

# Procesos

Pedro O. Pérez M., PhD.

Diseño de sistemas embebidos avanzados  
Tecnológico de Monterrey

*pperezm@tec.mx*

09-2022

## ① Procesos

¿Qué es un proceso?

Hilos

## ② Administración de procesos

Planificador a corto plazo

Planificador a largo plazo

Planificador a mediano plazo

## ③ Comunicación entre procesos

- Las primeras computadoras permitían que solo se ejecutara un programa a la vez. Este programa tenía el control completo del sistema y tenía acceso a todos los recursos del sistema. Por el contrario, los sistemas informáticos contemporáneos permiten que se carguen múltiples programas en la memoria y se ejecuten al mismo tiempo. Esta evolución requirió un control más firme y una mayor compartimentación de los distintos programas; y estas necesidades dieron lugar a la noción de **proceso**, que es un programa en ejecución. Un proceso es la unidad de trabajo en un sistema moderno de tiempo compartido.

- Un sistema informático actual consta de una colección de procesos: procesos del sistema operativo que ejecutan el código del sistema y procesos del usuario que ejecutan el código del usuario. Potencialmente, todos estos procesos pueden ejecutarse al mismo tiempo, con el CPU (o CPUs) multiplexados entre ellos. Al cambiar el CPU entre procesos, el sistema operativo puede hacer que la computadora sea más productiva.

Mencionamos antes que un proceso es un programa en ejecución. Un proceso es más que el código del programa, que a veces se conoce como **sección de texto**. También incluye la actividad actual, representada por el valor del **contador del programa** y el contenido de los registros del procesador. Un proceso generalmente también incluye la **pila de procesos**, que contiene datos temporales (como parámetros de función, direcciones de retorno y variables locales), y **una sección de datos**, que contiene variables globales. Un proceso también puede incluir un **heap**, que es memoria que es asignados dinámicamente durante el tiempo de ejecución del proceso.

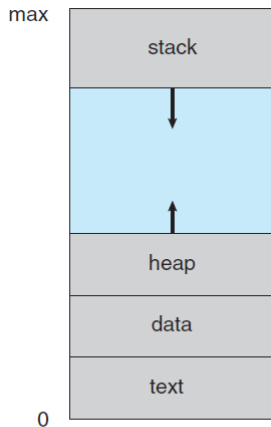
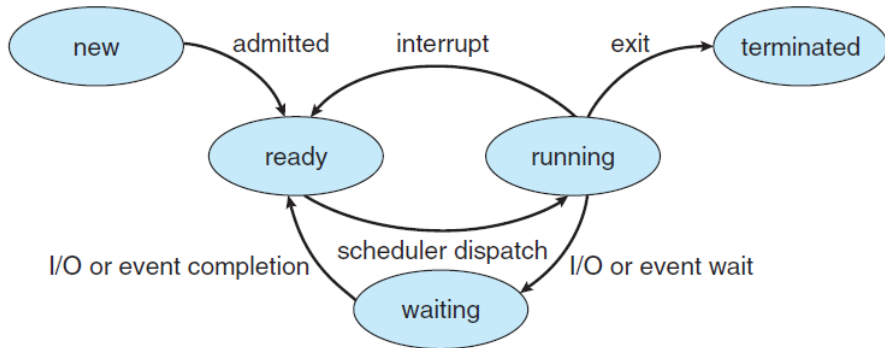


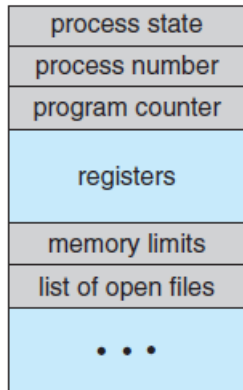
Figure 3.1 Process in memory.



