**System Programming Lab**

**BCSE 3rd year 1st Semester**

**Name: Sayantan Biswas**

**Roll No: 001910501057**

**----------------------------------------------------------------------**

***>> I have done this assignment in C++***

* **Assignment 3**

**Q:**

Design of 8086 simulator/Assembler which supports macroprocessor using C language following the working principle of Two pass Assembler.

Group A2

**C++ Implementation for Two pass Assembler-**

Run pass2.cpp ONLY. It automatically invokes functions for pass 1 of assembler.

By default, input file is "input\_fibonacci.txt". Can be changed in pass1.cpp

Intermediate file and a modification file is maintained, which is used during the run of the program. Need not be viewed by user.

Object file is saved in "object.txt"

Listing file is saved in "list.txt"

Errors, if any, are saved in "error.txt"

**FEATURES IMPLEMENTED-**

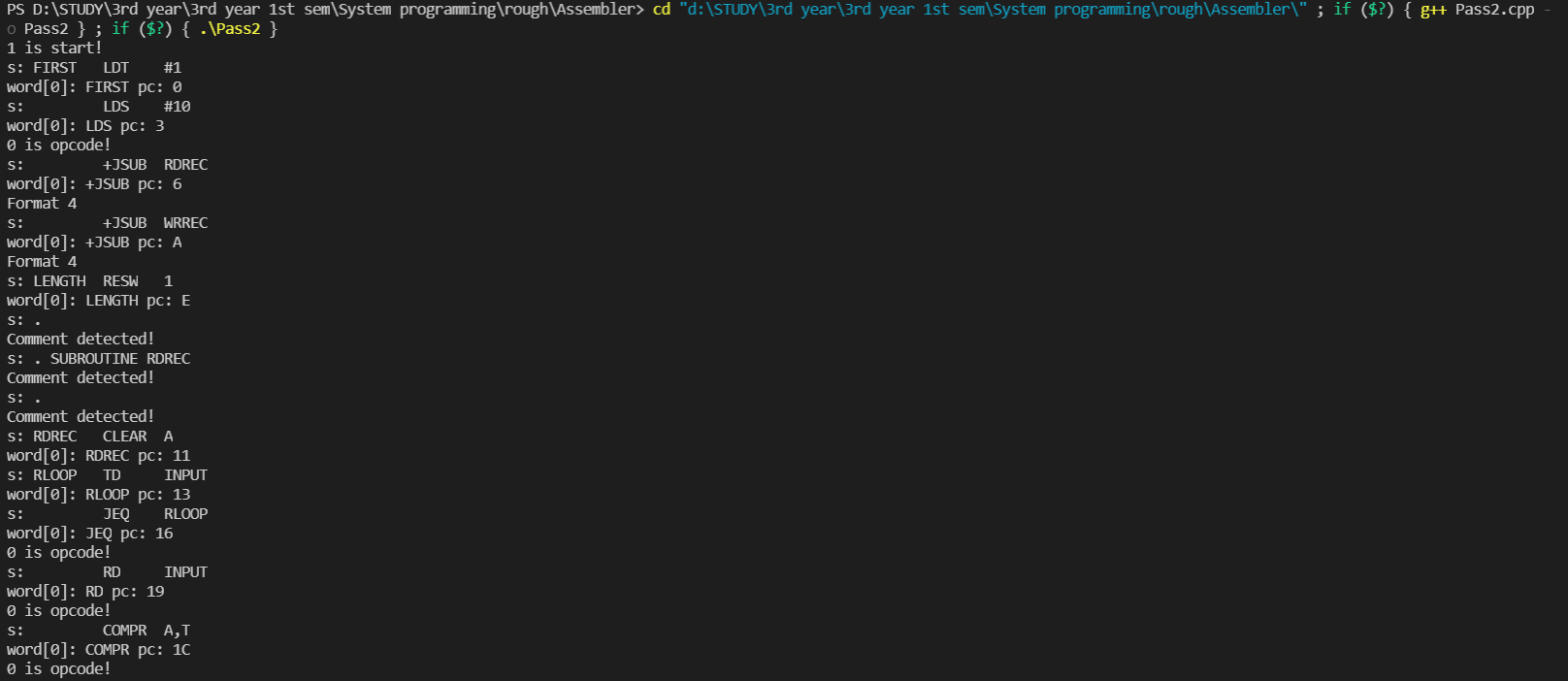
-> Different addressing modes for SIC/XE

-> Program Blocks

-> Modification Records

//-> Literals and expressions

Output:



Full ouput-

1 is start!

s: FIRST LDT #1

word[0]: FIRST pc: 0

s: LDS #10

word[0]: LDS pc: 3

0 is opcode!

s: +JSUB RDREC

word[0]: +JSUB pc: 6

Format 4

s: +JSUB WRREC

word[0]: +JSUB pc: A

Format 4

s: LENGTH RESW 1

word[0]: LENGTH pc: E

s: .

