**SSN College of Engineering Department of Computer Science and Engineering**

**III year - UCS1512 – Microprocessors Lab**

**Case conversion**

**Exp No:** 08

**Name:** Rahul Ram M

**Register Number:** 185001121

**Date:** 13/10/2020

**Aim:**

To design 8086 program for Case conversion.

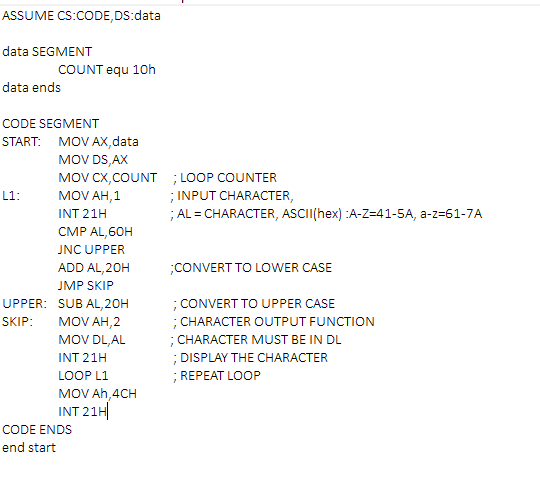
**Procedure for executing MASM:**

1. Run Dosbox and mount your masm folder to a drive in dosbox.
2. Goto the mounted drive.
3. Save the 8086 program with extension .asm in the same folder using command “edit”
4. After creating the file, assemble it using the command “masm filename.asm”
5. Link the file using the command “link filename.obj;”
6. Use debug command with filename.exe to execute and analyse the memory contents, “debug filename.exe”.
7. In debug, command “u” will display the unassembled code.
8. Use command “d segment:offset” to see the content of memory locations starting from segment:offset address.
9. To change the value in memory, use the command “e segment:offset”
10. Verify the memory contents to ensure the updates (using command “d”).
11. . Execute using the command “g” and check the outputs.
12. “q” to exit from debug and “exit” to exit from command prompt and to close the Dosbox.

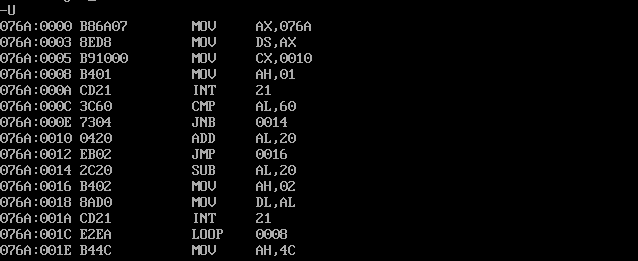
**Algorithm:**

1. START: Move the starting address of data segment to AX register and move the data from AX register to DS register.
2. Move the value of COUNT to CX register.
3. L1: Move 1h to AH register. When int 21H is called with value 1 in AH register will read the letters with echo.
4. Now compare AL and 60H using CMP.
5. If the value in AL register is greater than 60H which means the input character is a lower-case letter so we jump to UPPER using JNC.
6. If the value is less than 60H means the given character is an upper-case letter so we add 20H (32) to AL register to convert it to lower case letter and make an unconditional jump to SKIP using JMP.
7. UPPER: Subtract the value of AL register by 20H to convert the lower-case letter to upper case letter.
8. SKIP: Move 2h to AH register.
9. Move the contents of the AL register to DL to register.
10. When int 21H is called with 2 in AH register the contents in the DL register is displayed to the standard output device.
11. Loop to L1 till CX register becomes 0.
12. Move the hexadecimal value 4C into AH register. INT 21H means invoke the interrupt identified by the hexadecimal number 21. In MS-DOS, invoking interrupt 21h while AH = 4Ch causes the current process to terminate and uses the value of register AL as the exit code of the process.

**Program:**



|  |  |  |
| --- | --- | --- |
|  | **Program** | **Comments** |
| START: | MOV AX, DATA  MOV DS, AX | Transferring the data from DATA to AX register and  from AX register to DS register. |
| MOV CX, COUNT | CX <- COUNT. LOOP COUNTER. |
| MOV AH, 1 | AH <- 1. INPUT CHARACTER. |
| INT 21H | AL = CHARACTER, ASCII (hex): A-Z=41-5A, a-z=61-7A |
| CMP AL, 60H | Compare AL and 60h. |
| JNC UPPER | Jump to UPPER if value in AL is greater than 60H. |
| ADD AL, 20H | AL <- AL + 20H. CONVERT TO LOWER CASE. |
| JMP SKIP | Jump to SKIP. |
| UPPER: | SUB AL, 20H | AL <- AL – 20H, CONVERT TO UPPER CASE. |
| SKIP: | MOV AH, 2 | AH <- 2h, CHARACTER OUTPUT FUNCTION. |
| MOV DL, AL | DL <- AL. CHARACTER MUST BE IN DL. |
| INT 21H | DISPLAY THE CHARACTER. |
| LOOP L1 | REPEAT LOOP. |
| MOV AH, 4CH  INT 21H | Terminates the program. |

**Unassembled Code:**

**Snapshot of sample input and output:**



**Result:**

ShapeThus the 8086 program for Case conversion is executed successfully in DOS-BOX