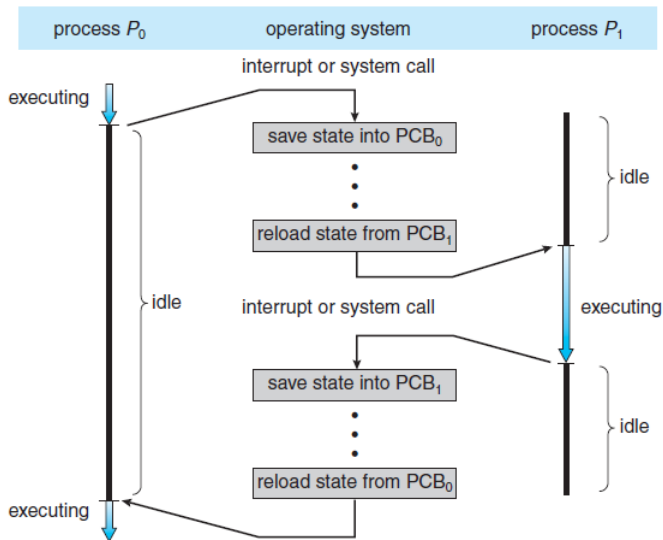
**Figure 3.2** Diagram of process state.

# Process Control Block (PCB)

- Estado del proceso. El estado puede ser nuevo, listo, en ejecución, en espera, detenido, etc.
- Contador de programa.
- Registros de CPU.
- Información de programación de CPU.
- Información de gestión de la memoria.



**Figure 3.3** Process control block (PCB).



**Figure 3.4** Diagram showing CPU switch from process to process.



- Un proceso es un programa que realiza un solo hilo de ejecución. Por ejemplo, cuando un proceso está ejecutando un programa de procesador de texto, se está ejecutando un solo hilo de instrucciones. Este único hilo de control permite que el proceso realice solo una tarea a la vez. El usuario no puede escribir caracteres simultáneamente y ejecutar el corrector ortográfico dentro del mismo proceso, por ejemplo. La mayoría de los sistemas operativos modernos han ampliado el concepto de proceso para permitir que un proceso tenga múltiples subprocesos de ejecución y, por lo tanto, realice más de una tarea a la vez. Esta característica es especialmente beneficiosa en sistemas multinúcleo, donde varios subprocesos pueden ejecutarse en paralelo. En un sistema que admite subprocesos, la PCB se amplía para incluir información para cada subproceso. También se necesitan otros cambios en todo el sistema para admitir subprocesos.

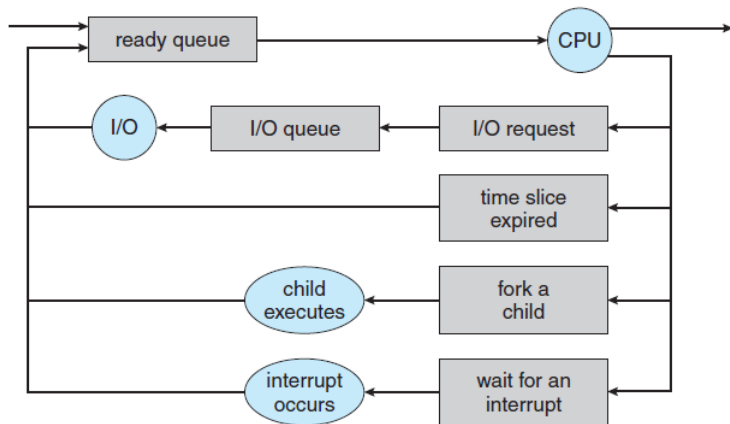


Figure 3.6 Queueing-diagram representation of process scheduling.

- Un proceso migra entre las distintas colas de programación a lo largo de su vida. El sistema operativo debe seleccionar, para fines de programación, procesos de estas colas de alguna manera. El proceso de selección lo lleva a cabo el planificador correspondiente. El programador a largo plazo, o programador de trabajos, selecciona procesos de este grupo y los carga en la memoria para su ejecución. El programador a corto plazo, o programador del CPU, selecciona entre los procesos que están listos para ejecutarse y asigna el CPU a uno de ellos.

