

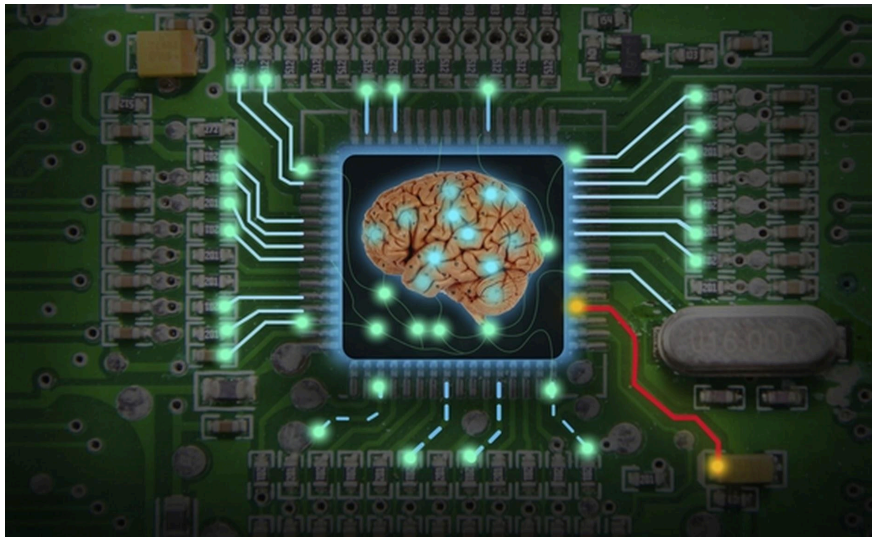


**UNIVERSIDAD AUTÓNOMA DE
CHIHUAHUA
FACULTAD DE INGENIERÍA
INGENIERÍA EN CIENCIAS DE LA
COMPUTACIÓN
GRUPO 5HW1**



SISTEMAS OPERATIVOS

Sistemas Operativos I - Proyecto II



Alumnos:

Irlanda Andrea Burciaga Herrera – 358353

Danna Maribel Corral Salcedo – 358147

Adrian Alejandro González Domínguez - 359834

Luis Dahrey Moncayo Beltran - 363641

Asesor: Iván Miguel Chavero Jurado

Fecha de entrega: 5 de junio de 2024

Funcionamiento del código:

Los módulos del kernel son fragmentos de código que se pueden cargar y descargar en el kernel según sea necesario. Extienden la funcionalidad del kernel sin la necesidad de reiniciar el sistema. Un módulo puede configurarse como integrado o cargable. Para cargar o eliminar un módulo dinámicamente, debe estar configurado como un módulo cargable en la configuración del kernel

Inclusión de Encabezados

El módulo incluye los encabezados necesarios que proporcionan las funciones y definiciones para crear módulos del kernel, manipular dispositivos de carácter y gestionar la comunicación entre el espacio de usuario y el espacio del kernel.

Información del Módulo

Se define la licencia GPL, los autores, una breve descripción y la versión del módulo. Esto es importante para proporcionar información básica sobre el módulo y asegurar la compatibilidad con las políticas del kernel de Linux.

Definiciones y Declaraciones

Se define el nombre del dispositivo como "metallicamodule" y se declara una variable para almacenar el número mayor asignado al dispositivo. También se declara una estructura `cdev` que representa el dispositivo de carácter.

Prototipos de Funciones

Se declaran las funciones que manejarán las operaciones del dispositivo: abrir, cerrar, leer y escribir. Estas funciones son:

- `metallica_open`: Se ejecuta cuando se abre el dispositivo.
- `metallica_release`: Se ejecuta cuando se cierra el dispositivo.
- `metallica_read`: Se ejecuta cuando se lee desde el dispositivo.
- `metallica_write`: Se ejecuta cuando se escribe en el dispositivo.

Se define una estructura que asocia las funciones del módulo con las operaciones estándar del archivo. Esta estructura es utilizada por el kernel para dirigir las llamadas del sistema a las funciones correctas del módulo.

Mensaje de Usuario

Se inicializa un mensaje con una ruta a un archivo MP3. Este mensaje se utilizará para almacenar y mostrar datos pasados desde el espacio de usuario.

Inicialización del Módulo

La función de inicialización se ejecuta cuando el módulo se carga en el kernel:

1. Se registra el dispositivo de carácter con el kernel y se obtiene un número mayor asignado automáticamente.
2. Si el registro falla, se imprime un mensaje de alerta y se retorna un código de error.
3. Si tiene éxito, se imprimen mensajes informativos sobre cómo interactuar con el dispositivo, como crear un archivo de dispositivo con ``mknod``.

Limpieza del Módulo

La función de limpieza se ejecuta cuando el módulo se descarga del kernel:

1. Se desregistra el dispositivo de carácter, liberando los recursos asignados.
2. Se imprime un mensaje indicando que el módulo se está descargando.

Función de Apertura

La función ``metallica_open`` se llama cuando se abre el dispositivo:

1. Se imprime un mensaje indicando que el dispositivo ha sido abierto correctamente.
2. Se retorna 0 para indicar éxito.

Función de Escritura

La función ``metallica_write`` se llama cuando se escribe en el dispositivo:

1. Se declara un buffer para almacenar los datos recibidos.
2. Se copian los datos desde el espacio de usuario al buffer en el espacio del kernel.
3. Se actualiza el mensaje de usuario con los datos recibidos.
4. Se imprime el mensaje recibido desde el espacio de usuario.
5. Se retorna la longitud de los datos escritos o un error si la copia falla.

Función de Cierre

La función ``metallica_release`` se llama cuando se cierra el dispositivo:

1. Se imprime un mensaje indicando que el dispositivo ha sido cerrado correctamente.
2. Se retorna 0 para indicar éxito.

Función de Lectura

La función ``metallica_read`` se llama cuando se lee desde el dispositivo:

1. Se calcula la longitud del mensaje almacenado.
2. Si el desplazamiento es mayor o igual a la longitud del mensaje, se retorna 0 indicando que no hay más datos por leer.
3. Se ajusta la longitud de los datos a leer según el tamaño restante del mensaje.
4. Se copian los datos desde el buffer en el espacio del kernel al espacio de usuario.
5. Si la copia es exitosa, se actualiza el desplazamiento de lectura y se retorna la longitud de los datos leídos.
6. Si la copia falla, se retorna un error.

Registro de las Funciones de Inicialización y Limpieza

Se utilizan macros para registrar las funciones de inicialización y limpieza del módulo, de modo que el kernel las llame en los momentos apropiados, cuando el módulo se carga y se descarga, respectivamente.

Conclusión

Este módulo del kernel crea un dispositivo de carácter que permite interactuar con él a través de operaciones de lectura y escritura. Maneja la comunicación entre el espacio de usuario y el espacio del kernel, proporcionando una interfaz para pasar datos desde el usuario al kernel y viceversa.