CLASS:S.E. COMP **SUBJECT: MPL**

EXPT. NO.: 05
ROLL NO.: 21129

TITLE: Data block Transfer

PROBLEM STATEMENT:

Write x86 ALP to perform non-overlapped and overlapped block transfer (with and without string specific instructions). Blocks containing data can be defined in the data segment.

OBJECTIVE :

To learn

- Overlapped / Non – overlapped data transfer in segments
- Block transfer instruction of 8086
- Data storage in the memory and segments

OUTCOME:

Students will study different block transfer instructions and also understand block transfer within different segments.

Prerequisites: Instruction set of 80386

THEORY :

One of the frequent operations used in programming is shifting the data from one memory location to another memory location. These operations can be with simple mov instructions which may result in more number of operations. We can make use of instructions like MOVSB to transfer the data. The relevant instructions are LOOP / MOVSB.

The data can be transferred either in overlapped fashion or non overlapped Fashion. In case of overlapped addresses the two possible situations are for the Source address to be greater than the destination address in which case the first element in the source is to be moved first or for the source address to be less than the destination address which requires the last element of the source to be moved first.

Algorithm:

A] Overlapping

1. Study system calls to read and display characters on the screen.

2. Accept the Value of „N" i.e. how many numbers to add
3. initialize Sum =0 4. Read a number (two digit)
5. add it to sum
6. repeat the steps 4 and 5 to add all N numbers
7. Print the result /Sum
8. End.

B] Non-Overlapping

1. Declare a source array.
2. Load the address of source array in one of the registers. (Index register)
3. Read the first byte from the source array.
4. Increment the pointer/SI by the length of the array which becomes the starting Address of the destination array.
5. Move the source element to the destination address.
6. In case of overlapping mode based on the destination address either move the first Element or last element in the beginning of transfer operation.

Test Cases Executed:

Before overlapped :

0000000000402000 11

0000000000402001 22

0000000000402002 33

0000000000402003 44

0000000000402004 55

After overlapped :

000000000040204D 11

000000000040204E 22

000000000040204F 33

0000000000402050 44

0000000000402051 55

CONCLUSION:

We have studied different block transfer instructions and also understood block transfer within different segments