

Software Systems-1

THE ASSEMBLER- PART-2

Dr. Azam E. Al-Rawachy

COLLEGE OF COMPUTER SCIENCE AND MATHEMATICS



1 Introduction

As discussed in the previous lecture, the two-pass assembler performs two passes over the source file. In the first pass it reads the entire source file, looking only for label definitions. All labels are collected, assigned values, and placed in the symbol table in this pass. No instructions are assembled and, at the end of the pass, the symbol table should contain all the labels defined in the program. In the second pass, the instructions are again read and are assembled, using the symbol table.

Continue with two pass assembler:

BEGIN:	JMP	MAIN	EB --
	N1	DW?	
	N2	DW -6	
RES	DW	0, 0	
MAIN:	MOV	AX, N1	A1 -- --
	MUL	WORD PTR N2	F6 26 -- --
	CMP	DX, 20h	83 FA --
	JB	CONT	72 - -
	XOR	DX, DX	32 D2
CONT:	MOV	RES, AX	A3 -- --
	MOV	RES+2, DX	89 16 -- --
	RET		C3
	END		



PASS-1

Intermediate File:

Address	Mnemonic field	Error flag
0000	BEGIN: JMP MAIN	0
0002	N1 DW ?	0
0004	N2 DW -6	0
0006	RES DW 0, 0	0
000A	MAIN: MOV AX,N1	0
000D	MUL WORD PTR N2	0
0011	CMP DX, 20h	0
0014	JB CONT	0
0016	XOR DX, DX	0
0018	CONT: MOV RES,AX	0
001B	MOV RES+2, DX	0
001F	RET	0
0020	END	0

SYMTAB

Symbol	Value	Error Flag
BEGIN:	0000	0
N1	0002	0
N2	0004	0
RES	0006	0
MAIN:	000A	0
CONT	0018	0



Pass2

Listing files

Line no	Address	Mnemonic filed	Machin code	Error flag
1	0000	BEGIN: JMP MAIN	EB 08	0
2	0002	N1 DW ?	00 00	0
3	0004	N2 DW -6	FA FF	0
4	0006	RES DW 0, 0	00 00 00 00	0
5	000A	MAIN: MOV AX,N1	A1 02 00	0
6	000D	MUL WORD PTR N2	F6 26 04 00	0
7	0011	CMP DX, 20h	83 FA 20	0
8	0014	JB CONT	72 02	0
9	0016	XOR DX, DX	33 D2	0
10	0018	CONT: MOV RES,AX	A3 06 00	0
11	001B	MOV RES+2, DX	89 16 08 00	0
12	001F	RET	C3	0
13	0020	END		0

A **LABEL** is an identifier that acts as a place marker for instructions and data. A label placed just before an instruction implies the instruction's address. Similarly, a label placed just before a variable implies the **variable's address**.

DATA LABEL: Providing a convenient way to reference the variable in code.

CODE LABEL: Must end with a colon (:) character.

Object program

H PRG --- 0000 0020

T 0000 10 EB 08 00 00 FA FF 00 00 00 00 A1 02 00 F6 26 04

T 0010 10 00 83 FA 20 72 02 33 D2 A3 06 00 89 16 08 00 C3

M 000B + PRG ← PRG is the given address by the program.

M 000F + PRG

M 0019 + PRG

M 001D +PRG

E 0000



	Address	Machine Code
BEGIN	0000	EB
	0001	08
N1	0002	00
	0003	00
N2	0004	FA
	0005	FF
RES	0006	00
	0007	00
	0008	00
	0009	00
MAIN:	000A	A1
Modify	000B	02
	000C	00
	000D	F6
	000E	26
Modify	000F	04
	0010	00
	0011	83
	0012	FA
	0013	02
	0014	72
	0015	02
	0016	33
	0017	D2
CONT:	0018	A3
Modify	0019	06
	001A	00
	001B	89
	001C	16
Modify	001D	08
	001E	00
	001F	C3

The Opcode



2 The One Pass Assembler

The operation of a one-pass assembler is different. As its name implies, this assembler reads the source file once. During that single pass, the assembler handles both label definitions and assembly.

Ex1:

	PRG	ORG 100h	
	Begin	JMP main	EB --
	ar	dB 3, 2, -5, 7, 9	
	Sum	dB ?	
	Flag	dB 0	
Main:	LEA	SI, ar	8D 04 -- --
	MOV	CL,5	B1 --
	XOR	AL, AL	30 C0
LP:	ADD	AL, [SI]	8A 04
	JAE	cont	73 --
	INC	byte ptr flag	FE 06
Cont:	INC	SI	0E
	DEC	CL	FE c9
	JNZ	LP	75 --
	RET		C3
	END		



SYMTAB

Symbol	Value	Error Flag
BEGIN:	100	0
ar	102	0
sum	107	0
flag	108	0
MAIN	109	0
LP	111	0
cont	117	0

RETTAB

Symbol	Location
Main	101
Cont	114

Object program

H PRG ... 0100h 001Ch

T 0100 10 EB 07 03 02 FB 07 09 00 00 8D 04 02 01 B1 05 30

T 0110 0D C0 8A 04 73 02 FE 06 0E FE C9 75 F5 C3

E 0100

Begin→	100	EB
	101	00 07
ar→	102	03
	103	02
	104	FB
	105	07
	106	09
Sum→	107	00
Flag→	108	00
Main→	109	8D
	10A	04
	10B	02
	10C	01
	10D	B1
	10E	05
	10F	30
	110	C0
LP→	111	8A
	112	04
	113	73
	114	00 02
	115	FE
	116	06
Cont→	117	0E
	118	FE
	119	C9
	11A	75
	11B	F5
	11C	C3
	11D	

} ar

} Cont