Phase 0 – Umps2 – OS project

This program is a chat between two terminal, that permits an interactive message exchange and the storage of data on a common disk device.

The communication must start from the **terminal 0.** When a '>' appear on a terminal (for example on the terminal number 0), it indicates the turn of this terminal write the message to be sent at the other terminal.

To send a save request of a value or a phrase at the other terminal, you can insert the number 0 in the terminal. After the orther terminal inserted the message, head and sector of the disk, you can decide to save the message (1) or not to save it (0).

TERMINAL DEVICE

Input and output operation on the terminal are accomplished by:

- term_putchar(c,terminal): write a single char c on the selected terminal;
- term_getchar(terminal): that read a single char from the selected terminal.

Functions term_puts(string,terminal) and readline(buffer, count, terminal), simply uses the atomic functions above.

DISK DEVICE

Once the program has enough information (data to be saved, head and sector number), function store(data0,head,sect) is invoked and the actual storage operation is done.

After that, in order to check the result, the saved value is read with the function read(head,sect).

For simplicity, all read and write operation used cylinder 0, nevertheless a function to seek a specified cylinder is implemented: seekCyl(cyl).

The device "disk0.umps" is a default device with 32 cylinders, 2 heads, 8 sectors.

To use another disk device (with different characteristics) is necessary to modify MIN_HEAD, MAX_HEAD, MIN_SECT, MAX_SECT in main.c, these represents the range of heads and sectors in the current disk device. Any input of these value out of range is detected by the program.

COMPILATION PROCESS

To compile the program is necessary to run 'make -f Makefile-templ' in the '/src' directory.

To create another disk device run the code: 'umps2-mkdev - D <diskfile.mps> [CYL [HEAD [SECT [RPM [SEEKT [DATAS]]]]]]'