

## 1. Teoría

**Explique los dos espacios existentes para el manejo y diseño de drivers. Para cada función de un driver especifique cómo se realiza en cada espacio**

El espacio del kernel es un espacio de memoria protegido que se utiliza para la ejecución de código del sistema operativo. Los drivers que se ejecutan en este espacio tienen acceso directo al hardware y pueden realizar operaciones de entrada/salida. El espacio del usuario es un espacio de memoria no protegido que se utiliza para la ejecución de aplicaciones de usuario. Los drivers que se ejecutan en este espacio no tienen acceso directo al hardware y deben comunicarse con los drivers del kernel para realizar operaciones de entrada/salida.

### **Espacio del kernel**

- **Inicialización del dispositivo:** El controlador inicializa el dispositivo, lo que incluye configurar sus registros y habilitar sus componentes.
- **Interacción con el hardware:** El controlador interactúa con el hardware del dispositivo para realizar operaciones como leer y escribir datos, controlar el estado del dispositivo y notificar eventos al sistema operativo.
- **Soporte para E/S:** El controlador proporciona soporte para E/S al sistema operativo, lo que incluye permitir que las aplicaciones accedan a los dispositivos.
- **Gestión de errores:** El controlador maneja los errores que se producen en el dispositivo.

### **Espacio del usuario**

- **Abrir y cerrar el dispositivo:** El controlador abre y cierra el dispositivo para que las aplicaciones puedan acceder a él y utiliza las funciones de la API del kernel para abrir y cerrar el dispositivo.
- **Leer y escribir datos:** la lectura y la escritura de datos se realizan en ambos espacios, el espacio del kernel y el espacio del usuario usando la API del kernel para ello.
- **Controlar el estado del dispositivo:** El controlador controla el estado del dispositivo, como su encendido/apagado y su velocidad.

```
• juan@juan-VirtualBox:~/Desktop/Embebidos/Taller8$ echo -n fghi >/dev/memory
• juan@juan-VirtualBox:~/Desktop/Embebidos/Taller8$ cat /dev/memory
• fghi juan@juan-VirtualBox:~/Desktop/Embebidos/Taller8$ echo -n abcd >/dev/memory
• juan@juan-VirtualBox:~/Desktop/Embebidos/Taller8$ cat /dev/memory
○ abcd juan@juan-VirtualBox:~/Desktop/Embebidos/Taller8$
```

Figura 1: Resultados de modificación a *device memory*

## 2. Práctica

Modifique el driver anterior para que soporte escrituras y lecturas al dispositivo virtual de un ancho de 32 bits.

La fig 1 muestra el resultado de la modificación hecha al driver en donde es capaz de escribir-leer datos de 32 bits, la prueba se hace escribiendo-leyendo cuatro caracteres al *device memory*.