- La principal distinción entre estos dos programadores radica en la frecuencia de ejecución. El planificador a corto plazo debe seleccionar un nuevo proceso para el CPU con frecuencia. Un proceso puede ejecutarse durante unos pocos milisegundos antes de esperar una solicitud de E/S. A menudo, el programador a corto plazo se ejecuta al menos una vez cada 100 milisegundos. Debido al poco tiempo entre ejecuciones, el planificador a corto plazo debe ser rápido. Si toma 10 milisegundos decidir ejecutar un proceso durante 100 milisegundos, entonces  $10 / (100 + 10) = 9$  por ciento de la CPU se está usando (desperdiciado) simplemente para programar el trabajo.

- El planificador a largo plazo se ejecuta con mucha menos frecuencia; minutos pueden separar la creación de un nuevo proceso y el siguiente. El planificador a largo plazo controla el grado de multiprogramación (el número de procesos en la memoria). Si el grado de multiprogramación es estable, entonces la tasa promedio de creación de procesos debe ser igual a la tasa de salida promedio de los procesos que abandonan el sistema.

- Algunos sistemas operativos, como los sistemas de tiempo compartido, pueden introducir un nivel intermedio adicional de programación. La idea clave detrás de un programador a mediano plazo es que a veces puede ser ventajoso eliminar un proceso de la memoria (y de la contención activa para el CPU) y así reducir el grado de multiprogramación. Posteriormente, el proceso puede reintroducirse en la memoria y su ejecución puede continuar donde se detuvo. Este esquema se llama intercambio. El proceso es reemplazado y luego reemplazado por el programador de mediano plazo. El intercambio puede ser necesario para mejorar la combinación de procesos o porque un cambio en los requisitos de memoria ha comprometido en exceso la memoria disponible, lo que requiere que se libere memoria.

- Los procesos que se ejecutan simultáneamente en el sistema operativo pueden ser procesos independientes o procesos cooperantes. Un proceso es independiente si no puede afectar o ser afectado por los otros procesos que se ejecutan en el sistema. Cualquier proceso que no comparta datos con ningún otro proceso es independiente. Un proceso está cooperando si puede afectar o verse afectado por los otros procesos que se ejecutan en el sistema. Claramente, cualquier proceso que comparte datos con otros procesos es un proceso cooperativo.

- Hay varias razones para proporcionar un entorno que permita la cooperación de procesos:
  - **El intercambio de información.** Dado que varios usuarios pueden estar interesados en la misma información (por ejemplo, un archivo compartido), debemos proporcionar un entorno que permita el acceso simultáneo a dicha información.
  - **Aceleración de la computación.** Si queremos que una tarea en particular se ejecute más rápido, debemos dividirla en subtareas, cada una de las cuales se ejecutará en paralelo con las demás. Aceleración de la computación. Ten en cuenta que tal aceleración solo se puede lograr si la computadora tiene múltiples núcleos de procesamiento.
  - **Modularidad.** Es posible que deseemos construir el sistema de manera modular, dividiendo las funciones del sistema en procesos o subprocesos separados.
  - **Conveniencia.** Incluso un usuario individual puede trabajar en muchas tareas al mismo tiempo. Por ejemplo, un usuario puede estar editando, escuchando música y compilando en paralelo.



- Los procesos de cooperación requieren un mecanismo de comunicación entre procesos (IPC) que les permita intercambiar datos e información. Hay dos modelos fundamentales de comunicación entre procesos: memoria compartida y transmisión de mensajes. En el modelo de memoria compartida, se establece una región de memoria compartida por procesos cooperantes. Luego, los procesos pueden intercambiar información leyendo y escribiendo datos en la región compartida. En el modelo de transmisión de mensajes, la comunicación se realiza mediante mensajes intercambiados entre los procesos cooperantes.

En grupos de 2 integrantes,

- Investiguen cuál son los mecanismos de comunicación entre procesos que tiene el sistema operativo Raspbian.
- Generen un reporte con los resultados, suban el reporte a Canvas.