Project report

Afonso Pereira 201505870
David Cunha 201604317
João Loueiro 201604453

We have implemented a poll-based device driver as well as an interrupt driven one with all the features enumerated up to Lab 4. Furthermore, we have implemented 7 of the enhancements described in Lab 5: **1)**we have minimized the use of global variables **2)** it uses semaphores to eliminate race conditions **3)** it is able to avoid block or busy waiting using the O_NONBLOCK flag **4)** ioctl operations were added **5)** it allows the user to interrupt a process inside a read syscall **6)** acess control was implemented in order to prevent more than one "user" to access the serial port **7)**it supports the select/poll system call.
It is worth mentioning that, although the seri implementation allows acess to 4 device drivers, only seri0 has updates regarding received and transmitted information, being the only one operational.

Enhancements:
3.1) The usage of global vairables was minimized.
3.2) We eliminated race conditions using semaphores (line 246).
3.3) On the condition statement beginnig in line 252 we honor the O_NONBLOCK flag.
3.4) We created the function seri_ioctl, lines 152-215, which implements iotctl operations. The message received and returned by the ioctl is composed of 24 bits with the format: DLM DLL LCR. The implemented operations are as follows: to get and set the bitrate, the number of informations bits, parity and number of stop bits.
3.5) In line 259, the posibility for user to terminate reading is implemented.
3.6) By implementing the condition statement in line 126, we prevent more than user to open the device.
3.8) Select/poll operations were added in function seri_poll (lines 348-424).