

Macros and Macro Processors

Ref: Chapter 4 of [Beck].

Macros:

- Abbreviations which are expanded by the macro processor (also called a macro preprocessor).
- Can also be used with assembly languages.
- Improve readability and facilitate program modification.

Macros in SIC/XE:

- **Examples:** See Handout 9.1.
- New directives MACRO and MEND.
- The MACRO directive specifies all the dummy parameters.
- Dummy parameters start with '&'.

Examples of macro calls: These examples use the macros defined in Handout 9.1.

```
SAVREG    LOC
GETBYT    AO,CBUF,LOC
```

Notes:

- In a macro call, the name of the macro appears in the opcode field.
- The actual parameters are specified in the operand field.
- The correspondence between dummy parameters and actual parameters is by position.
- Note the use of LC-relative addressing in macros (Example 2 of Handout 9.1). This is to avoid using labels in macro definitions. (Why?)

Design of a simple macro processor:

Assumptions:

- (a) All macro definitions appear at the beginning.
- (b) The body of a macro does not contain another macro definition or call.

High-level Outline: See Handout 9.2.

Notes regarding the outline:

- Step 1: Need to choose a data structure.
- Steps 2.1 and 2.2(b): Trivial.
- Step 2.2(a): To be discussed.

Data structure for Step 1:

- Macro Name Table (MNT): Stores the macro bodies.
- Implementation of MNT: To be discussed in class.

- Modifications to improve efficiency of macro expansion: To be discussed in class.

Carrying out expansion (Step 2.2(a)):

- See Handout 9.2.

Extending the simple design:

I. Handling use-before-definition:

- Definitions and call may occur in any order.
- Assume no nested definitions or nested calls.
- Two pass macro processor.
- Pass 1: Collect all macro definitions and construct MNT.
- Pass 2: Carry out expansion.
- The definition-before-use restriction allows one-pass design.
- The definition-before-use restriction is reasonable for macros.

II. Handling nested definitions:

- **Example:** Handout 9.3.
- To read in nested definitions:
 - Need to match MACRO and MEND directives.
 - Similar to matching parentheses in arithmetic expressions.
 - Can be done using a level counter (or a stack).
- Macro expansion may cause new macros to be defined.
- **Example:** Consider a call to MACS (or MACX) in Handout 9.3.
- **Important:** Macros RDBUF and WRBUF in Handout 9.3 *get defined only after* a call to MACS (or MACX).
- Macro processor may need to alternate between definition phase and expansion phase.

- Algorithm outlined in Figure 4.5 of [Beck].
 - Boolean flag EXPANDING to remember whether the program is handling definition or expansion.
 - Procedure GETLINE obtains next line either from the input file or from MNT depending on the value of EXPANDING.
 - Procedure PROCESSLINE calls DEFINE when a definition is seen.
 - Procedure DEFINE maintains the level counter.