



Procesadores Segmentados y Superescalares

Motivación:

- ✓ ¿Qué implican la segmentación y el paralelismo en el procesador?
- ✓ ¿Qué mejoras de rendimiento son alcanzables con ellas?
- ✓ ¿Qué problemas surgen con la segmentación y como se enfrentan?
- ✓ ¿Qué técnicas derivan de la segmentación?
- ✓ ¿Cómo se organiza un procesador superescalar?



Procesadores Segmentados y Superescalares

- Segmentación y Paralelismo
- Segmentación de un Procesador
- Parámetros fundamentales de rendimiento con segmentación
- Tipos de Segmentación
- Riesgos de la Segmentación. Rendimiento real obtenido
 - ✓ Riesgos estructurales
 - ✓ Riesgos por dependencias de datos
 - ✓ Riesgos de control
- Segmentación avanzada. Procesadores Superescalares

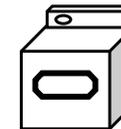
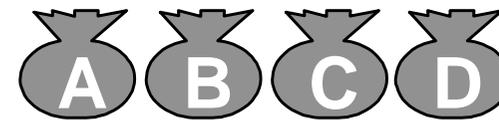


Segmentación (*Pipelining*)

- Ejemplo: hacer la colada

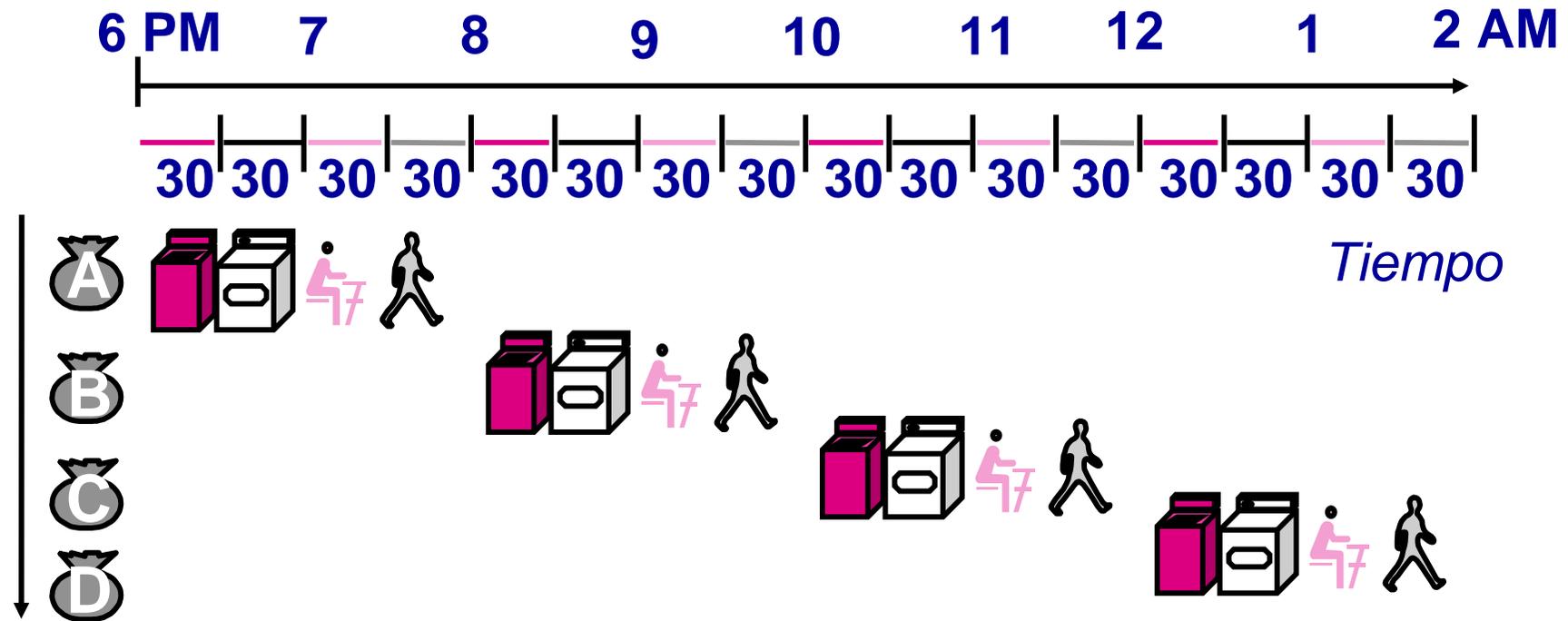
(cuatro cargas de ropa para lavar, secar, planchar y colocar)

- ✓ Lavar lleva 30 minutos
- ✓ Secar lleva 30 minutos
- ✓ Planchar lleva 30 minutos
- ✓ Colocar lleva 30 minutos





