Register-Register Encoding (R2)

47 42	41 36	35 29	28 22	21 15	14 8	7 6	5 0
$Func_6$	$\sim_6$	$Mask_7$	Rb_7	Ra_7	Rt_7	Fmt_2	4_6

Register-Immediate Encoding (RI)

47 37	36	35 29	28 22	21 15	14 8	7 6	5 0
$Imm_{17..7}$	S	$Mask_7$	$Imm_{6..0}$	Ra_7	Rt_7	Fmt_2	$Opcode_6$

Register-Immediate High Encoding (RIH)

4743	44 37	36	35 29	28 22	21 15	14 8	7 6	5 0
$\sim_3$	$Imm_{31..24}$	S	$Mask_6$	$Imm_{23..17}$	Ra_7	Rt_7	Fmt_2	$Opcode_6$

$Format_2$	**Rb_6**	**Ra_6**	**Rt_6**
00	Scalar	Scalar	Scalar
01	Scalar	Vector	Vector
10	Vector	Vector	Vector
11	reserved		

S	**Immediate Swap**		
0	Immed.	Ra	Rt
1	Rb	Immed.	Rt

Register Spec Field	
6	5 0
Invert	Regno

The invert bit inverts or negates a register depending on the instruction.

Instructions are a fixed 48 bits in size.

The masking register spec is present in all instructions. This makes it possible to use a two-bit format field in the instruction.

Branch Instruction Format

47 37	36 32	31 29	28 22	21 15	14 8	7 6	5 0
$Disp_{17..7}$	$\sim_5$	$Cond_3$	$Disp_{6..0}$	Rb_7	Ra_7	Fmt_2	$Opcode_6$

47 37	36	3532	31 29	28 22	21 15	14 8	7 6	5 0
$Disp_{17..7}$	S	Im_4	$Cond_3$	$Disp_{6..0}$	$Imm_{6..0}$	Ra_7	Fmt_2	$Opcode_6$

bra / bsr format

47 37	36	35 29	28 22	21 15	14 8	7 6	5 0
$Disp_{17..7}$	~	$Disp_{24..18}$	$Disp_{6..0}$	$Disp_{31..25}$	Rt_7	$\sim_2$	57_6

jal / ret format

47 37	36	35 29	28 22	21 15	14 8	7 6	5 0
$Imm_{17..7}$	~	$Imm_{24..18}$	$Imm_{6..0}$	Ra_7	Rt_7	Op_2	58_6

Fmt_2	Operands	
0	Register – Register	
1	Register - Immediate	
2,3	reserved	

Cond	Mnemonic	Comment
0	beq	
1	bne	
2	bgt / blt	Swap operands to switch between bgt and blt
3	bge / ble	
4	bgtu /bltu	
5	bgeu / bleu	
6	bbc / bbs	Invert register to check for bits set
7	reserved	

Op2	Mnemonic	Operation
0	JAL	Jump and link
1	RTD	Return and deallocate
2	RTE	Return from exception
3	JALI	Jump and link indirect

$Opcode_6/Func_6$	Mnemonic	Description
00	or / nor /orc / norc	Bitwise or
01	and / nand / andc / nandc	Bitwise and
02	syscall	System call
03	Xor / xnor / xorc / xnorc	Bitwise exclusive or
05	add / sub	
06		
07	mul / nmul	
08	mulhu	
09	asr	Arithmetic shift right
10	lsr	Logical shift right
11	asl	Arithmetic shift left
12	clz / clo	
13	shuffle	
14	ctz / cto	
16	ceq / cne	
17		
18	cgt / cle	
19	cge / clt	
20		
21		
22	cgtu / cleu	
23	cgeu / cltu	
24		
25		
26	getlane	
27	ftoi	Float to integer
28	reciprocal	
29	sextb	Sign extend byte
30	sextw	Sign extend wyde
31	mulh	
32	Fadd fsub	
33		
34	fmul	
35		
42	itof	Integer to float
43		
44	fcgt	
45	fcge	
46	fclt	
47	fcle	
48	fceq	
49	fcne	
52	orh	Or high order bits
53	andh	Bitwise and high order bits
55	xorh	Exclusive or high order bits
56	addh	Add high order bits
57	bra / bsr	Branch, Branch to subroutine
58	jal	Jump and link
62	break	
