Capítulo 9 Señales y Procesamiento de Señales

9.1 Señales e Interrupciones

En el capítulo 8 se vio que las interrupciones son peticiones enviadas a la CPU, lo cual hace que la CPU pase de sus ejecuciones normales a hacer procesamiento de interrupción. Como las interrupciones a una CPU, las señales son peticiones enviadas a un proceso, el cual pasa de sus ejecuciones normales a hacer procesamiento de señal. Antes de discutir señales y procesamiento de señal, revisaremos los conceptos y el mecanismo de interrupciones, lo cual nos ayudará a poner las señales en un Sistema Operativo (SO) en una perspectiva apropiada.

(1). Primero, generalizamos la noción de proceso para que signifique: un “proceso” (entre comillas) es una secuencia de actividades. Algunos ejemplos de “procesos” generalizados son

Una persona, que realiza sus tareas de rutina diarias.

Un proceso de Unix (o de MTX), el cual corre en su espacio(s) de direcciones.

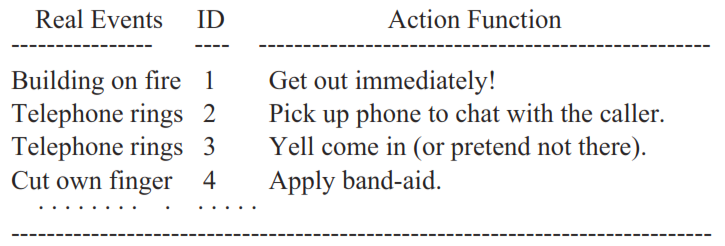
Una CPU, la cual ejecuta instrucciones de máquina.

(2). Una “interrupción” es un evento enviado a un “proceso”, lo cual hace que un proceso pase de hacer sus actividades normales a hacer algo más, llamado “procesamiento de interrupción”. El proceso podría retomar sus actividades normales cuando termine el procesamiento de la interrupción.

(3). El término “interrupción” puede ser aplicado a cualquier “proceso”, no solo a una CPU en una computadora. Por ejemplo, podríamos hablar de los siguientes tipos de “interrupciones”.

(3).1. Interrupciones de PERSONA:

Mientras estoy leyendo, calificando, soñando sespierto, etc. en mi oficina, algunos eventos reales podrían ocurrir, tal como



Todas estas podrían ser llamadas interrupciones de PERSONA dado que todas ellas sacan a una persona de sus actividades normales para “procesar o manejar la interrupción”. Después de procesar una interrupción, una persona podría retomar cualquier cosa que el/ella estaba haciendo antes (si la persona sobrevive y aun recuerda lo que el/ella estaba haciendo antes). A cada interrupción se le asigna un número de identificación único, y tiene una función de acción per-instalada, la cual una persona puede “ejecutar” una vez que reciba una interrupción. Dependiendo de su origen, las interrupciones podrían ser clasificadas en 3 categorias:

De hardware: incendio en el edificio, alarma de reloj, etc.

De otra persona: llamada telefónica, alguien toca la puerta, etc.

Causada por simismo: cortarse un dedo, comer demasiado, etc.

Dependiendo de su urgencia, las interrupciones pueden ser clasificadas como

No enmascarables (NMI): incendio en el edificio.

Enmascarables: Alguien toca la puerta, etc.

Cada una de las funciones de acción de una PERSONA es instalada por sus instintos o por experiencia. Es imposible completar la tabla de arriba dado que hay demasiados tipos diferentes de interrupciones de PERSONA, pero la idea debe ser clara.

(3). 2. Interrupciones de PROCESO:

Estas son interrupciones enviadas a un proceso. Mientras un proceso se está ejecutando, podría recibir interrupciones de tres fuentes diferentes:

Desde el hardware: Combinación de teclas Control\_C desde terminal, temporizador de intervalo, etc.

Desde otro proceso: kill(pid,SIG#), death\_of\_child, etc.

Causada por simismo: división entre cero, dirección no válida, etc.

Cada interrupción de proceso es convertida a un número ID único, el cual es enviado al proceso. A diferencia de las interrupciones de PERSONA, las cuales son de muchos tipos, siempre podemos limitar el número de interrupciones a un proceso. En Unix, las interrupciones de proceso son llamadas señales, y están numeradas del 1 al 31. Para cada señal, un proceso tiene una función de acción en su estructura PROC, la cual el proceso puede ejecutar cuando recibe una señal. De forma similar a una persona, un proceso podría enmascarar ciertos tipos de señales para posponer su procesamiento. Si es necesario, un proceso también podría cambiar sus funciones de acción de señal.

(3). 3. Interrupciones HARDWARE:

Estas son señales enviadas a un procesador o CPU. Estas también tienen origen en tres posibles fuentes:

Desde el hardware: Timer, dispositivos I/O, etc.

Desde otros procesadores: FFP, DMA, otras CPUs en un sistema multiprocesador.

Generada por simismo: división entre 0, error de protección, instrucción INT.

Cada interrupción tiene un número de vector de interrupción único. La función de acción es un manejador de interrupción en la tabla de vectores de interrupción. La CPU siempre está ejecutando un proceso. La CPU no causa interrupciones por simisma (a menos que esté dañada). Las interrupciones causadas por simismo son debidas al proceso que la está usando, o en la mayoría de los casos, subutilizando la CPU. Lo anterior incluye la instrucción INT n o instrucciones equivalentes, la cual causa que la CPU se cambie del modo Umode al modo Kmode. Lo último incluye todos los errores trap reconocidos por la CPU como excepciones. Por lo tanto, podríamos excluir las interrupciones autoprovocadas de una CPU, dejando solamente aquellas externas a una CPU.

(3). 4. Errores de proceso Trap:

Un proceso podría causarse interrupciones a simismo. Tales interrupciones son debidas a errores, p.e. dividir entre 0, dirección no válida, instrucción ilegal, violación de privilegio, etc. las cuales son reconocidas por la CPU como excepciones.

[pag. 259, 274/551, K.C. Wang D&Impl of the MTX OS]