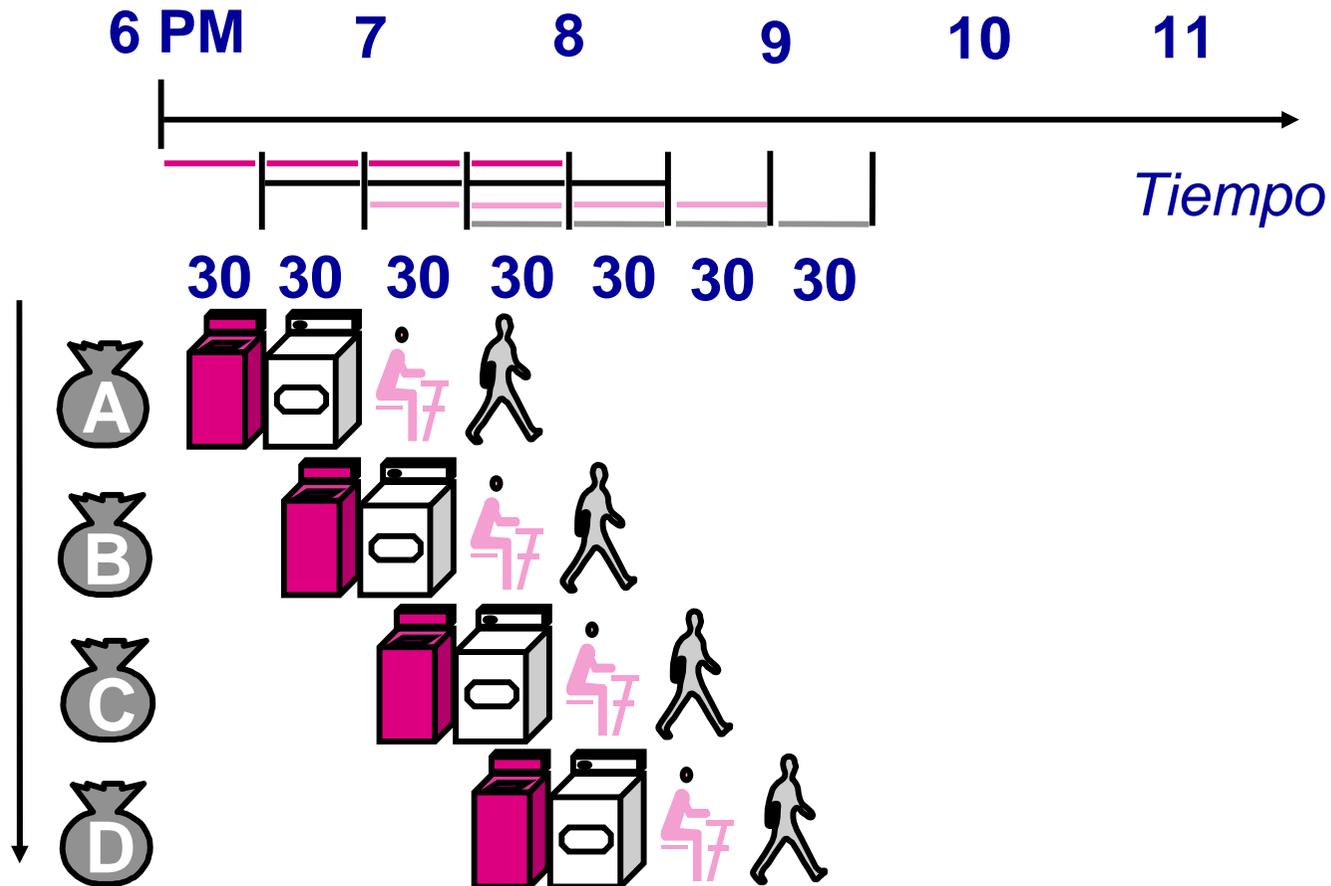
Colada secuencial



- ✓ La colada secuencial para cuatro cargas lleva 8 horas
- ✓ Si aplicásemos la técnica de segmentación ¿cuánto llevaría?

