Pass 2

> ➤ Generate object code.

Macro processor

The assembly language programmer often finds it necessary to repeat some statements or block of code several times in program. The block may consist of code to swap sets of registers, do some arithmetic operations.

In this situation the programmer find a macro instruction facility useful. Macro instruction (often called macros) are single line abbreviations for group of instructions.

For every occurrence of this one line macro instruction in program, the macro processing assembler substitute the entire block.

Example:

Macro		
INCRMT	&A, &B	Macro
LOAD	&A	Definition
ADD	&B	
STORE	&A	
ENDM		

INCRMT	X,Y	Load X Macro ADD Y expression Store X

ENDM		Macro program.

A macro definition is placed at the start of a program, enclosed between the statements MACRO and ENDM. A MACRO statement indicates that a macro definition starts, while the statement ENDM indicates the End of a macro definition.

Issues related to the design of a macro per-processor

step 1.   Scan all macro definitions one by one, for each macro Defined

     I.   Enter its name in the macro name table (MNT).

    II.   Store the entire macro definition in the macro definition table (MDT).

    III.   Add auxiliary information to the MNT indicating where the definition of MACRO can be found in MDT.

step 2.   Examine all statements in the assembly source program to defect macro calls for each macro call