  $RD = x$ .



## Vereinfachter DLX Instruktionssatz

### I-type Instruktionen

I[31:26]	mnemonic	Effekt
load/store		
100011	lw	$RD = m_4(RS1 + sxtimm)$
101011	sw	$m_4(RS1 + sxtimm) = RD$
01****	comp.imm	$RD = aluop(RS1, sxtimm, I[29:26])$
control		
110100	beqz	$PC = PC + (RS1 = 0 ? sxtimm:4)$
110101	bnez	$PC = PC + (RS1 \neq 0 ? sxtimm:4)$
110110	jr	$PC = RS1$
110111	jalr	$R31 = PC+4 ; PC = RS1$

### R-type Instruktionen

I[31:26]	I[5:0]	mnemonic	Effekt
000000	00****	compute	$RD = aluop(RS1, RS2, f_u)$
special moves			
000000	010000	movs2i	$RD = SA \quad (SPR[SA] \rightarrow GPR[RS1])$
000000	010001	movi2s	$SA = RS1 \quad (GPR[RD] \rightarrow SPR[SA])$

### J-type Instruktionen

I[31:26]	mnemonic	Effekt
control		
000010	j	$PC = PC + sxtimm$
000011	jal	$R31=PC+4 ; PC = PC + sxtimm$
111110	trap	spaeter
111111	rfe	spaeter



## Vereinfachter DLX Instruktionssatz

### I-type comp.imm

I[31:26]	mnemonic	Effekt
arithmetische und logische Operationen		
01 0***	comp.imm	$RD = \text{aluop}(RS1, \text{sxtimm}, I[29:26])$
01 0000	addio	$RD = RS1 + \text{sxtimm}$
01 0001	addi	$RD = RS1 + \text{sxtimm}$ (no overflow)
01 0010	subio	$RD = RS1 - \text{sxtimm}$
01 0011	subi	$RD = RS1 - \text{sxtimm}$ (no overflow)
01 0100	andi	$RD = RS1 \wedge \text{sxtimm}$
01 0101	ori	$RD = RS1 \vee \text{sxtimm}$
01 0110	xori	$RD = RS1 \oplus \text{sxtimm}$
01 0111	lhgi	$RD = \text{imm } 0^{16}$
Test und Set Operationen		
01 1***	Test Set	$RD = \text{xcc}(RS1, \text{sxtimm}, I[29:26])$
01 1000	clri	$RD = 0^{32};$
01 1001	sgri	$RD = (RS1 > \text{sxtimm} ? 0^{31}1 : 0^{32});$
01 1010	seqi	$RD = (RS1 = \text{sxtimm} ? 0^{31}1 : 0^{32});$
01 1011	sgei	$RD = (RS1 \geq \text{sxtimm} ? 0^{31}1 : 0^{32});$
01 1100	slsi	$RD = (RS1 < \text{sxtimm} ? 0^{31}1 : 0^{32});$
01 1101	snei	$RD = (RS1 \neq \text{sxtimm} ? 0^{31}1 : 0^{32});$
01 1110	slei	$RD = (RS1 \leq \text{sxtimm} ? 0^{31}1 : 0^{32});$
01 1111	seti	$RD = 1^{32};$



## Vereinfachter DLX Instruktionssatz

### R-type compute

I[31:26]	I[5:0]	mnemonic	Effekt
000000	00 ****	compute	$RD = \text{aluop}(RS1, RS2, f_u)$
arithmetische und logische Operationen			
000000	00 0000	addo	$RD = RS1 + RS2$
000000	00 0001	add	$RD = RS1 + RS2$ (no overflow)
000000	00 0010	subo	$RD = RS1 - RS2$
000000	00 0011	sub	$RD = RS1 - RS2$ (no overflow)
000000	00 0100	and	$RD = RS1 \wedge RS2$
000000	00 0101	or	$RD = RS1 \vee RS2$
000000	00 0110	xor	$RD = RS1 \oplus RS2$
000000	00 0111	lhg	$RD = RS2[15:0] \cdot 0^{16}$
Test Set Operationen			
000000	00 1000	clr	$RD = 0^{32};$
000000	00 1001	sgr	$RD = (RS1 > RS2 ? 0^{31}1 : 0^{32});$
000000	00 1010	seq	$RD = (RS1 = RS2 ? 0^{31}1 : 0^{32});$
000000	00 1011	sge	$RD = (RS1 \geq RS2 ? 0^{31}1 : 0^{32});$
000000	00 1100	sls	$RD = (RS1 < RS2 ? 0^{31}1 : 0^{32});$
000000	00 1101	sne	$RD = (RS1 \neq RS2 ? 0^{31}1 : 0^{32});$
000000	00 1110	sle	$RD = (RS1 \leq RS2 ? 0^{31}1 : 0^{32});$
000000	00 1111	set	$RD = 1^{32};$