Informe TPE PC

***ACLARACIONES GENERALES***

El trabajo fue realizado en mayor medida en el lenguaje C para facilitar la programación de las funciones utilizadas a lo largo de todo el TPE. Sin embargo, fue necesario recurrir al lenguaje Assembler para realizar aquellas funciones que solo pueden programarse en este lenguaje.

Se realizo una separación del kernel-space y el user-space de modo tal que las funciones que interactúen con elementos de hardware como el Timer Tick o aquellas que se encarguen de manejar interrupciones o excepciones se encuentren en el kernel-space. Por otra parte, en el user-space se encuentran aquellas funciones con las que si interactúa el usuario: imprimir un mensaje, jugar al pong, entre otras.

***DIFICULTADES***

Por querer hacer las interrupciones con el formato de INT80 de Linux se terminan pasando parámetros que no son necesarios y hacen que todo sea un poco menos claro. Relacionado con esto, se podrían hacer distintos dispatchers de system calls en el kernel-space a fin de mejorar un poco el código y modularizar mejor todas las funcionalidades.

En cuanto al manejo de excepciones, hubo problemas inicialmente con como retornar al Shell luego de haber capturado una excepción. Si bien esto se pudo solucionar, no estamos seguros de que se “acomode” el stack después de cada interrupción que se lanza. Otro problema de excepciones, en particular la de división por cero, es que la rutina de atención falla en algunas situaciones y no hemos podido identificar la raíz de este error. Creemos que puede ser por el problema mencionado con el stack, pero como la excepción de opcode parece funcionar correctamente todo el tiempo, no estamos seguros.

Si bien el driver de teclado fue programado para poder identificar combinaciones de teclas y discernir entre mayúsculas y minúsculas, la fuente utilizada no refleja esta funcionalidad ya que se ve todo en mayúsculas, esto es algo que se podría mejorar sobre todo porque genera conflictos con los scancodes que se cargan en el buffer y los que se muestran en la Shell, ya que si bien el usuario, por ejemplo, escribe “Help”, vera “HELP”, y al querer correr dicho comando no será ejecutado porque la Shell no reconoce “Help”, solo “help”.

El driver de video también resultó complicado de hacer, sobre todo para el funcionamiento del pong. La forma de solucionarlo esto hace que el juego fluya bastante bien, pero igualmente los periodos de actualización son bastante altos. Cada refresh de pantalla tarda bastante tiempo lo que hace que sea difícil testear cuantos ticks serían necesarios para que el juego funcione de la misma manera en una computadora que procese más rápidamente los refresh de pantalla. Igualmente se puede modificar la velocidad del juego con las constantes definidas en el programa.

Con respecto al método de actualización del pong, inicialmente se intentó mandar una matriz que represente el campo de juego con caracteres que representen la pelota, las paletas y los bordes. Esta opción fue descartada pues era aún más pesado en tiempos de ejecucíon que el método que se termino implementando, que actualiza solamente las paletas y la pelota.

El beep fue algo difícil de implementar debido a la poca documentación sobre su funcionamiento. Por dicho motivo, se utilizó código de internet y se lo adapto a nuestras necesidades pero ocasionalmente falla o hace un doble beep.

Para terminar, hay una interrupción INT21 que no se terminó de implementar. Mediante esta interrupción se puede pasar de modo texto a modo video y viceversa, de esta manera se podría utilizar todo el sistema operativo en modo texto, (que, por cierto, resultaría mucho más ligero) y solamente utilizar el modo video para funciones como el pong, pero por falta de tiempo y documentación no pudo completarse su implementación.

***MANUAL DE USUARIO***

Requerimientos: Docker, qemu, nasm.

Instrucciones: Basta con posicionarse en el directiorio raíz del proyecto y ejecutar la siguiente línea:

./Docker.sh nombre-de-usuario-docker

Esto iniciará el kernel. Si todo funciona correctamente se iniciara la Shell y se mostrara un mensaje de ayuda, de todas maneras se puede volver a acceder a este mensaje utilizando el comando HELP.

***FUNCIONALIDADES***

* Pong
* Ayuda
* Fecha/Hora del sistema
* Testeo de excepciones
* Testeo de beep

***BIBLIOGRAFÍA***

Gran parte de la documentación utilizada en el TPE fue obtenida de las páginas www.stanislavs.org y de wiki.osdev.org