Assembly 1

The 68000 Processor is the first implementation of the 68000 16/-32 bit microprocessor architecture. The 68000 has a 16-bit data bus and 24-bit address bus while the full architecture provides for 32-bit address and data buses.

In 68000 Processor, there are eight data registers: D0-D7, which are intended to hold numbers that will have various mathematical and logical operations performed on them. And there are seven address registers: A0-A6, which are typically used as pointers. There is one active stack pointer: SP, also called A7. In user mode, SP refers to the User Stack Pointer (USP) register. The Program Counter (PC) points to the current instruction, which can also be used as a pointer in PC relative addressing modes. The Condition Code Register (CCR) consists of the lower byte of the Status Register (SR).

Each addressing mode has one assembler syntax. The 14 addressing modes, shown in the table, include 6 basic types -- register direct, register indirect, absolute, immediate, program counter relative and implied.

Figure 1. Addressing Modes

mode	**syntax**
register direct addressing	
data register direct	Dn
address register direct	An
absolute data addressing	
absolute short	(xxx).W
absolute long	(xxx).L

program counter relative addressing	
relative with displacement	d(PC)
relative with index	d(PC, Ri)
register indirect addressing	
address register indirect	(An)
address register indirect with postincrement	(An)+
address register indirect with predecrement	-(An)
address register indirect with displacement	d(An)
address register indirect with index	d(An, Ri)
immediate data addressing	
immediate	#<data>
quick immediate	
implied addressing	
implied register	SR, USP, SSP, PC

Assembly language is a kind of machine codes that human can read. Motorola, as the designer of the 68000 microprocessor family, originated its assembly language. The 68000 instruction set is very orthogonal. Most instructions can operate on all data sizes, and very few are restricted to less than three addressing modes. 68000 instructions are rather slow, but they do a lot more than instructions for the Z80 or x86 processors.

We are using assembler because it is important to know how the computer's hardware works, and we may understand some low level functions. We must recognize that most of the instructions executed on the computer or mainframe are written in assembler. And because each assembly instruction corresponds directly to a machine instruction, the programs could be extremely fast and usually small. As a result, assembler is still important and we are still using it.

68000 simulator is a kind of tools that lets people write and test assembly language programs for the 68000 processor running on people's personal computer. With this program, people can edit, assemble and simulate these programs without any 68000 hardware in different environments or operating systems.

Reference

[1] Motorola Inc. (1996). M68000 8-/16-32-Bit Microprocessor User's Manual (Ninth ed.). Prentice Hall.

[2] 68000 Assembly. (n.d.). Retrieved September 25, 2015, from https://en.wikibooks.org/wiki/68000_Assembly

[3] Thomas P. Skinner. (1988). Assembly Language Programming for the 68000 Family. John Wiley & Sons.