0x00400000	0x3C011001
0x00400004	0x34280000
0x00400008	0x8D100000
0x0040000C	0x2011000F
0x00400010	0x16300001
0x00400014	0x02118021

0x00400000	001111 00000000010001000000000001
0x00400004	001101 00001010000000000000000000
0x00400008	100011 01000100000000000000000000
0x0040000C	001000 00000100010000000000001111
0x00400010	000101 10001100000000000000000001
0x00400014	000000 10000100011000000000100001

R –Tip (opcode 000000)

J –Tip (opcode 000010 & 000011)

I –Tip (svi ostali??)

0x00400000 001111 00000000010001000000000001
0x00400004 001101 00001010000000000000000000
0x00400008 100011 01000100000000000000000000
0x0040000C 001000 00000100010000000000001111
0x00400010 000101 10001100000000000000000001
0x00400014 000000 10000100011000000000100001

Type	-31- format (bits) -0-					
R	opcode (6)	rs (5)	rt (5)	rd (5)	shamt (5)	funct (6)
I	opcode (6)	rs (5)	rt (5)	immediate (16)		
J	opcode (6)	address (26)				

0x00400000 001111 00000 00001 0001000000000001
0x00400004 001101 00001 01000 0000000000000000
0x00400008 100011 01000 10000 0000000000000000
0x0040000C 001000 00000 10001 0000000000001111
0x00400010 000101 10001 10000 0000000000000001
0x00400014 000000 10000 10001 10000 00000 100001

Instruction		Function
add	rd, rs, rt	100000
addu	rd, rs, rt	100001
and	rd, rs, rt	100100
break		001101
div	rs, rt	011010
divu	rs, rt	011011
jalr	rd, rs	001001
jr	rs	001000
mfhi	rd	010000
mflo	rd	010010
mthi	rs	010001
mtlo	rs	010011
mult	rs, rt	011000
multu	rs, rt	011001
nor	rd, rs, rt	100111
or	rd, rs, rt	100101
sll	rd, rt, sa	000000
sllv	rd, rt, rs	000100
slt	rd, rs, rt	101010
sltu	rd, rs, rt	101011
sra	rd, rt, sa	000011
srav	rd, rt, rs	000111
srl	rd, rt, sa	000010
srlv	rd, rt, rs	000110
sub	rd, rs, rt	100010
subu	rd, rs, rt	100011
syscall		001100
xor	rd, rs, rt	100110

Instruction		Opcode	Target
j	label	000010	coded address of label
jal	label	000011	coded address of label

Instruction		Opcode
addi	rt, rs, immediate	001000
addiu	rt, rs, immediate	001001
andi	rt, rs, immediate	001100
beq	rs, rt, label	000100
bgez	rs, label	000001
bgtz	rs, label	000111
blez	rs, label	000110
bltz	rs, label	000001
bne	rs, rt, label	000101
lb	rt, immediate(rs)	100000
lbu	rt, immediate(rs)	100100
lh	rt, immediate(rs)	100001
lhu	rt, immediate(rs)	100101
lui	rt, immediate	001111
lw	rt, immediate(rs)	100011
lwc1	rt, immediate(rs)	110001
ori	rt, rs, immediate	001101
sb	rt, immediate(rs)	101000
slti	rt, rs, immediate	001010
sltiu	rt, rs, immediate	001011
sh	rt, immediate(rs)	101001
sw	rt, immediate(rs)	101011
swc1	rt, immediate(rs)	111001
xori	rt, rs, immediate	001110

```
0x00400000    001111 00000 00001 0001000000000001
0x00400004    001101 00001 01000 0000000000000000
0x00400008    100011 01000 10000 0000000000000000
0x0040000C    001000 00000 10001 0000000000001111
0x00400010    000101 10001 10000 0000000000000001
0x00400014    000000 10000 10001 10000 00000 100001
```

```
0x00400000    lui $1, 0x1001
0x00400004    ori $8, $1, 0
0x00400008    lw $16, 0($8)
0x0040000C    addi $17, $0, 0xF
0x00400010    bne $17, $16, 1
0x00400014    addu $16, $16, $17
```

```
    lui $at, 0x1001
    ori $t0, $at, 0
    lw $s0, 0($t0)
    addi $s1, $0, 15
    bne $s1, $s0, L
    addu $s0, $s0, $s1
L:
```

Registers		
0	zero	Always equal to zero
1	at	Assembler temporary; used by the assembler
2-3	v0-v1	Return value from a function call
4-7	a0-a3	First four parameters for a function call
8-15	t0-t7	Temporary variables; need not be preserved
16-23	s0-s7	Function variables; must be preserved
24-25	t8-t9	Two more temporary variables
26-27	k0-k1	Kernel use registers; may change unexpectedly
28	gp	Global pointer
29	sp	Stack pointer
30	fp/s8	Stack frame pointer or subroutine variable
31	ra	Return address of the last subroutine call