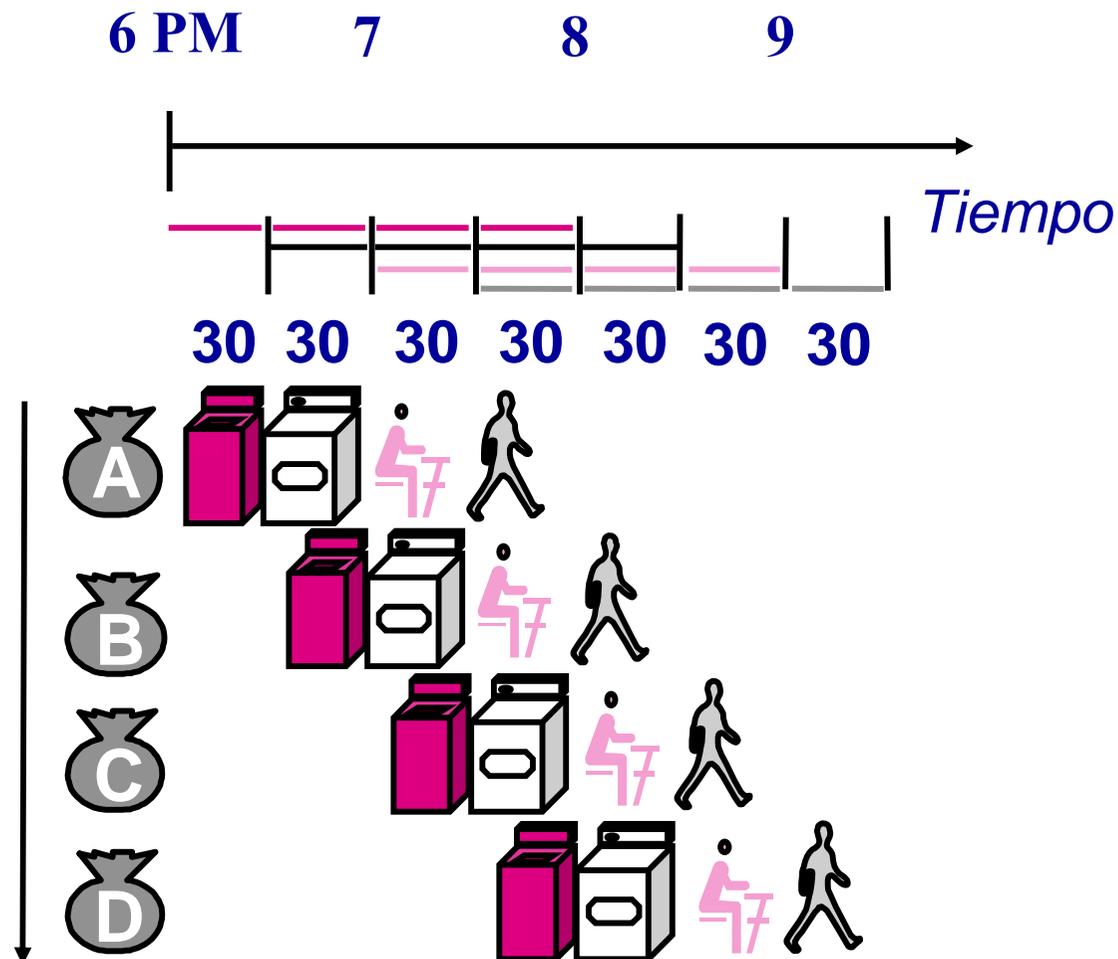


Colada con Segmentación



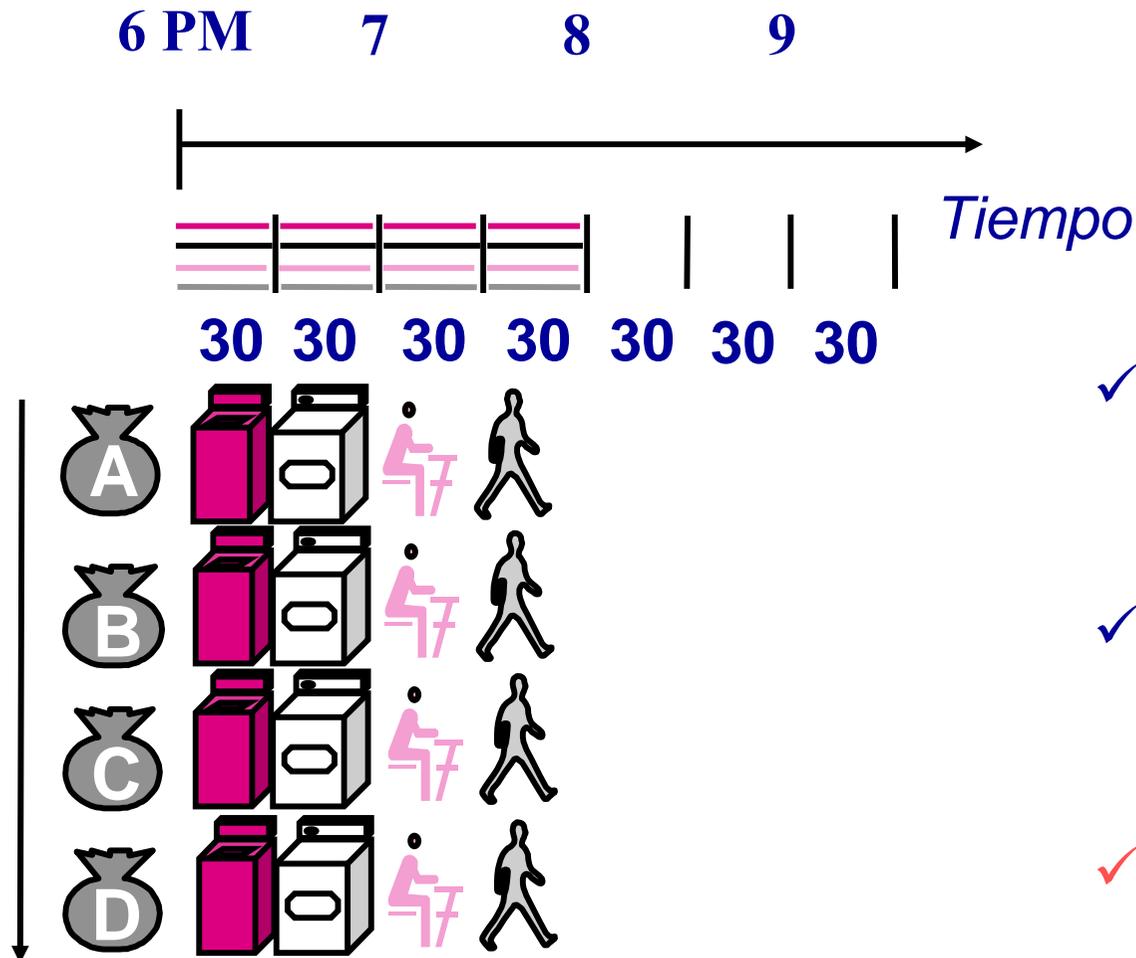
✓ La colada con segmentación para cuatro cargas lleva 3,5 horas

