Implementation of text Scrolling in Megha O**S**

Line Processor and VGA Text Mode Driver

17th November 2019

The user interacts with Megha via text and the Operating System reports back also in text. So it is only logical for me to think about how text should behave on screen. Due to my exposure on Linux terminals and having seen the Linux 0.01 source, I inclined to having a ‘Terminal/Line Processor’ module which when provided with an array of characters will display it on the screen, parsing few special characters like Line Feed, Carriage Return, Tabs, Backspace along the way.

This ‘Terminal’ module talks directly with the VGA driver and commands it to perform certain tasks, based on the user input and the cursor location.

### VGA Driver Basics

The VGA driver in mind should be a monolithic module that takes input and writes directly to the VGA CRT Controller Card. It however provides some very important functions that modules at a later stage and use.

* Cursor positioning and maintaining the memory location to be written next.
* Routines for writing, setting cursor attributes, setting text attributes, Scroll up and Scroll Down.

However, it ***do not interpret characters*** and ***certainly not scroll on its own.*** These responsibilities lies with the ‘Terminal’ implementation.

### VGA Memory

IBM PC compatibles have about 128 KiB of memory mapped for Video (I do not know the whole history but would like to know). Memory starts from ***0xA0000*** and ends at ***0xBFFFF***. In Text mode, the memory which is mapped for VGA starts from ***0xB8000*** and ends at ***0xBFFFF***.

|  |  |  |  |
| --- | --- | --- | --- |
| Memory start | Memory End | Size | Description |
| 0xA0000 | 0xB7FFF | 96 KiB | Used primarily for Mode 3 – 320x200 256 color VGA Graphics mode. (Requiring 64 KB of space).  Thus space for only 1 page, if both text mode and graphics modes are used. |
| 0xB8000 | 0xBFFFF | 32 KiB | Used for 80x25 text mode (2 bytes per character).  Thus 32 KiB/(80 x 25 x 2) ~ 8 pages worth of memory. |

Ignoring the part used for Graphics mode, we can use 32 KibiBytes of space in text mode. However we only require 4 Kibibytes to display 80x25x2 characters and their attributes on the screen. Thus there are about 8 screen/pages worth of memory available.