Comment detected!

s: . SUBROUTINE RDREC

Comment detected!

s: .

Comment detected!

s: RDREC CLEAR A

word[0]: RDREC pc: 11

s: RLOOP TD INPUT

word[0]: RLOOP pc: 13

s: JEQ RLOOP

word[0]: JEQ pc: 16

0 is opcode!

s: RD INPUT

word[0]: RD pc: 19

0 is opcode!

s: COMPR A,T

word[0]: COMPR pc: 1C

0 is opcode!

s: JLT RLOOP

word[0]: JLT pc: 1E

0 is opcode!

s: COMPR A,S

word[0]: COMPR pc: 21

0 is opcode!

s: JGT RLOOP

word[0]: JGT pc: 23

0 is opcode!

s: STA LENGTH

word[0]: STA pc: 26

0 is opcode!

s: RSUB

word[0]: RSUB pc: 29

0 is opcode!

s: INPUT BYTE X'F1'

word[0]: INPUT pc: 2C

s: .

Comment detected!

s: .SUBROUTINE WDREC

Comment detected!

s: .

Comment detected!

s: WRREC LDX #0

word[0]: WRREC pc: 2D

s: LDS #1

word[0]: LDS pc: 30

0 is opcode!

s: LDA #0

word[0]: LDA pc: 33

0 is opcode!

s: LDT LENGTH

word[0]: LDT pc: 36

0 is opcode!

s: WLOOP TD OUTPUT

word[0]: WLOOP pc: 39

s: JEQ WLOOP

word[0]: JEQ pc: 3C

0 is opcode!

s: WD OUTPUT

word[0]: WD pc: 3F

0 is opcode!

s: ADDR S,A

word[0]: ADDR pc: 42

0 is opcode!

s: STA VALUE1

word[0]: STA pc: 44

0 is opcode!

s: STS VALUE2

word[0]: STS pc: 47

0 is opcode!

s: LDA VALUE2

word[0]: LDA pc: 4A

0 is opcode!

s: LDS VALUE1

word[0]: LDS pc: 4D

0 is opcode!

s: TIXR T

word[0]: TIXR pc: 50

0 is opcode!

s: JLT WLOOP

word[0]: JLT pc: 52

0 is opcode!

s: OUTPUT BYTE X'05'

word[0]: OUTPUT pc: 55

s: VALUE1 RESW 1

word[0]: VALUE1 pc: 56

s: VALUE2 RESW 1

word[0]: VALUE2 pc: 59

s: END FIRST

Input for line: 5

FIBO

START

0

0

Input for line: 10

FIRST

LDT

#1

0

3

Format 3

LDT

Immediate!

a[2]: LDT:::750001

opcode: LDT:::Input for line: 15

LDS

#10

3

6

Format 3

LDS

Immediate!

a[2]: LDS:::6D000A

opcode: LDS:::Input for line: 20

+JSUB

RDREC

6

A

a[2]: +JSUB:::4B100011

opcode: +JSUB:::Input for line: 25

+JSUB

WRREC

A

E

a[2]: +JSUB:::4B10002D

opcode: +JSUB:::Input for line: 30

LENGTH

RESW

1

E

11

opcode: RESW:::Input for line: 35

$

.

Input for line: 40

$

. SUBROUTINE RDREC

Input for line: 45

$

.

