Table 1: Instruction Translation

MIPS Instruction	Equivalent iDEA	Resolution	HW Change
Arithmetic/ Logical			
add rd, rs, rt	add rd, ra, rb	none	No
addi rd, rs, #imm16	add rd, ra, #imm11	immediate value has to be limited to 11 bits or less	Depends
addiu rt, rs, #imm16	add rd, ra, #imm11	immediate value has to be limited to 11 bits or less	Depends
addu rd, rs, rt	add rd, ra, rb	none	No
and rd, rs, rt	and rd, ra, rb	none	No
andi rd, rs, #imm16	movl rd, #imm16; and rd, ra, rb	move imm value to a register first	No
div rs, rt	not supported	none – just make sure no div in software code	No
sub rd, rs, rt	sub rd, ra, rb	none	No
subu rd, rs, rt	subu rd, ra, rb	none	No
sll rd, rt, #imm5	movl rd, #imm5; mul rd, ra, rb	none	No
sra rd, rt, #imm5	movl rd, #imm5; mul rd, ra, rb	change in hardware to pick p[33:18]	Yes
srl rd, rt, #imm5	movl rd, #imm5; mul rd, ra, rb	change in both hw and compiler needed	Yes
sllv rd, rt, rs	mul rd, ra, rb[4:0]	change in both hw and compiler needed	Yes
srlv rd, rt, rs	mul rd, ra, rb[4:0]	change in both hw and compiler needed	Yes
srav rd, rt, rs	mul rd, ra, rb[4:0]	change in both hw and compiler needed	Yes
mul rs, rt	mul rd, ra, rb	several iterations of multiply	No
multu rs, rt	mul rd, ra, rb	several iterations of multiply	No
nor rd, rs, rt	nor rd, ra, rb	none	No
or rd, rs, rt	or rd, ra, rb	none	No
ori rd, rs, #imm16	movl rd, #imm16; or rd, ra, rb	move imm value to a register first	No
xor rd, rs, rt	xor rd, ra, rb	none	No
xori rt, rs, #imm16	movl rd, #imm16; xor rd, ra, rb	move imm value to a register first	No
Program Control			
beq rs, rt, #offset16	cbeq ra, rb, #target11	assembler to grab only the offset value $< main + 0x198 >$	No
bne rs, rt, #offset16	cbne ra, rb, #target11	assembler to grab only the offset value $< main + 0x198 >$	No
bltz rs, #offset16	cblt ra, zero, #target11	assembler to grab only the offset value $< main + 0x198 >$	No
bgez rs, #offset16	cbge ra, zero, #target11	assembler to grab only the offset value $< main + 0x198 >$	No
beqz rs, #offset16	cbeq ra, zero, #target11	assembler to grab only the offset value $< main + 0x198 >$	No
bgtz rs, #offset16	cgtz ra, zero, #target11	assembler to grab only the offset value $< main + 0x198 >$	No
blez rs, #offset16	cble ra, zero, #target11	assembler to grab only the offset value $< main + 0x198 >$	No
j #target	b #target	none	No
jal #target	b #target; mov rd, pc	none	Yes
jalr rs	no equivalent inst	change in hardware needed – or make sure sw doesn't use it	No
jr rs	no equivalent inst	change in hardware needed	Yes
slt rd, rs, rt	no equivalent inst	closest is cmp ra, rb, but no way to set rd	Yes
slti rt, rs, #imm16	no equivalent inst	closest is cmp ra, #imm11, but no way to set rd	Yes
sltiu rt, rs, #imm16	no equivalent inst	closest is cmp ra, #imm11, but no way to set rd	Yes
sltu rd, rs, rt	no equivalent inst	closest is cmp ra, #imm11, but no way to set rd	Yes
Data Transfer			
lb rt, #offset16 (rs)	not supported	change data type to int in sw	No
lbu rt, #offset16 (rs)	not supported	change data type to int in sw	No
lh rt, #offset16 (rs)	not supported	change data type to int in sw	No
lhu rt, #offset16 (rs)	not supported	change data type to int in sw	No
lui rt, #offset16 (rs)	not supported	change data type to int in sw	No
lw rt, #offset16 (rs)	movl rd, #imm16; ldr rd, [ra, rb]	move offset value to a register first	No
sb rt, #offset16 (rs)	not supported	change data type to int in sw	No
sh rt, #offset16 (rs)	not supported	change data type to int in sw	No
sw rt, #offset16 (rs)	movl rd, #imm16; str rd, [ra, rb]	move offset value to a register first	No

Table 2: Composite iDEA Instructions

iDEA Instruction	Equivalent MIPS
cbeq ra, rb, #target	slt rd, rs, rt; beq rs, rt, #offset
mac	
madd	