INSTRUCTION	OP CODE				REGISTER	ADDR.MODE			REGISTER			REGISTER		
LOAD/STORE INSTRUCTIONS:														
li *r5, #100*	0	0	0	1	r5	0	0	1	x	x	x	x	x	x
	IMMEDIATE													
lr r5, r7	0	0	0	1	r5	0	1	0	r7			x	x	x
lx r5, 10(r1, r7)	0	0	0	1	r5	0	1	1	r1			r7		
	OFFSET													
ldn *r5, @ 10(r1, r7)*	0	0	0	1	r5	1	0	0	r1			r7		
	OFFSET													
stx r5, 10(r1,r7)	0	0	1	0	r5	0	1	1	r1			r7		
	OFFSET													
stdn *r5, @ 10(r1, r7)*	0	0	1	0	r5	1	0	0	r1			r7		
	OFFSET													
stx *-5(r2), r3*	0	0	1	0				1	0	1	r2		r3	
	OFFSET													
stn *@-5(r2), r3*	0	0	1	0				1	1	0	x	x	x	x	x	x
	OFFSET													
ARITHMETIC INSTRUCTIONS:														
addi *r1, #43*	0	0	1	1	r1	0	0	1	r2			r3		
	IMMEDIATE													
addr r5, r7	0	0	1	1	r5	0	1	0	r7			x	x	x
addx r5, 10(r1, r7)	0	0	1	1	r5	0	1	1	r1			r7		
	OFFSET													
adddn *r5, @ 10(r1, r7)*	0	0	1	1	r5	1	0	0	r1			r7		
	OFFSET													
subi *r1, #43*	0	1	0	0	r1	0	0	1	r2			r3		
	IMMEDIATE													
subr r5, r7	0	1	0	0	r5	0	1	0	r7			x	x	x
subx r5, 10(r1,r7)	0	1	0	0	r5	0	1	1	r1			r7		
	OFFSET													
subdn *r5, @ 10(r1, r7)*	0	1	0	0	r5	1	0	0	r1			r7		
	OFFSET													
JUMP INSTRUCTIONS:														
j addr	0	1	0	1	x	x	x	0	0	1	x	x	x	x	x	x
	ADDRESS JUMP													

jz addr	0	1	0	1	x	x	x	0	1	0	x	x	x	x	x	x
	ADDRESS JUMP															
jnz addr	0	1	0	1	x	x	x	0	1	1	x	x	x	x	x	x
	ADDRESS JUMP															
jc addr	0	1	0	1	x	x	x	1	0	0	x	x	x	x	x	x
	ADDRESS JUMP															
jnc addr	0	1	0	1	x	x	x	1	0	1	x	x	x	x	x	x
	ADDRESS JUMP															
jm addr	0	1	0	1	x	x	x	1	1	0	x	x	x	x	x	x
	ADDRESS JUMP															
jnm addr	0	1	0	1	x	x	x	1	1	1	x	x	x	x	x	x
	ADDRESS JUMP															
SUBROUTINE CALL INSTRUCTIONS:																
jal r5, sub	0	1	1	1	r5			0	0	1	x	x	x	x	x	x
	SUB															
jr r5	0	1	1	1	r5			0	0	1	x	x	x	x	x	x