Instrução	Tipo	Formato	Opcode	Funct	AluOp	0xOpcode	0xFunct
Add	R	rd=rs+rt	000000	100000	0	0	20
And	R	rd=rs&rt	000000	100100	2	0	24
Shift Left Logical	R	rd=rt<<sa	000000	000000	6	0	0
Shift Right Logical	R	rd=rt>>sa	000000	000010	7	0	2
Sub	R	rd=rs-rt	000000	100010	1	0	22
Or	R	rd=rs|rt	000000	100101	3	0	25
Xor	R	rd=rs⊕rt	000000	100110	4	0	26
Nor	R	rd=rs⊽rt	000000	100111	5	0	27
Set Less Than	R	rd=rs<rt	000000	101010	8	0	2A
Jump Register	R	pc=rs	000000	001000		0	8
Add Immediate	I	rt=rs+imm	001000	N/A	0	C	N/A
And Immediate	I	rt=rs&imm	001100	N/A	2	C	N/A
Or Immediate	I	rt=rs|imm	001101	N/A	3	D	N/A
Load Word	I	rt=*(int*)(offset+rs)	100011	N/A	0	23	N/A
Store Word	I	*(int*)(offset+rs)=rt	101011	N/A	0	2B	N/A
Branch on Equal	I	if(rs==rt) pc+=offset*4	000100	N/A	9	4	N/A
Branch on Not Equal	I	if(rs!=rt) pc+=offset*4	000101	N/A	9	5	N/A
Set Less Than Immediate	I	rt=rs<imm	001010	N/A	8	A	N/A
Jump	J	pc=pc_upper|(target<<2)	000010	N/A		2	N/A
Jump and Link	J	$31=pc;pc +=i<<2	000011	N/A		3	N/A

ALUOperation	OperationPos
ADD	0
SUB	1
AND	2
OR	3
XOR	4
NOR	5
SLL	6
SRL	7
SLT	8
EQUAL	9

ALUOpDecode	Por Opcode				
Instrução	Opcode	bit3	bit2	bit1	bit0
Add Immediate	001000	0	0	0	0
And Immediate	001100	0	0	1	0
Or Immediate	001101	0	0	1	1
Load Word	100011	0	0	0	0
Store Word	101011	0	0	0	0
Branch on Equal	000100	1	0	0	1
Branch on Not Equal	000101	1	0	0	1
Set Less Than Immediate	001010	1	0	0	0

	Por Funct				
Instrução	Funct	bit3	bit2	bit1	bit0
Add	100000	0	0	0	0
And	100100	0	0	1	0
Shift Left Logical	000000	0	1	1	0
Shift Right Logical	000010	0	1	1	1
Sub	100010	0	0	0	1
Or	100101	0	0	1	1
Xor	100110	0	1	0	0
Nor	100111	0	1	0	1
Set Less Than	101010	1	0	0	0
Jump Register	001000	-	-	-	-

FlagsDecode													
Instrução	Opcode	JalFlag	JRFlag	MemToReg	RegWrite	Branch	Bne	ALUSrc	ZeroExtend	MemRead	MemWrite	Jump	RegDest
Add	000000	0	0	0	1	0	0	0	0	0	0	0	1
And	000000	0	0	0	1	0	0	0	0	0	0	0	1
Shift Left Logical	000000	0	0	0	1	0	0	0	0	0	0	0	1
Shift Right Logical	000000	0	0	0	1	0	0	0	0	0	0	0	1
Sub	000000	0	0	0	1	0	0	0	0	0	0	0	1
Or	000000	0	0	0	1	0	0	0	0	0	0	0	1
Set Less Than	000000	0	0	0	1	0	0	0	0	0	0	0	1
Jump Register	000000	0	1	0	0	0	0	0	0	0	0	0	0
Add Immediate	001000	0	0	0	1	0	0	1	0	0	0	0	0
And Immediate	001100	0	0	0	1	0	0	1	1	0	0	0	0
Or Immediate	001101	0	0	0	1	0	0	1	1	0	0	0	0
Load Word	100011	0	0	1	1	0	0	1	0	1	0	0	0
Store Word	101011	0	0	0	0	0	0	1	0	0	1	0	0
Branch on Equal	000100	0	0	0	0	1	0	0	0	0	0	0	0
Branch on Not Equal	000101	0	0	0	0	1	1	0	0	0	0	0	0
Set Less Than Immediate	001010	0	0	0	1	0	0	1	0	0	0	0	0
Jump	000010	0	0	0	0	0	0	0	0	0	0	1	0
Jump and Link	000011	1	0	0	1	0	0	0	0	0	0	1	0

Register Read					
Instrução	Formato	Opcode	Funct	Rs	Rt
Add	rd=rs+rt	000000	100000	1	1
And	rd=rs&rt	000000	100100	1	1
Shift Left Logical	rd=rt<<sa	000000	000000	0	1
Shift Right Logical	rd=rt>>sa	000000	000010	0	1
Sub	rd=rs-rt	000000	100010	1	1
Or	rd=rs|rt	000000	100101	1	1
Set Less Than	rd=rs<rt	000000	101010	1	1
Jump Register	pc=rs	000000	001000	1	0
Add Immediate	rt=rs+imm	001000	N/A	1	0
And Immediate	rt=rs&imm	001100	N/A	1	0
Or Immediate	rt=rs|imm	001101	N/A	1	0
Load Word	rt=*(int*)(offset+rs)	100011	N/A	1	0
Store Word	*(int*)(offset+rs)=rt	101011	N/A	1	1
Branch on Equal	if(rs==rt) pc+=offset*4	000100	N/A	1	1
Branch on Not Equal	if(rs!=rt) pc+=offset*4	000101	N/A	1	1
Set Less Than Immediate	rt=rs<imm	001010	N/A	1	0
Jump	pc=pc_upper|(target<<2)	000010	N/A	0	0
Jump and Link	$31=pc;pc +=i<<2	000011	N/A	0	0