

Experiment-6

Aim:- Generate a Geometric Progression (GP) Series.

Algorithm:

- 1) Assign source index a value to point at an address in RAM (let's say '0100:2000' or 2000h).
- 2) Store the value at address in RAM to the AL register (the first term), in next address in RAM to the DL register (common difference between terms) and to adjacent address in RAM to the CX register (number of terms).
- 3) Assign source index a value to point at an address in RAM to store the value of Arithmetic series (usually next row in memory window).
- 4) Create a loop 'gp', which begins with storing the value in the AL register to the address in RAM pointed by updated source index.
- 5) Multiply the values in AL and DL register and stores the result in AL register.
- 6) Increase the value of source index, moving to next address in RAM.
- 7) Loop 'gp' ends when it reaches the value of CX and the program is halted.

Program:

```
; You may customize this and other start-up templates;  
; The location of this template is c:\emu8086\inc\0_com_template.txt  
  
org 100h  
  
; add your code here  
mov si,2000h  
mov al,[si]  
mov dl,[si+1]  
mov cx,[si+2]  
mov si,2010h  
gp:  
mov [si],al  
mul dl  
inc si  
loop gp  
hlt
```

Observation:

The screenshot displays the 8086 emulator interface with three main windows:

- Random Access Memory:** Shows a memory dump starting at address 0700:0100. The data at 0700:0102 is highlighted in blue, showing the value 20 032 SPA.
- emulator: noname.com_:** Contains a toolbar with icons for Load, reload, step back, single step, and run. Below the toolbar, the registers window shows the AX register containing 00 00. The memory window shows the same memory dump as the first window, with the instruction at 07102: 20 032 SPA highlighted in blue.
- original source c...:** Displays the assembly code for the program. The instruction `mov si, 2000h` is highlighted in yellow, corresponding to the instruction at 07102: 20 032 SPA in the memory window.

