Assembler & Assembly Language Programming)

Systems Programming Lab (CS-244)

Dr. Samit Bhattacharya

CSE, IIT Guwahati

Things to know

- Compiler
- Interpreter
- Assembler

COMPILER VS INTERPRETER VS ASSEMBLER

Software that converts programs written in a high level language into machine language	Software that translates a high level language program into machine language	Software that converts programs written in assembly language into machine language
Converts the whole high level language program to machine language at a time	Converts the high level language program to machine language line by line	Converts assembly language program to machine language
Used by C, C++	Used by Ruby, Perl, Python, PHP	Used by assembly language

Assembler Design

- Single pass
  - Entire conversion done with a single scan (reading) of the code
  - Not easy

- Two pass
  - Two readings of the code for conversion
  - Easier to do

Things to know

- Instruction format
- Addressing models

Instruction Format

Opcode	Operand

e.g. ADD, MUL …

How to get the operands (arguments) – Addressing modes

Addressing Modes

- The way in which the operand of an instruction is specified
- Specifies a rule for interpreting or modifying the address field of the instruction before the operand is actually executed

Example (8086 processor)

1) Immediate addressing mode
   MOV AX, 2000

2) Register addressing mode
   MOV AX, BX

3) Direct addressing mode
   MOV AX, [0500]

Two Pass Assembler

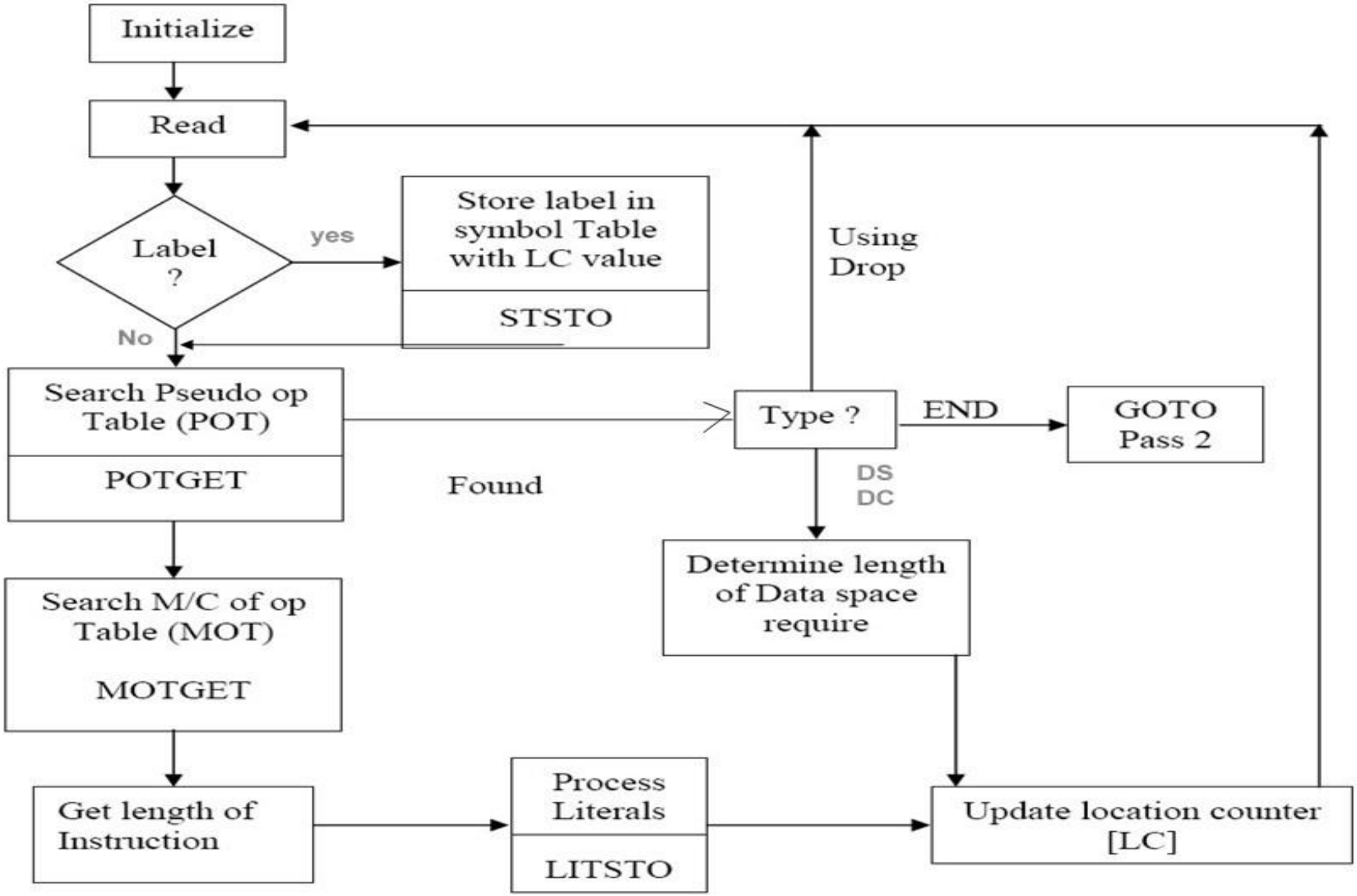
Pass 1 (Define symbol and literals)