Segmentación: conclusiones



- ✓ La segmentación no reduce la latencia, sino que aumenta la productividad
- ✓ Las múltiples tareas concurrentes utilizan diferentes recursos
- ✓ **Ganancia potencial**
=

Paralelismo

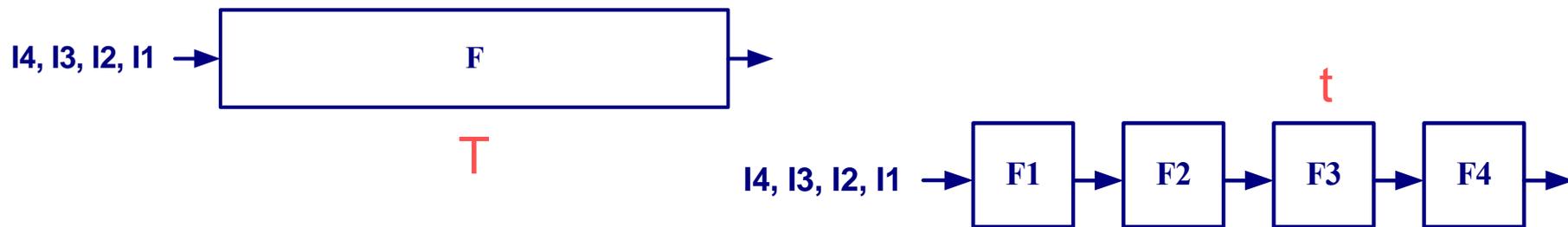


- ✓ El paralelismo no reduce la latencia, sino que aumenta la productividad
- ✓ Las múltiples tareas concurrentes utilizan diferentes recursos
- ✓ **Ganancia potencial**
=

