



## Riesgos en la Segmentación

- Tipos de riesgos
- Soluciones
- Rendimiento real teniendo en cuenta los riesgos
- Ejemplo de procesador segmentado sencillo
- Riesgos Estructurales. Causas
- Riesgos por Dependencias de Datos. Tipos y soluciones
- Riesgos de Control. Tipos y soluciones



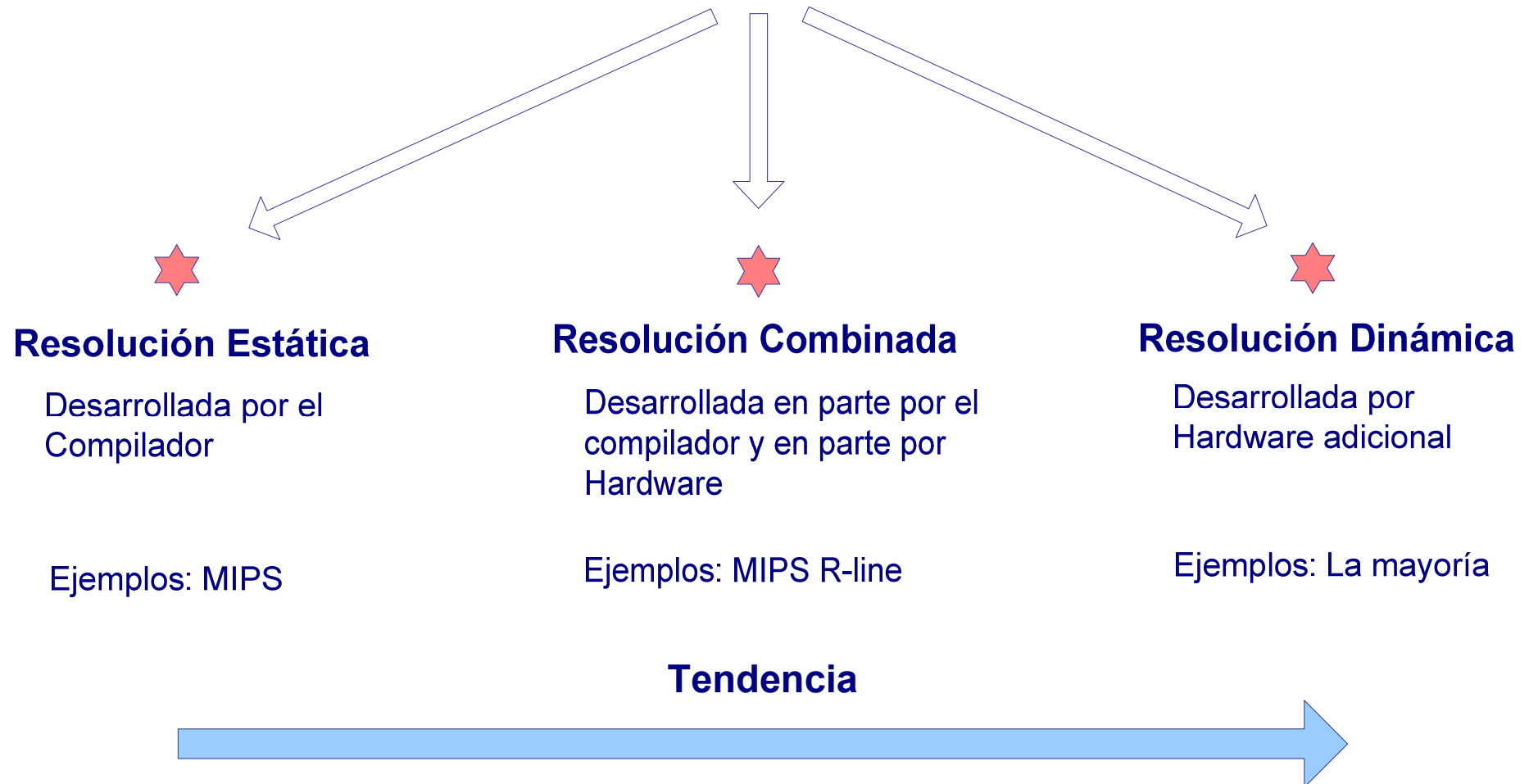
## Tipos de Riesgos

- **Riesgos Estructurales**
  - ✓ Conflictos en el uso de los recursos del cauce segmentado por parte de las múltiples instrucciones cuya ejecución se solapa (recursos insuficientemente replicados o UF no segmentadas)
- **Riesgos por Dependencias de Datos**
  - ✓ Problemas derivados de dependencias entre instrucciones cuya ejecución se solapa
- **Riesgos de Control**
  - ✓ Problemas derivados de las instrucciones de control de flujo, debido al desconocimiento temporal de la siguiente instrucción a ejecutar

