ADD – Add (with overflow)

```
Operation: $d = $s + $t;
advance_pc (4);
Syntax: add $d, $s, $t
Encoding: 0000 00ss ssst
tttt dddd d000 0010 0000
```

ADDI -- Add immediate (with overflow)

```
Operation: $t = $s + imm;
advance_pc (4);
Syntax: addi $t, $s, imm
Encoding: 0010 00ss ssst
tttt iiii iiii iiii iiii
```

ADDIU -- Add immediate unsigned (no overflow)

```
Operation: $t = $s + imm;
advance_pc (4);
Syntax: addiu $t, $s, imm
Encoding: 0010 01ss ssst
tttt iiii iiii iiii iiii
```

ADDIU -- Add immediate unsigned (no overflow)

```
Operation: $t = $s + imm;
advance_pc (4);
Syntax: addiu $t, $s, imm
Encoding: 0010 01ss ssst
tttt iiii iiii iiii iiii
```

ADDU -- Add unsigned (no overflow)

```
Operation: $d = $s + $t;
advance_pc (4);
Syntax: addu $d, $s, $t
Encoding: 0000 00ss ssst
tttt dddd d000 0010 0001
```

AND -- Bitwise and

```
Operation: $d = $s & $t;
advance_pc (4);
Syntax: and $d, $s, $t
Encoding: 0000 00ss ssst
tttt dddd d000 0010 0100
```

ANDI -- Bitwise and immediate

```
Operation: $t = $s & imm;
advance_pc (4);
Syntax: andi $t, $s, imm
Encoding: 0011 00ss ssst
tttt iiii iiii iiii iiii
```

ANDI -- Bitwise and immediate

```
Operation: $t = $s & imm;
advance_pc (4);
Syntax: andi $t, $s, imm
Encoding: 0011 00ss ssst
tttt iiii iiii iiii iiii
```

BEQ -- Branch on equal

```
Operation: if $s == $t
advance_pc (offset << 2));
else advance_pc (4);
Syntax: beq $s, $t, offset
Encoding: 0001 00ss ssst
tttt iiii iiii iiii iiii
```

BGEZ -- Branch on greater than or equal to zero

```
Operation: if $s >= 0
advance_pc (offset << 2));
else advance_pc (4);
Syntax: bgez $s, offset
Encoding: 0000 01ss sss0
0001 iiii iiii iiii iiii
```

BGEZAL -- Branch on greater than or

```
equal to zero and link
Operation: if $s >= 0 $31
= PC + 8 (or nPC + 4);
advance_pc (offset << 2));
else advance_pc (4);
Syntax: bgezal $s, offset
Encoding: 0000 01ss sss1
0001 iiii iiii iiii iiii
```

BGTZ -- Branch on greater than zero

```
Operation: if $s > 0
advance_pc (offset << 2));
else advance_pc (4);
Syntax: bgtz $s, offset
Encoding: 0001 11ss sss0
0000 iiii iiii iiii iiii
```

BLEZ -- Branch on less than or equal to zero

```
Operation: if $s <= 0
advance_pc (offset << 2));
else advance_pc (4);
Syntax: blez $s, offset
Encoding: 0001 10ss sss0
0000 iiii iiii iiii iiii
```

BLTZ -- Branch on less than zero

```
Operation: if $s < 0
advance_pc (offset << 2));
else advance_pc (4);
Syntax: bltz $s, offset
Encoding: 0000 01ss sss0
0000 iiii iiii iiii iiii
```

BLTZAL -- Branch on less than zero and link

```
Operation: if $s < 0 $31 =
PC + 8 (or nPC + 4);
advance_pc (offset << 2));
else advance_pc (4);
Syntax: bltzal $s, offset
Encoding: 0000 01ss sss1
0000 iiii iiii iiii iiii
```

BLTZAL -- Branch on less than zero and link

```
Operation: if $s < 0 $31 =
PC + 8 (or nPC + 4);
advance_pc (offset << 2));
else advance_pc (4);
Syntax: bltzal $s, offset
Encoding: 0000 01ss sss1
0000 iiii iiii iiii iiii
```