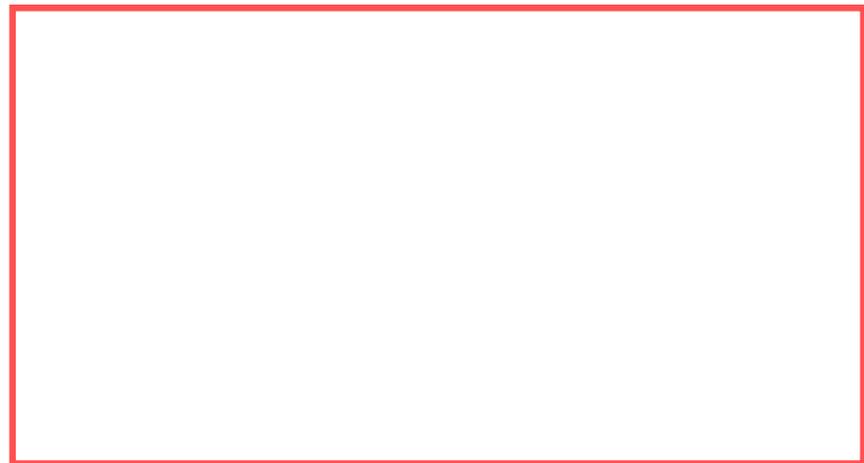


Segmentación de un Procesador

- Cauce de ejecución de instrucciones segmentado



Ejemplo: F1: BI, F2: DEC, F3: BO, F4: EJ



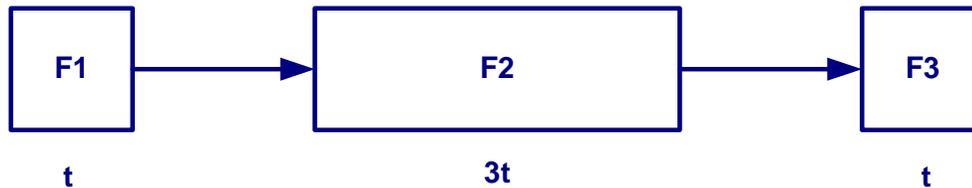


Segmentación de un Procesador

- Nivel al que se explota el paralelismo
 - ✓ Entre instrucciones de un flujo secuencial
 - ✓
- Ventaja importante respecto a otras técnicas (vectorización / multiprocesamiento)
 - ✓ Invisible al programador
- Aspectos importantes
 - ✓ Equilibrado del cauce de ejecución de instrucciones
 - Tiempos de etapas diferentes → etapa cuello de botella →
 - $G =$
 - ✓ Implementación del cauce de ejecución de instrucciones
 - Funcionamiento síncrono

