

Motorola Assembly Language - Introduction

Name-Field	Opcode-Field	Operand-Field	Comment-Field
[symbol[:]]	mnemonic	operands	[:text]

Symbol

- Case-sensitive
- starts at column 1
- Two types of symbols
 - o Non-local
 - First character must be alphabet, underscore (_), or a period (.) followed by alphanumeric, _ , or a .
 - If : present, ignored by assembler
 - If first character is ; or * , it is a comment line
 - Non-local symbols are valid in entire module, they cannot be reused
 - o Local
 - 2 to 4 characters in length
 - last character is always \$
 - other characters must be numeric
 - Local symbols are valid between any two non-local symbols, they can be reused

```
sum  add a1,a2
1$   ---
2$   ---
diff sub d1,d2
1$   ---
```

Mnemonics

- Case insensitive
- with extensions .b, .w, .l
- default .w
- these are Instructions or Directives

Operands

- zero operand
- one operand (represents destination)
- two operands (first represents source, second is destination)
- no spaces between operands

The data, displacement, addresses are specified via *exp*

a) Absolute expression ::= arithmetic exp

- arithmetic expression can use operators, shifts, logical operations, modulus, etc.

b) Relative expression – is a symbol

- relative expression and absolute expression can combine to form absolute or relative expression

$\text{rel_exp} + \text{abs_exp} = \text{rel_exp}$

$\text{abs_exp} + \text{rel_exp} = \text{rel_exp}$

$\text{rel_exp} - \text{abs_exp} = \text{rel_exp}$

$\text{rel_exp} - \text{rel_exp} = \text{abs_exp}$

other rules are invalid.

Valid examples for *exp* are

3

3+4

3-4

HERE+4

4+3-2+THERE

HERE-THERE

Invalid expressions are:

HERE + THERE

5 - THERE

where HERE and THERE represent two symbols