Consecuencias a minimizar:



# Soluciones





# Rendimiento real teniendo en cuenta los riesgos

$$G = \text{_____}$$

**Profundidad Segmentación =**

**Ciclos Detención  
por Instrucción = \***



## Ejemplo de procesador segmentado sencillo

- ★ Procesador segmentado lineal de 5 etapas (MIPS R3000)



- ★ Etapas:

BUS	Busqueda de Instrucción
D-L	Decodificación y Lectura de Registros
EJE	Ejecución en la ALU (operación + calculo de direcciones)
MEM	Acceso a Memoria (lectura o escritura)
ESC	Escritura en Registro

- ➔ Todas las instrucciones tardan el mismo tiempo y "atraviesan" todas las etapas (las utilicen o no)
- ➔ Arquitectura de Carga/Almacenamiento (el resto de instrucciones opera sobre registros)



## Ejemplo de procesador segmentado sencillo

Ciclo de Reloj	1	2	3	4	5	6	7	8	9 ...
Instrucción i	BUS	D-L	EJE	MEM	ESC				
Instrucción i+1		BUS	D-L	EJE	MEM	ESC			
Instrucción i+2			BUS	D-L	EJE	MEM	ESC		
Instrucción i+3				BUS	D-L	EJE	MEM	ESC	
Instrucción i+4					BUS	D-L	EJE	MEM	ESC



## Riesgos Estructurales

- Causas:

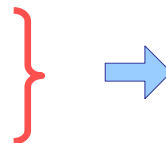
- ✓ **Recursos insuficientes:** *los recursos no se han replicado lo suficiente como para permitir la ejecución solapada de todas las combinaciones de instrucciones sin dar lugar a pérdida de ciclos*
- ✓ **Unidades Funcionales no segmentadas:** *no es posible iniciar una secuencia de instrucciones en la que varias consecutivas utilicen esa unidad funcional sin esperas y por tanto pérdida de ciclos*

## Riesgos por recursos insuficientes

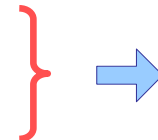
Ciclo de Reloj	1	2	3	4	5	6	7	8	9...
Instrucción i	BUS	D-L	EJE	MEM	ESC				
Instrucción i+1		BUS	D-L	EJE	MEM	ESC			
Instrucción i+2			BUS	D-L	EJE	MEM	ESC		
Instrucción i+3				BUS	D-L	EJE	MEM	ESC	
Instrucción i+4					BUS	D-L	EJE	MEM	ESC

- Requerimientos: En cada ciclo debe efectuarse ...

- ✓ lectura de una instrucción
- ✓ lectura o escritura de un dato



- ✓ lectura del contenido de dos registros
- ✓ escritura de un registro



- ✓ operación de ALU y cálculo del nuevo valor del PC (*en la etapa BUS*)







