

Se dispone de un sistema multiprocesador con 2 procesadores y con sistema operativo de propósito general. En un momento determinado existen los siguientes procesos planificables, además del proceso nulo:

- Proceso A:
 - Lee de una tubería `pipe1` vacía [Actividad A1]
 - Posteriormente abre un fichero `file1`, lo que implica una operación en disco [Actividad A2]
 - Realiza una escritura de los datos leídos de la tubería en el fichero `file1`, lo que implica de nuevo una operación en disco [Actividad A3]
 - Finalmente invoca la llamada `exit()` [Actividad A4]
- Proceso B:
 - Se trata de un proceso nuevo, que debe ponerse en marcha [B1]
 - Una vez puesto en marcha, lee de una tubería `pipe2` vacía [B2]
 - Abre el fichero `file1`, pero no necesita llevar a cabo ninguna operación en disco [B3]
 - Lee bloque de fichero (requiere operación en disco) sobre variable global sin valor inicial, variable que nunca ha sido accedida y sólo ocupa una página. En el sistema hay marcos de páginas libres. El acceso a esa página implica un fallo que no requiere operación del disco [B4]
 - El proceso continúa en modo usuario [B5]
 - Posteriormente le llega una señal enviada por el proceso D, lo que provoca que muera [B6]
- Proceso C:
 - Escribe en la tubería `pipe1` [C1]
 - Escribe en la tubería `pipe2` [C2]
 - Posteriormente sigue en modo usuario [C3]
 - Finalmente el usuario aborta el proceso tecleando `Ctrl-C` [C4]
- Proceso D:
 - En modo usuario, causa un fallo de página, lo que requiere en primer lugar escribir la página modificada a disco [D1]
 - Y a continuación leer de disco la nueva página [D2]
 - Posteriormente el proceso invoca la llamada `kill()` dirigida hacia el proceso B [D3]

Se usa planificación por prioridades, ignorando la afinidad al procesador. La prioridad de los procesos cumple la siguiente relación: $P(A) > P(B) > P(C) > P(D)$.

Tanto las interrupciones de disco como las de teclado usan interrupción SW de sistema.

Responder razonadamente a las siguientes cuestiones:

1. Teniendo en cuenta que se trata de núcleos expulsivos, realizar una traza temporal de ambos procesadores. En la traza especificar los 2 procesos en ejecución, el modo en que están dichos procesos (usuario o sistema), todos los tratamientos de eventos que ocurren durante la ejecución, la activación de interrupciones SW que sean necesarias, las IPI que sean necesarias, los cambios de contexto voluntarios (CCV) e involuntarios (CCI) que se producen y la actividad que se encuentra realizando en ese instante (AX, BX, ...). Para realizar la traza, considerar que la interrupción de disco se trata por parte del procesador que tenga el proceso de menor prioridad entre los que estén ejecutando y que la interrupción llega cuando ha habido dos cambios de modo en dicho procesador o ninguno si está ejecutando el proceso nulo.
2. ¿Para qué utilizan las interrupciones SW de sistema las interrupciones de disco y de teclado?

- Explicar con un ejemplo el uso de las mismas.
3. En el escenario descrito, especificar en qué situación o situaciones se utilizan interrupciones SW de proceso y de qué tipo son (expulsivas o no expulsivas).
 4. En el escenario descrito, realizar un estudio de los posibles conflictos de sincronización específicos que se podrían dar, explicando cuáles serían las soluciones que debería proporcionar el sistema operativo de propósito general para resolver dichos conflictos.

