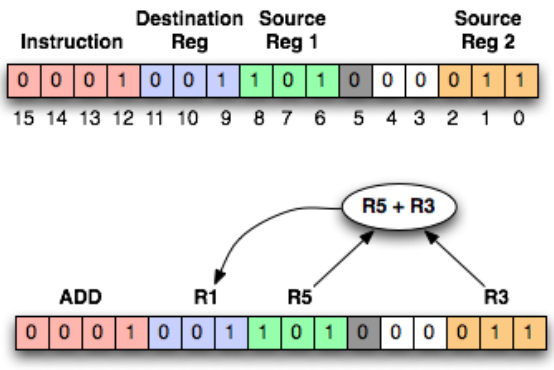
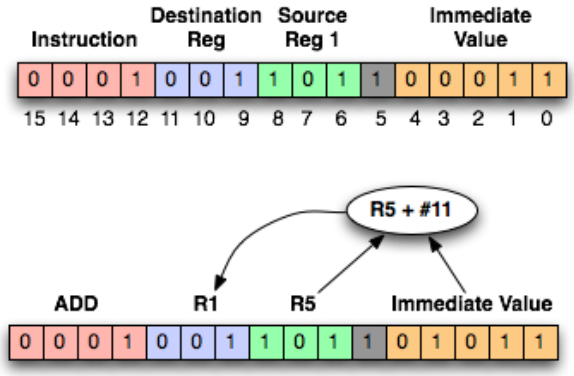
# The ADD Instruction

**The High Level**
The **ADD** instruction is a type of **Arithmetic/Logic instruction instruction** that adds two values together and stores the result in a **destination register**.




(a) The ADD Instruction – Flavor 1




(b) The ADD Instruction – Flavor 2

**Flavor 1 Breakdown**:

- Bits [15-12] specify what instruction to execute (0001 is ADD)
- Bits [11-9] specify which register to store the result in
- Bits [8-6] specify the left-hand side of the expression: **A + B**
- Bit [5] specifies which form of ADD is being used (in this case, the kind that adds two registers together)
- Bits [2-0] specify the right-hand side of the expression: **A + B**

**Flavor 2 Breakdown**:

- Bits [15-12] specify what instruction to execute (0001 is ADD)
- Bits [11-9] specify which register to store the result in

- Bits [8-6] specify the left-hand side of the expression: **A + B**
- Bit [5] specifies which form of ADD is being used (in this case, the kind that adds a register to an immediate value)
- Bits [4-0] specify the right-hand side of the expression **A + B** as an immediate value

**The Examples!**

```
ADD R1, R3, R6      ; R1 <-- R3 + R6
ADD R2, R2, R2      ; R2 <-- R2 + R2    (Protip: Doubles the value)
ADD R4, R4, #1      ; R4 <-- R4 + 1     (Protip: Increments R4 by 1)
ADD R4, R4, #-1     ; R4 <-- R4 - 1     (Protip: Decrements R4 by 1)
```

**Pitfalls... (aka: Erroneous code makes baby kittens cry)**
The example below is erroneous. Please do NOT try to code this way!

```
ADD R1, #1, R2      ; (ERROR: Immediate value must be the last operand)
ADD R1, #5, #12     ; (ERROR: No flavor allows two immediate values)
```

The first example pitfall code above is incorrect because neither of the two flavors of ADD allow for an immediate value to be the second operand given to the instruction.

The second example pitfall code above is incorrect because there are no forms of the ADD instruction that allow two immediate values to be added together. You must either put both values into registers and ADD the registers together, or put one value into a register and use the second flavor of ADD.