The String Instructions

Chapter 11

By Rezwana Reaz Rimpi

Modified by (slightly): Mahjabin Nahar

General form of String Instructions

Instruction	Byte form	Word form	Source	Destination
Move String	MOVSB	MOVSW	DS:SI	ES:DI
Load String	LODSB	LODSW	DS:SI	AL or AX
Store String	STOSB	STOSW	AL or AX	ES:DI
Compare String	CMPSB	CMPSW	DS:SI	ES:DI
Scan String	SCASB	SCASW	AL or AX	ES:DI

CLD: Clear Direction FLAG; sets DF = 0;

STD: sets DF = 1;

After execution of each of above instruction SI and DI automatically increased (if CLD is called) or decreased (if STD is called)

For byte instructions SI and DI increased/decreased by 1 byte

For word instructions SI and DI increased/decreased by 2 byte

Copy String1 to String2

```
.data
Str1 db 'Hello'
Str2 db 5 DUP(?)
.code
Main proc
    Mov ax, @data
                          DS:SI points
    Mov ds, ax
                           to source
    Mov es, ax
                             string
    Lea SI, str1
                            ES:DI points to
    Lea DI, str2
                           destination string
    CLD
    movsb
                 You can replace
    movsb
                 these 5 lines just
                 with 2 lines
    movsb
    movsb
                                             Mov cx, 5
                               Equivalent to
                 Mov cx, 5
                                             Copy:
    movsb -
                 REP Movsb
                                                Movsb
Main endp
                                                 Loop copy
```

Copy String1 to String2 in reverse order

```
.data
Str1 db 'Hello'
Str2 db 5 DUP(?)
.code
Main proc
                             DS: SI points to
    Mov ax, @data
                            the last element
                                 of str1
    Mov ds, ax
    Mov es, ax
                           ES:DI points
     Lea SI, str1 + 4
                              to str2
    Lea DI, str2
    STD
                               Moves str1[4] to
                                   str2[0]
    Mov cx, 5
                            * decrements both SI
    Copy:
                                 and DI by 1
       Movsb
       Add DI, 2
                            We need to decrement SI but increment DI
    Loop copy
                                                by 1
                                           * add 2 to DI
Main endp
```

Use of Lodsb and Stosb

Input str : ABC Output str : DEF

```
.data
Str1 db 'ABC'
Str2 db 3 DUP(?)
.code
Main proc
Mov ax, @data
Mov ds, ax
Mov es, ax
Lea SI, str1
Lea DI, str2
CLD
```

```
Moves st
Str_loop:

lodsb
add al, 3
stosb

Loop str_loop

Main endp
End main
```

Moves str1[SI] to AL

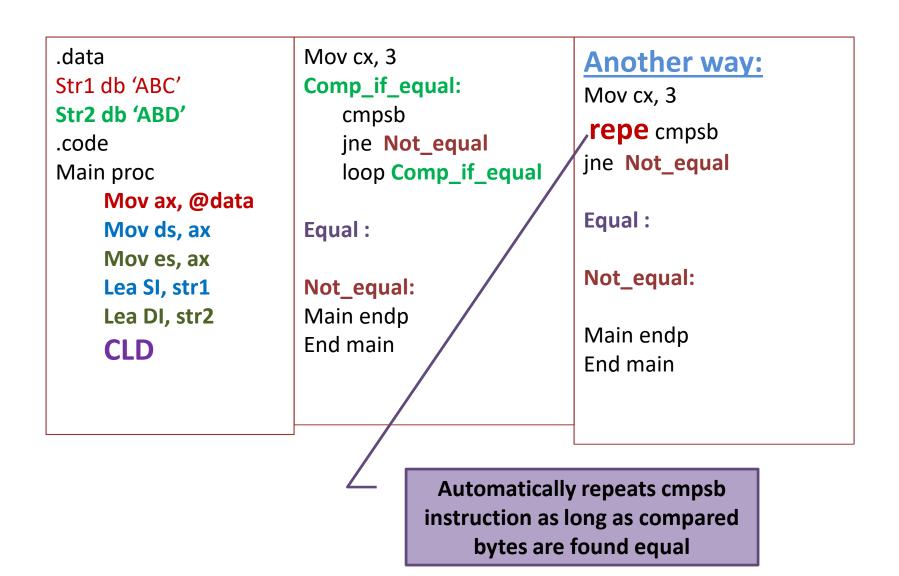
Moves AL content to str2[DI]

Conditional REP Instruction

REPE: Repeat while equal

• REPZ: Repeat while zero

Compare 2 strings: Use of cmpsb, repz, repe



Use of scasb

Compares an element pointed to by ES: DI with AL (if scasb is used) or AX (if scasw is used)