SOLUCIÓN.-

1. Véase figura 1.
2. Las interrupciones SW de sistema se utilizan para diferir la ejecución de operaciones no urgentes del tratamiento de interrupciones. En el escenario descrito, en el caso de las interrupciones de disco y teclado existirán determinadas actividades no urgentes que pueden diferirse. Por ejemplo, en el caso del teclado, una actividad urgente es obtener el código de la tecla que originó el evento (para que se pueda atender una interrupción del mismo tipo lo antes posible) y una actividad no urgente que se puede diferir mediante el uso de una interrupción SW de sistema es la interpretación de dicho código.
3. Las interrupciones SW de proceso se pueden utilizar para la implementación de cambios de contexto involuntarios (planificación), para la terminación involuntaria de un proceso y para la implementación de operaciones asíncronas. En el escenario descrito, sólo las 2 primeras situaciones aparecen. Dado que se trata de un núcleo expulsivo, el primer tipo de interrupción SW de proceso es expulsiva. El segundo tipo de interrupción SW (abortar un proceso) es no expulsiva.
4. Al tratarse de un sistema multiprocesador, tenemos 3 tipos de problemas de sincronización: producidos por tratamiento de eventos asíncronos, debido a cambios de contexto voluntarios e involuntarios y debido al paralelismo real de un multiprocesador. Trataremos el último caso por ser el más crítico e incluir toda la problemática de los otros 2 tipos de problemas. La técnica que se utiliza en este tipo de sistemas es la de spinlocks. Para la sincronización entre llamada/rutina de interrupción y otra rutina de interrupción se usan spinlocks y la rutina interrumpida inhibe localmente la interrupción. Para la sincronización entre llamadas concurrentes, uso de spinlocks más la interrupción de planificación inhibida (se trata de un núcleo expulsivo). Si la sección crítica es larga o implica bloqueo se utilizan semáforos, asegurando la atomicidad de dichos semáforos mediante el uso de spinlocks.

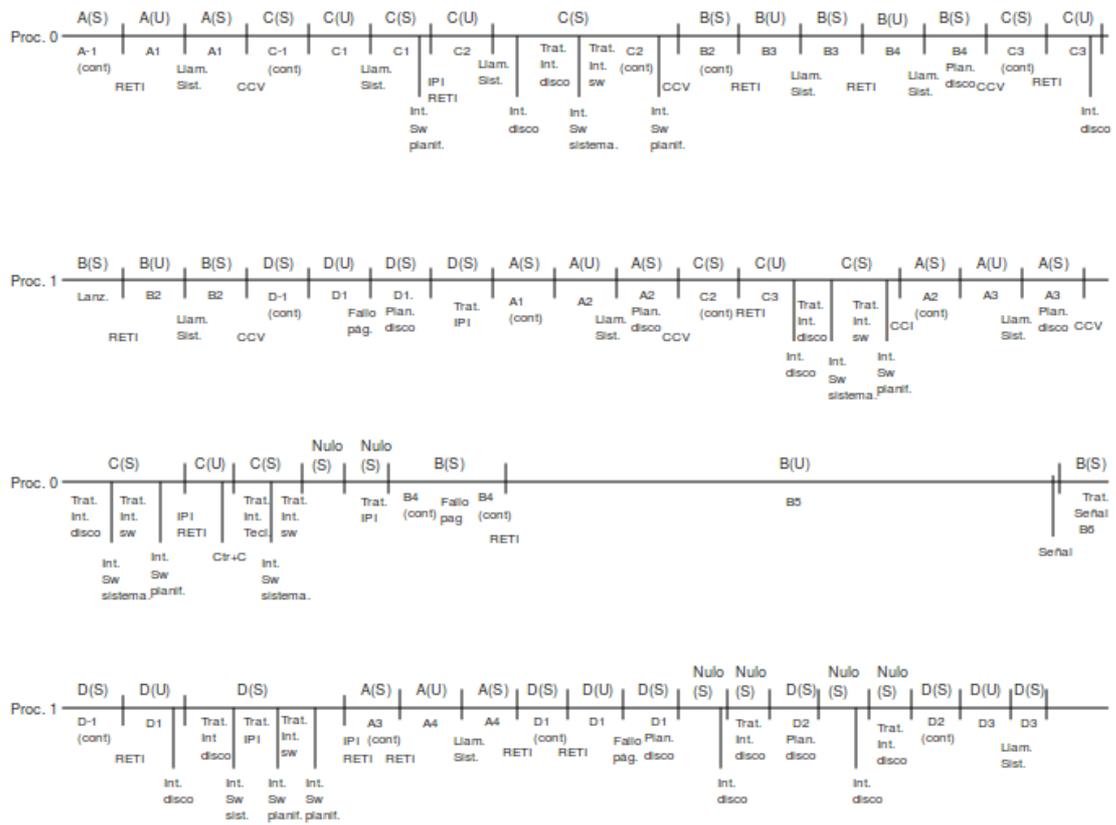


Figura 1: Traza temporal de ejecución