

Plataformas de Tiempo Real

PRÁCTICAS: Dispositivos y drivers

Por: J. Javier Gutiérrez gutierjj@unican.es
<http://www.ctr.unican.es/>

Grupo de Computadores y Tiempo Real, Universidad de Cantabria

Práctica 1: Gestión de drivers de dispositivos - cola circular



Objetivo:

- Practicar la instalación, desinstalación y uso de drivers de dispositivos en Linux

Descripción:

- Modificar el ejemplo de buffer virtual visto en clase para que se comporte como una **cola circular** de caracteres en que se puede escribir hasta que se llene y se puede leer hasta que se vacíe.
- Se pueden realizar operaciones de lectura y escritura con tamaños arbitrarios.
- Realizar las pruebas de instalación y desinstalación del driver comprobando los resultados.

Práctica 1: Gestión de drivers de dispositivos - cola circular (cont.)



- Realizar dos programas de prueba:
 - uno para enviar al driver los datos introducidos por el usuario (en un lazo hasta que decida terminar)
 - el otro para obtenerlos y pintarlos en pantalla pidiendo al usuario la cantidad de caracteres que quiere leer.
- Ejecutar los programas desde dos terminales diferentes

Práctica 2: Control de la concurrencia en los drivers



Objetivo:

- Experimentar con la concurrencia en los drivers mediante el control del acceso y mediante el uso de mecanismos de sincronización en los datos.

Descripción:

- Modificar el driver de la práctica 1 para controlar la concurrencia de dos modos:
 - controlando el acceso en el **open** no permitiendo la apertura por más de un proceso
 - controlando las estructuras de datos internas con mecanismos de acceso a las secciones críticas (la cola será un recurso compartido)

Práctica 2: Control de la concurrencia en los drivers (cont.)



- Realizar dos programas de prueba:
 - uno que escriba periódicamente los datos pasados por el usuario al inicio
 - otro que lea datos periódicamente y los muestre en pantalla
- Realizar pruebas con varios procesos leyendo y varios escribiendo (desde diferentes terminales):
 - probar que el driver con control de la concurrencia no falla
 - verificar que los programas de prueba fallan con el driver de la práctica 1 (forzar si es necesario artificialmente las condiciones de fallo)

Práctica 3: Programación del puerto serie



Objetivo:

- Experimentar con la programación del puerto serie y el uso de interrupciones

Descripción:

- Instalar el driver del puerto serie visto en clase en dos computadores que estarán conectados por el puerto serie
- Realizar programas de prueba en los que se pueda comprobar:
 - el correcto funcionamiento de la transmisión y recepción de los mensajes
 - la comunicación a diferentes velocidades



Práctica 4: Programación de funciones de control de dispositivos



Objetivo:

- Experimentar con la programación de funciones de control a través del punto de entrada `ioctl` sobre el puerto serie

Descripción:

- Añadir al driver del puerto serie el punto de entrada `ioctl` con un comando en el que se permita cambiar la velocidad de transmisión del puerto serie
 - utilizar un identificador para el comando y pasarle la velocidad como argumento
- Hacer un programa que permita cambiar la velocidad de transmisión y usar los programas de prueba de la práctica 3 para verificar que todo sigue funcionando después del cambio

Práctica 4: Programación de funciones de control de dispositivos (cont.)



- Recomendación: probar cambios importantes en las velocidades para que puedan ser apreciables

Parte opcional:

- Utilizar las líneas de protocolo (DTR y RI) para detener cualquier envío por parte del driver ante una petición de cambio de velocidad en el equipo que se produce el cambio
 - programar la activación de nuevo con un cierto retraso (ej., 2 seg.)
 - activar DTR para informar al otro equipo del cambio de velocidad
 - enviar un byte con el código de la nueva velocidad
- En el equipo que recibe el cambio a través de RI
 - detener la transmisión y esperar la recepción del código con la nueva velocidad, y programar el cambio
 - esperar un tiempo a activar de nuevo la transmisión

Práctica 5: Programación de dispositivos PCI



Objetivo:

- Experimentar con la programación de dispositivos conectados al bus PCI, con interfaces de entradas y salidas analógicas y digitales y con la operación *ioctl*

Descripción:

- Realizar un sistema de muestreo y reconstrucción de una señal analógica utilizando la tarjeta PCI-9111
- Escribir un driver sencillo:
 - capaz de digitalizar el valor de la señal de entrada como un comando de la operación *ioctl*
 - capaz de establecer el valor de la señal analógica de salida también como un comando de la operación *ioctl*

Práctica 5: Programación de dispositivos PCI (cont.)



- Escribir un programa de aplicación que a petición del usuario:
 - tome un número determinado de muestras de una señal analógica
 - reproduzca la señal almacenada que previamente se ha digitalizado