Input for line: 50

RDREC

CLEAR

A

11

13

Format 2

Not Isdigit! ob2: 0

a[2]: CLEAR ob1:B4ob2:0

a[2]: CLEAR:::B400

opcode: CLEAR:::Input for line: 55

RLOOP

TD

INPUT

13

16

Format 3

TD

prgm\_ctr: 16 operand\_addr: 2C

disp: 22

a[2]: TD:::E32016

opcode: TD:::Input for line: 60

JEQ

RLOOP

16

19

Format 3

JEQ

prgm\_ctr: 19 operand\_addr: 13

disp: -6

a[2]: JEQ:::332FFA

opcode: JEQ:::Input for line: 65

RD

INPUT

19

1C

Format 3

RD

prgm\_ctr: 1C operand\_addr: 2C

disp: 16

a[2]: RD:::DB2010

opcode: RD:::Input for line: 70

COMPR

A,T

1C

1E

Format 2

a[2]: COMPR ob1:A0ob2:5

a[2]: COMPR:::A005

opcode: COMPR:::Input for line: 75

JLT

RLOOP

1E

21

Format 3

JLT

prgm\_ctr: 21 operand\_addr: 13

disp: -14

a[2]: JLT:::3B2FF2

opcode: JLT:::Input for line: 80

COMPR

A,S

21

23

Format 2

a[2]: COMPR ob1:A0ob2:4

a[2]: COMPR:::A004

opcode: COMPR:::Input for line: 85

JGT

RLOOP

23

26

Format 3

JGT

prgm\_ctr: 26 operand\_addr: 13

disp: -19

a[2]: JGT:::372FED

opcode: JGT:::Input for line: 90

STA

LENGTH

26

29

Format 3

STA

prgm\_ctr: 29 operand\_addr: E

disp: -27

a[2]: STA:::0F2FE5

opcode: STA:::Input for line: 95

RSUB

29

2C

a[2]: RSUB:::4F0000

opcode: RSUB:::Input for line: 100

INPUT

BYTE

X'F1'

2C

2D

a[2]: BYTE:::F1

opcode: BYTE:::Input for line: 105

$

.

Input for line: 110

$

.SUBROUTINE WDREC

Input for line: 115

$

.

Input for line: 120

WRREC

LDX

#0

2D

30

Format 3

LDX

Immediate!

a[2]: LDX:::050000

opcode: LDX:::Input for line: 125

LDS

#1

30

33

Format 3

LDS

Immediate!

a[2]: LDS:::6D0001

opcode: LDS:::Input for line: 130

LDA

#0

33

36

Format 3

LDA

Immediate!

a[2]: LDA:::010000

opcode: LDA:::Input for line: 135

LDT

LENGTH

36

39

Format 3

LDT

prgm\_ctr: 39 operand\_addr: E

disp: -43

a[2]: LDT:::772FD5

opcode: LDT:::Input for line: 140

WLOOP

TD

OUTPUT

39

3C

Format 3

TD

prgm\_ctr: 3C operand\_addr: 55

disp: 25

a[2]: TD:::E32019

opcode: TD:::Input for line: 145

JEQ

WLOOP

3C

3F

Format 3

JEQ

prgm\_ctr: 3F operand\_addr: 39

disp: -6

a[2]: JEQ:::332FFA

opcode: JEQ:::Input for line: 150

WD

OUTPUT

3F

42

Format 3

WD

prgm\_ctr: 42 operand\_addr: 55

disp: 19

a[2]: WD:::DF2013

opcode: WD:::Input for line: 155

ADDR

S,A

42

44

Format 2

a[2]: ADDR ob1:90ob2:40

a[2]: ADDR:::9040

opcode: ADDR:::Input for line: 160

STA

VALUE1

44

47

Format 3

STA

prgm\_ctr: 47 operand\_addr: 56

disp: 15

a[2]: STA:::0F200F

opcode: STA:::Input for line: 165

STS

VALUE2

47

4A

Format 3

STS

prgm\_ctr: 4A operand\_addr: 59

disp: 15

a[2]: STS:::7F200F

opcode: STS:::Input for line: 170

LDA

VALUE2

4A

4D

Format 3

LDA

prgm\_ctr: 4D operand\_addr: 59

disp: 12

a[2]: LDA:::03200C

opcode: LDA:::Input for line: 175

LDS

VALUE1

4D

50

Format 3

LDS

prgm\_ctr: 50 operand\_addr: 56

disp: 6

a[2]: LDS:::6F2006

opcode: LDS:::Input for line: 180

TIXR

T

50

52

Format 2

5

Not Isdigit! ob2: 50

a[2]: TIXR ob1:B8ob2:50

a[2]: TIXR:::B850

opcode: TIXR:::Input for line: 185

JLT

WLOOP

52

55

Format 3

JLT

prgm\_ctr: 55 operand\_addr: 39

disp: -28

a[2]: JLT:::3B2FE4

opcode: JLT:::Input for line: 190

OUTPUT

BYTE

X'05'

55

56

a[2]: BYTE:::05

opcode: BYTE:::Input for line: 195

VALUE1

RESW

1

56

59

opcode: RESW:::Input for line: 200

VALUE2

RESW

1

59

5C

opcode: RESW:::Input for line: 205

END

FIRST

5C

INPUT FILE ASSEMBLED SUCCESSFULY!!

----------------------------