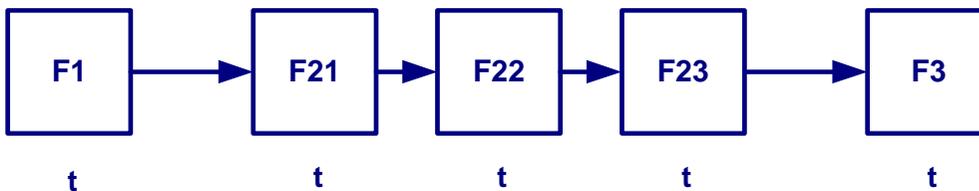


Equilibrado de un Cauce Segmentado



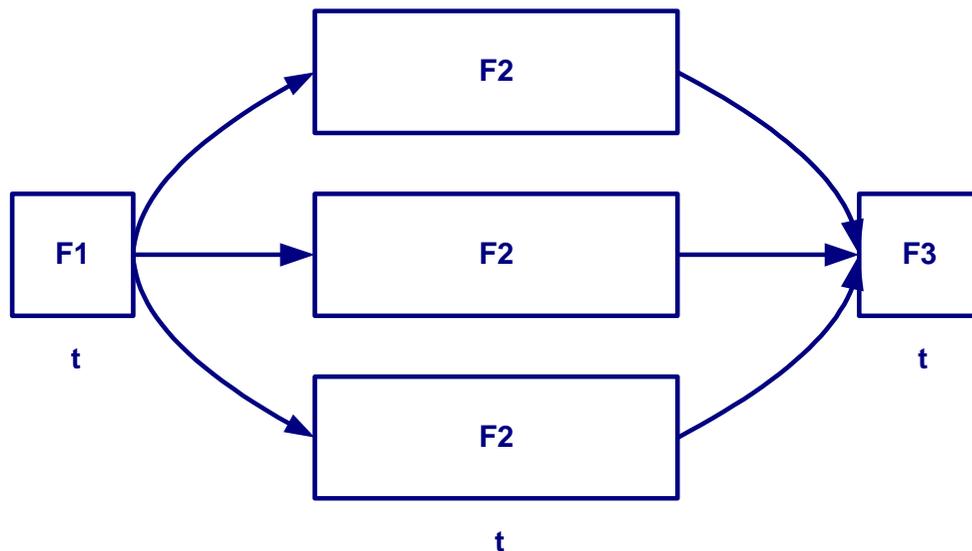
F2 = cuello de botella

G =



Cauce Equilibrado

G =

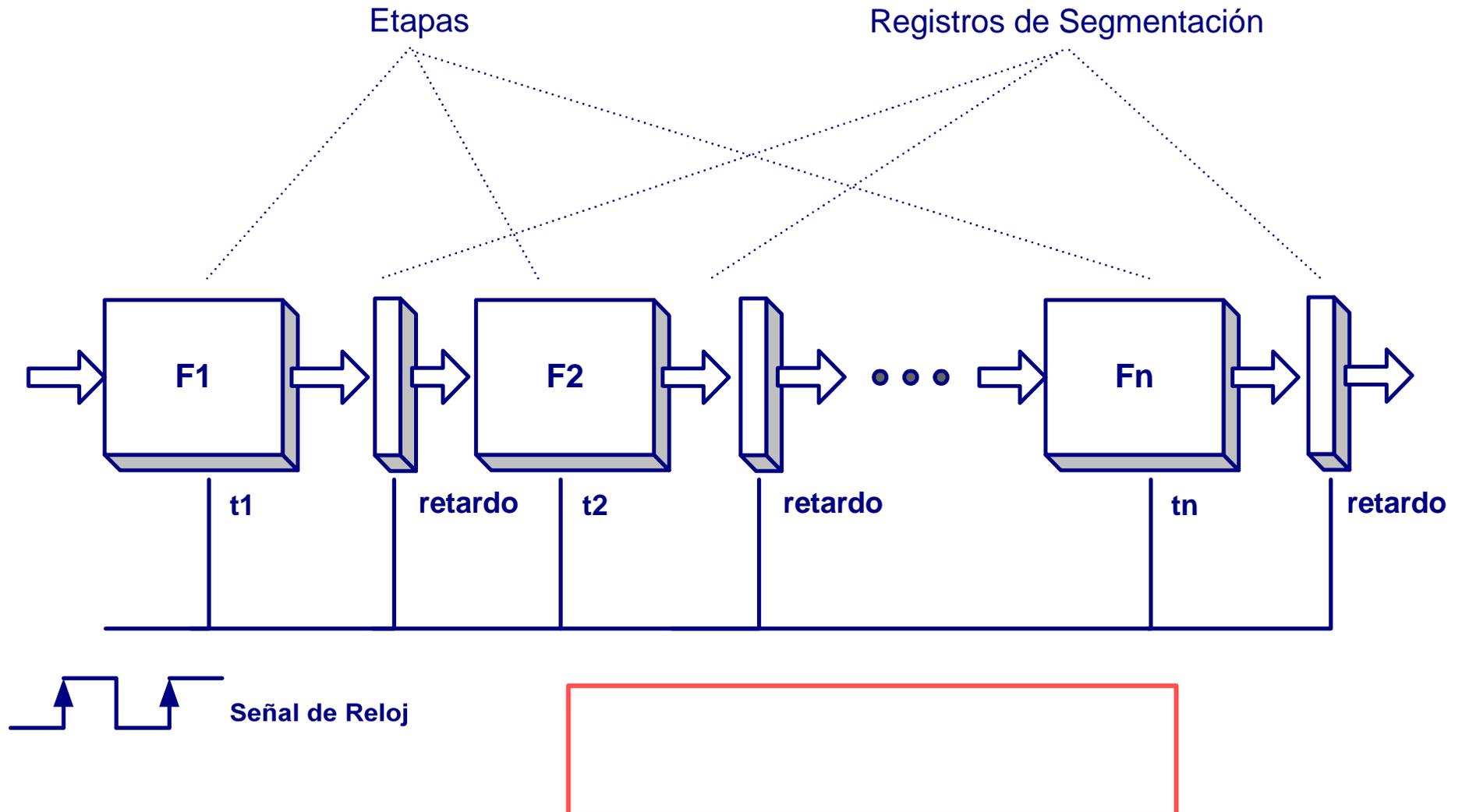


Cauce Equilibrado

G =

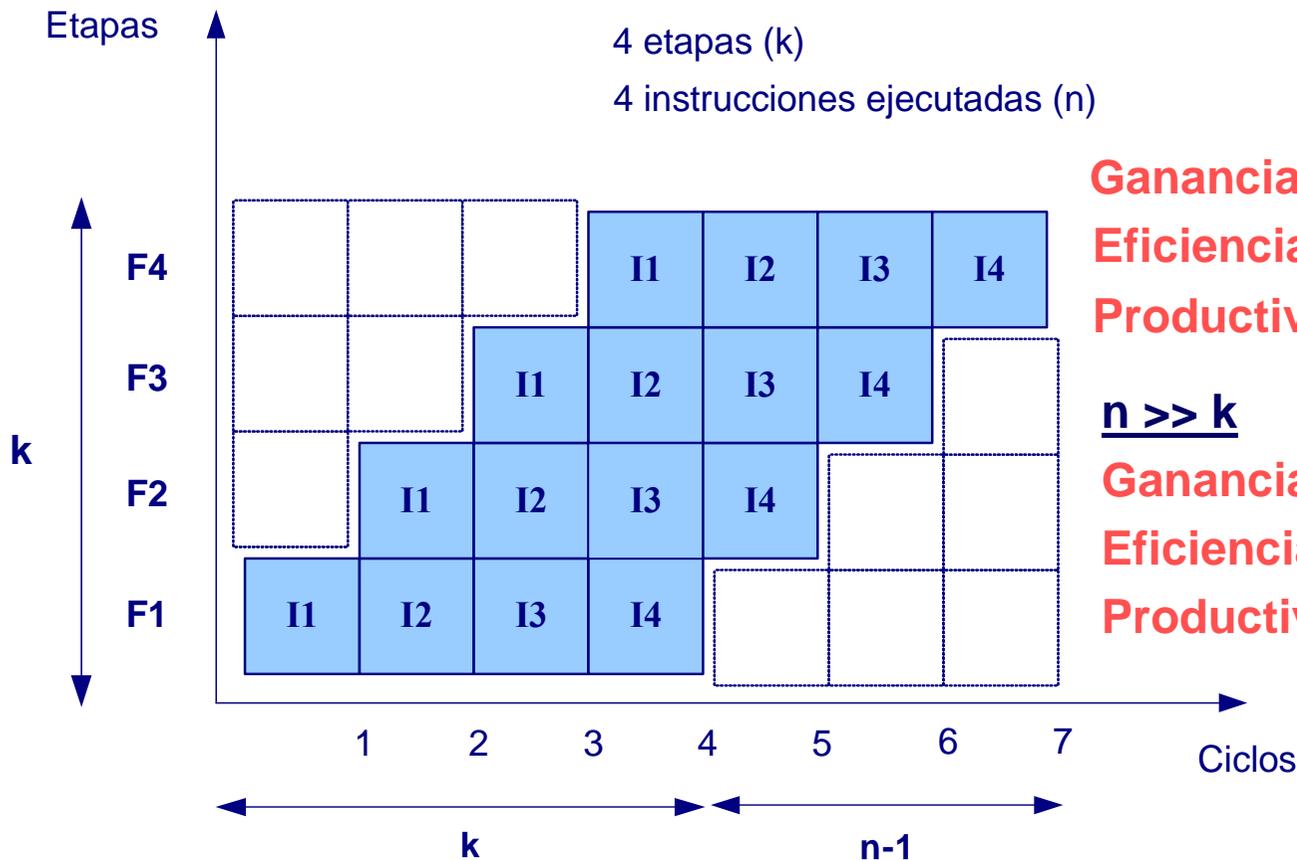
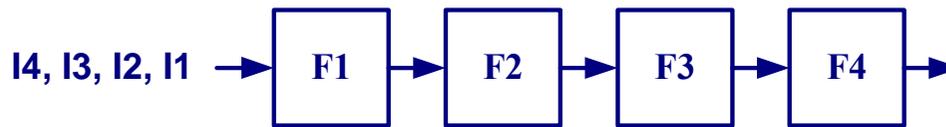


Implementación de un Cauce Segmentado



