



DEPARTAMENTO
DE COMPUTACION

Facultad de Ciencias Exactas y Naturales - UBA

Trabajo Práctico 3

System Programming

Organización del Computador II
Primer Cuatrimestre de 2016

Grupo: Yo no manejo el raiting, yo manejo un Rolls-Royce

Integrante	LU	Correo electrónico
Luis Enrique Badell Porto	246/13	luisbadell@gmail.com
Nicolas Bukovits	546/14	nicko_buk@hotmail.com
Kevin Frachtenberg	247/14	kevinfra94@gmail.com



Facultad de Ciencias Exactas y Naturales

Universidad de Buenos Aires

Ciudad Universitaria - (Pabellón I/Planta Baja)

Intendente Güiraldes 2160 - C1428EGA

Ciudad Autónoma de Buenos Aires - Rep. Argentina

Tel/Fax: (54 11) 4576-3359

<http://www.fcen.uba.ar>

Resumen

En el siguiente trabajo práctico, se busca hacer una implementación de un sistema operativo básico para correr en el programa Bochs. Las decisiones tomadas por los integrantes, se encuentran plasmadas en este informe.

Índice

1. Resolucion

1.1. GDT Parte 1

En este ejercicio, se completa la GDT con 4 segmentos, dos para datos y código para el kernel y otros dos para usuario. Por restricciones del TP, el primer índice a utilizar en la GDT es el 4, contando desde 0 y los segmentos deben direccionar los primeros 878 MB de memoria. Para pasar a modo protegido, lo que hicimos fue, cargar el registro GDTR mediante la instrucción LGDT, habilitar el bit de PE de CR0 y hacer un far jump usando el segmento de código de nivel 0 de la GDT, posteriormente se cargan los registros CS, DS, GS, FS y SS y se establece el ESP y EBP del kernel. Para este ejercicio, se modificaron los archivos gdt.c, gdt.h, defines.h y kernel.asm

1.2. Excepciones

En esta parte del TP, se nos pidió completar la IDT con las excepciones del procesador, dados que las excepciones las definió Intel en 1978, no hay mucho que comentar. Para habilitarlas, agregamos en kernel.asm agregamos un idt [IDT_DESC] y para probarla se hizo una división por 0. (Líneas 108-110 del kernel.asm, se encuentra comentada)

1.3. Video y MMU

1.4. Interrupciones

En esta parte, se completaron las interrupciones del reloj, la del teclado y la 0x66 que posteriormente será usada en el juego dado que la interrupción del reloj la genera la CPU cada x tiempo, recién en este momento se habilitan las interrupciones mediante la instrucción STI. El código de las interrupciones se encuentra en isr.asm

2. El retorno de la GDT y la aparición de la TSS

3. Conclusiones y trabajo futuro