

UNIVERSIDAD CARLOS III DE MADRID

# Guía Presentación Bloque 4 (Temas 7, 8 y 9)

---

Departamento de Ingeniería de Sistemas y  
Automática

RAÚL PÉRULA MARTÍNEZ  
LUIS ENRIQUE MORENO LORENTE  
ALBERTO BRUNETE GONZALEZ  
CESAR AUGUSTO ARISMENDI GUTIERREZ  
DOMINGO MIGUEL GUINEA GARCIA ALEGRE  
JOSÉ CARLOS CASTILLO MONTOYA



Universidad  
Carlos III de Madrid



Esta obra se publica bajo una licencia Creative Commons Reconocimiento-NoComercial-CompartidaIgual 3.0  
España.



## Guía de presentación del bloque IV (Tema 7. Introducción a la ejecución multiciclo; tema 8. Introducción a la planificación dinámica de instrucciones: Scoreboard; y tema 9. Planificación dinámica de instrucciones II: Algoritmo de Tomasulo)

En este bloque de contenidos se abordan algoritmos de planificación dinámica de instrucciones, implementados en procesadores reales. Estos métodos permiten reducir el CPI aportando ventajas como la ejecución de instrucciones fuera de orden, sistemas de buffers virtuales para mejorar la gestión de los recursos hardware y renombramiento de registros y adelantamiento para reducir los problemas causados por dependencias de datos.

Para permitir las mejoras anteriores es necesario resolver un problema existente en las arquitecturas tradicionales. Este se debe al tiempo necesario para ejecutar instrucciones complejas (como divisiones y multiplicaciones en punto flotante o modos de direccionamiento complejos), comparado con el tiempo que requieren operaciones sencillas. Esto provoca que el tiempo necesario para ejecutar las primeras tenga un impacto en las últimas, por lo que la utilización de un único cauce es inviable. Por este motivo, se propone la utilización de cauces dinámicos con etapas que se ejecuten opcionalmente según el tipo de instrucción. Esta solución requiere que la CPU disponga de hardware específico (unidades funcionales) para algunas de estas operaciones de forma que mientras que se ejecuta una instrucción en una se pueda proceder a ejecutar otras instrucciones en otras unidades funcionales, de esta forma se solapan las ejecuciones de estas instrucciones.

La planificación dinámica es especialmente útil en el caso de que existan dependencias desconocidas durante la compilación (p. ej. debidas a referencias a memoria). El uso de estas técnicas simplifica el diseño de los compiladores y permite que el código que ha sido compilado para un cierto procesador pueda ejecutarse en otro procesador segmentado diferente. En este sentido, el **tema 8** introduce la **arquitectura scoreboard**, la cual implementa algunas de las mejoras mencionadas anteriormente como la ejecución fuera de orden si hay recursos suficientes y no existen dependencias. Además, si hay recursos hardware disponible intenta mantener un CPI de 1, realizando la detección de riesgos para controlar la ejecución de las instrucciones entre etapas.

Finalmente, el **tema 9** aporta una primera toma de contacto con otra técnica de planificación dinámica muy conocida, el **algoritmo de Tomasulo**. Esta técnica permite, al igual que la anterior, ejecución fuera de orden. Una de sus grandes ventajas es la resolución de riesgos de datos mediante renombramiento de registros, por lo que la tasa de instrucciones por ciclo aumenta considerablemente. Además, cada unidad funcional lleva



Universidad Carlos III de Madrid  
Departamento de Ingeniería de Sistemas y Automática

asociada una cola virtual para disminuir el impacto de las dependencias hardware. El control de estas colas se realiza de manera distribuida, utilizando un bus de datos común para facilitar el adelantamiento.