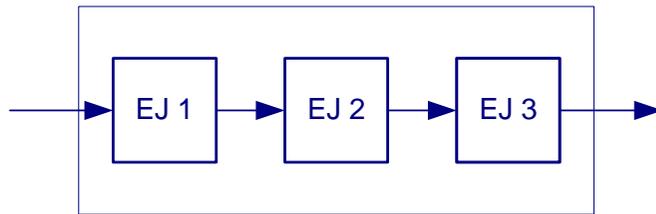


Parámetros fundamentales de Rendimiento con Segmentación

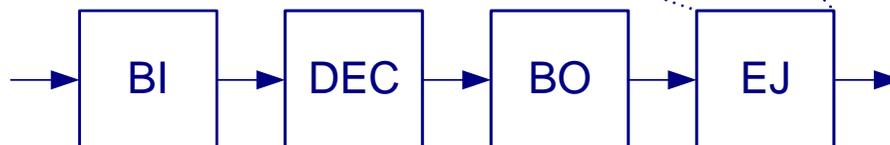




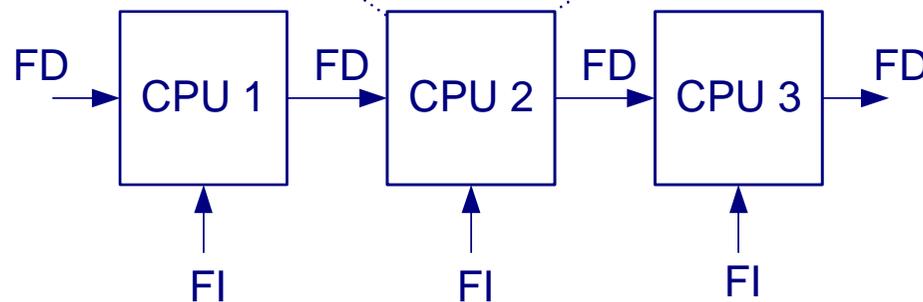
Tipos de Segmentación Encauzada



Encauzamiento Aritmético
(Segmentación de ALU)



Encauzamiento de Instrucciones
(Segmentación de CPU)



Encauzamiento de procesadores
(Arquitectura MISD)