- Determine length of machine instructions (MOTGET1)
- Keep track of location counter (LC)
- Remember values of symbol until pass 2 (STSTO)
- Process some pseudo-ops …EQU, DS
- Remember literals (LITSTO)

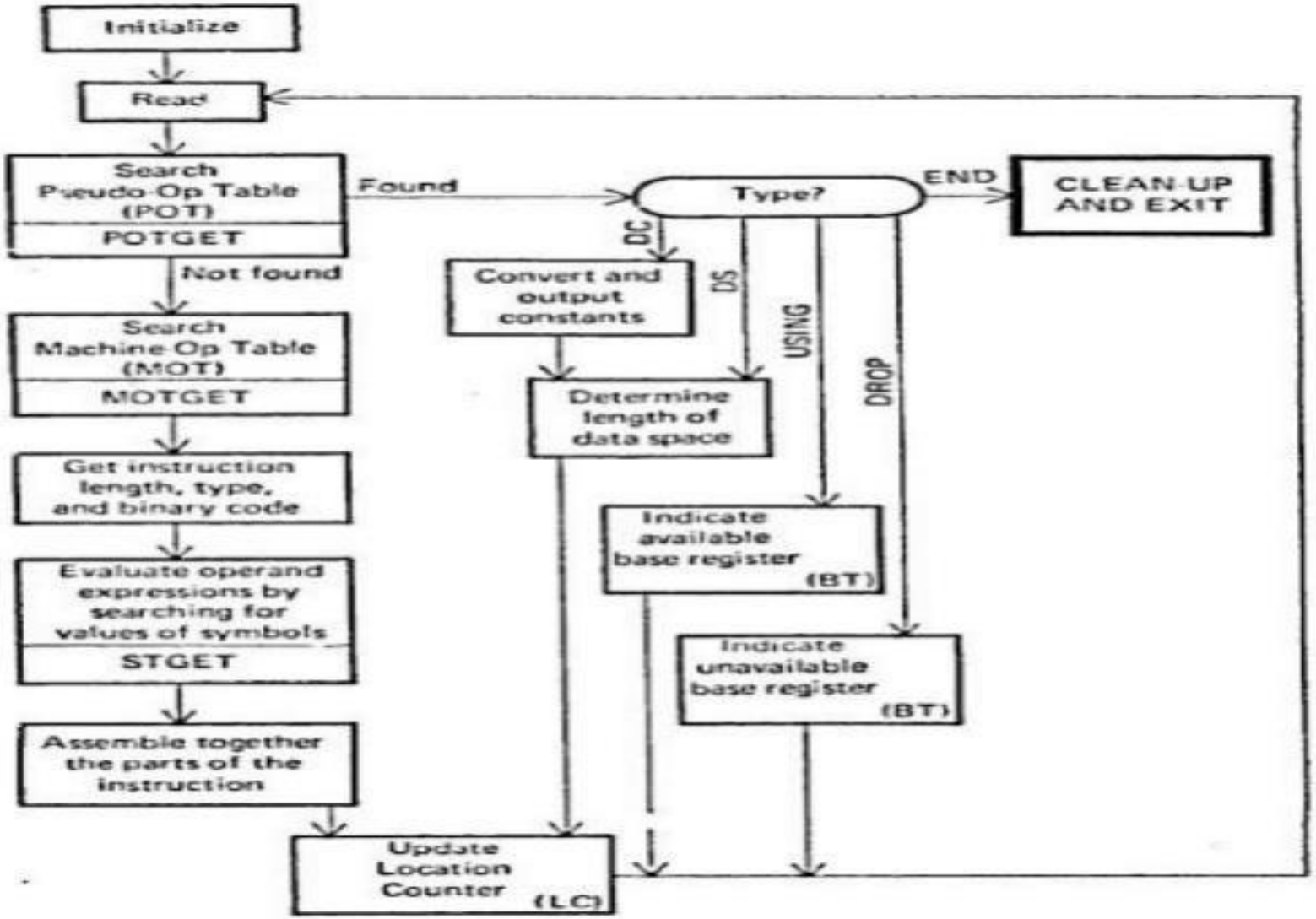
Pass 2 (Generate object program)

- Look up value of symbols (STGET)
- Generate Instructions (MOTGET2)
- Generate data for DS, DC and literals
- Process pseudo-ops (POTGET2)

Pass 1 Flowchart:



Pass 2 Flowchart:



- We will be dealing with 8086 microprocessor for this lab

- Other tutorials will cover the simpler instructions (and assignments) !!

- Book: J J Donovan, Systems Programming (very old book but useful)
- Book: L L Beck & D Manjula, Systems Software.

On 8086 instruction set:

- http://www.gabrielececchetti.it/Teaching/CalcolatoriElettronici/Docs/i8086_instruction_set.pdf
- https://www.tutorialspoint.com/microprocessor/microprocessor_8086_instruction_sets.htm

Emulator:

- We will be using **emu8086** Emulator in lab sessions for assembly language programming.