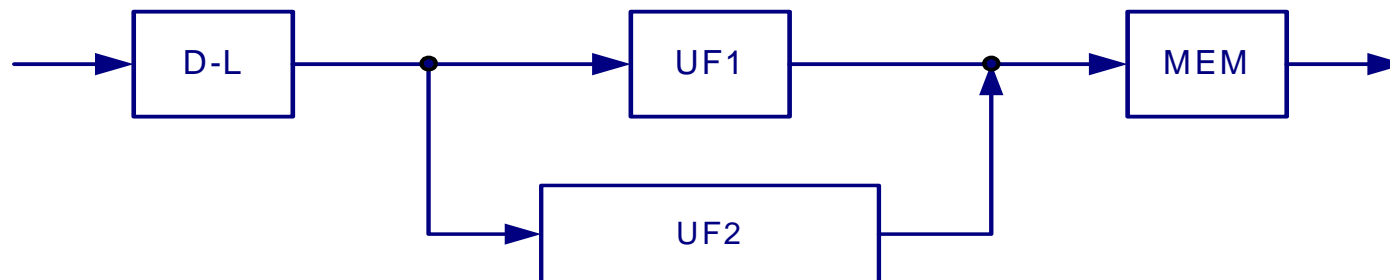
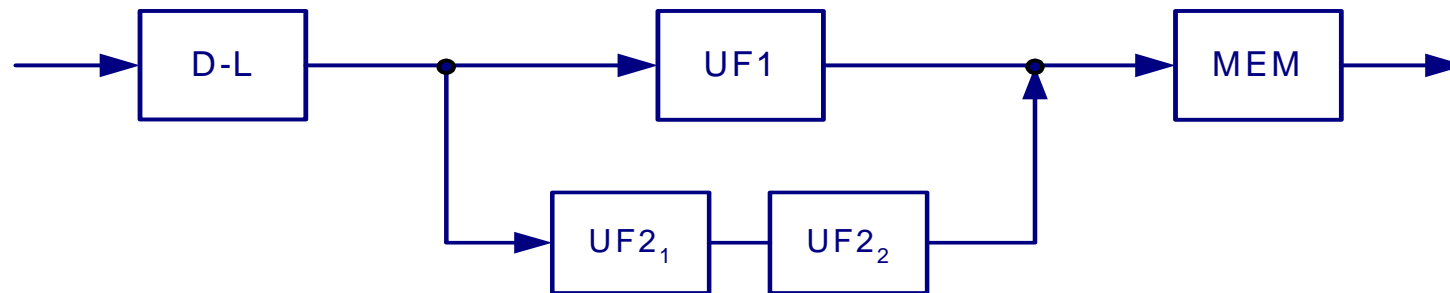
# Riesgos por recursos insuficientes

Ciclo de Reloj	1	2	3	4	5	6	7	8	9...
No Carga/Alm.	BUS	D-L	EJE	MEM	ESC				
Instrucción i+1		BUS	D-L	EJE	MEM	ESC			
Instrucción i+2			BUS	D-L	EJE	MEM	ESC		
Instrucción i+3				BUS	D-L	EJE	MEM	ESC	
Instrucción i+4					BUS	D-L	EJE	MEM	ESC

Ciclo de Reloj	1	2	3	4	5	6	7	8	9...	
Carga/Almac.	BUS	D-L	EJE	MEM	ESC					
Instrucción i+1		BUS	D-L	EJE	MEM	ESC				
Instrucción i+2			BUS	D-L	EJE	MEM	ESC			
Instrucción i+3					BUS	D-L	EJE	MEM	ESC	
Instrucción i+4						BUS	D-L	EJE	MEM	ESC

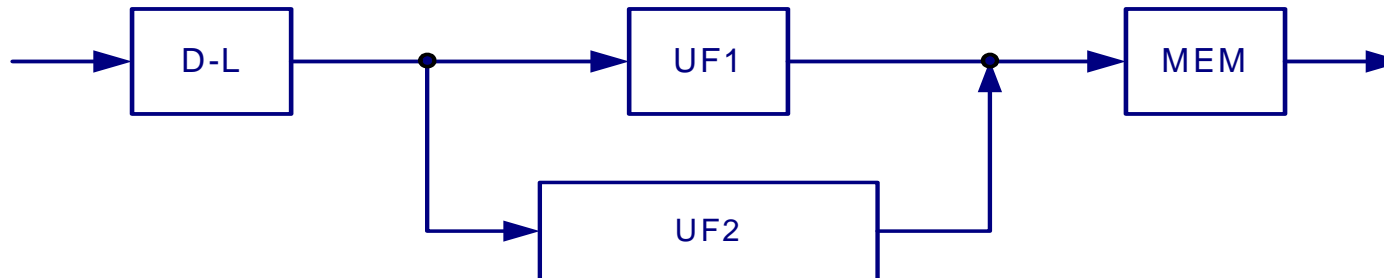


## Riesgos por Unidades Funcionales no segmentadas





## Riesgos por Unidades Funcionales no segmentadas



- ➔ No se pueden soportar 2 instrucciones que necesiten UF2 en 2 ciclos consecutivos
- ➔ La frecuencia de instrucciones que puede soportar la unidad no segmentada está limitada:

$$f_{\max} =$$

$$f_{\max} =$$