FI =

FD =



Ejemplo de Unidad Aritmética Segmentada

Dato A

Exp	Mantisa
-----	---------

Dato B

Exp	Mantisa
-----	---------

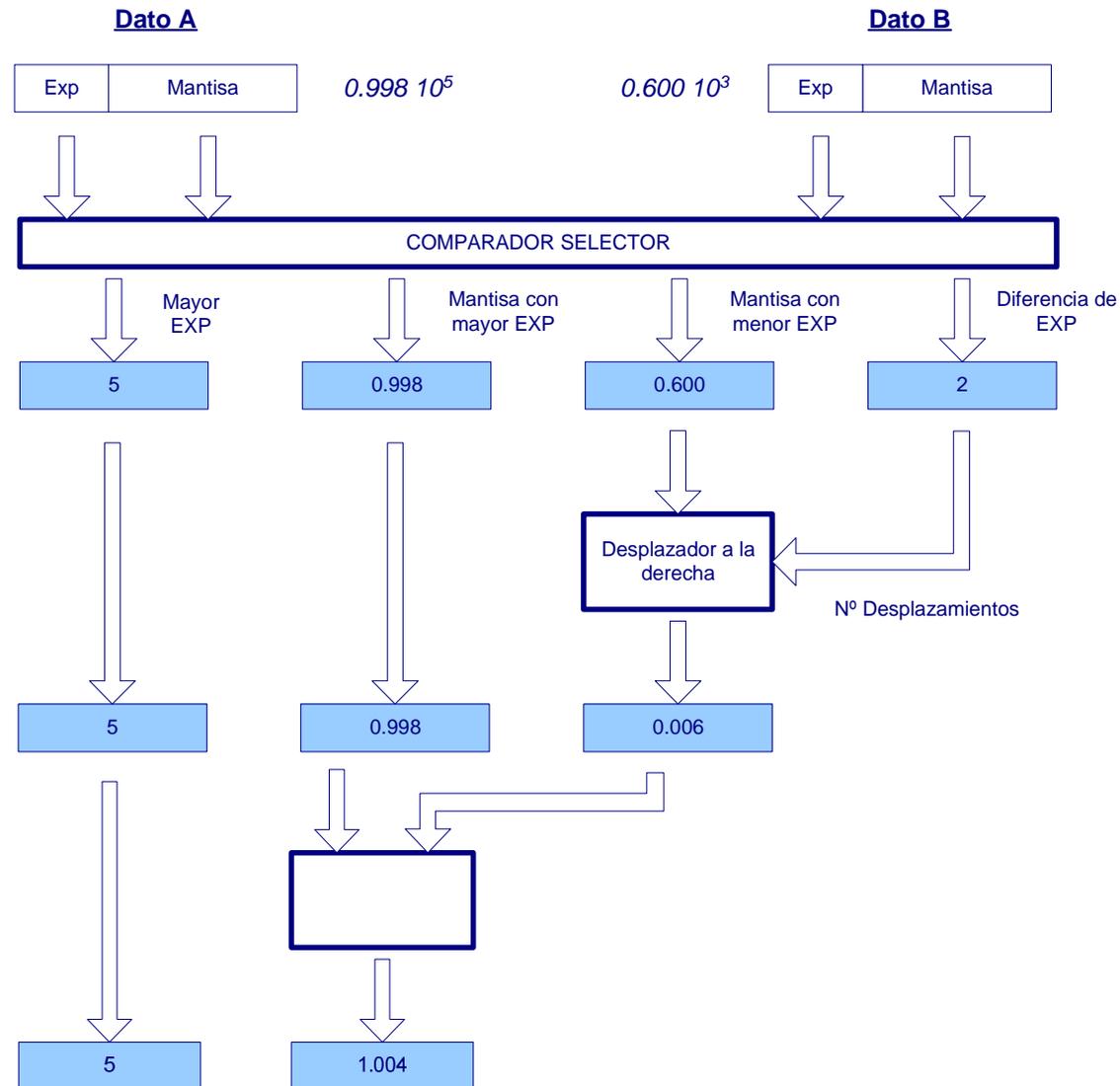
Resultado

Exp	Mantisa
-----	---------

- Suma en punto flotante
 - ✓ Resta de exponentes y selección del dato de menor exponente
 - ✓ Desplazamiento a la derecha de la mantisa con menor exp. hasta igualar exp.
 - ✓ Suma de las mantisas
- Normalización del resultado
 - ✓ Obtención del n° de dígitos significativos a la izquierda del punto decimal
 - ✓ Desplazamiento a la derecha de la mantisa suma (incrementando el exponente) tantos lugares como n° de dígitos significativos



Unidad de Suma en Punto Flotante Segmentada





Unidad de Suma en Punto Flotante Segmentada

