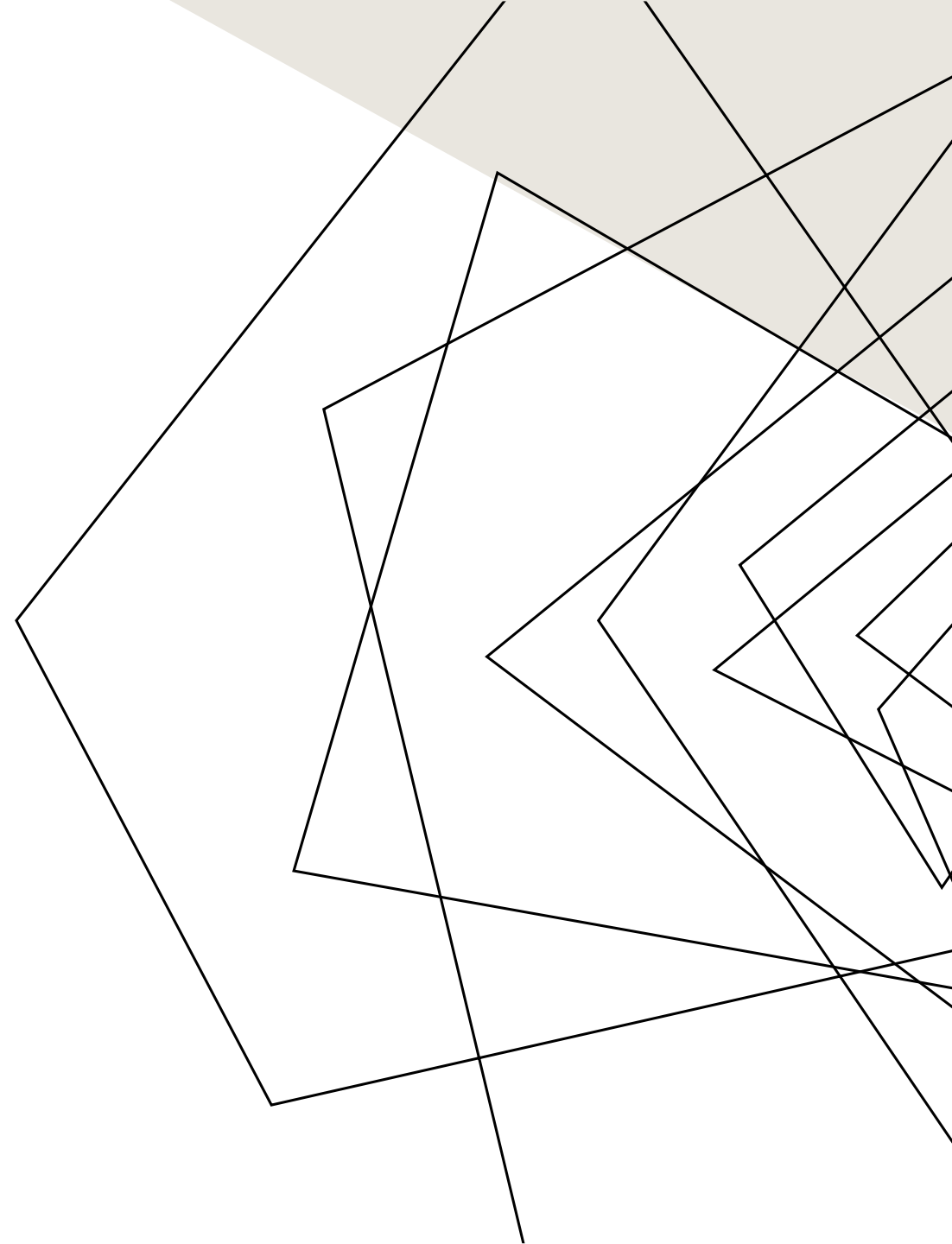


# ARQUITECTURA VLIW

FIGUEROA TORRES OIRAM

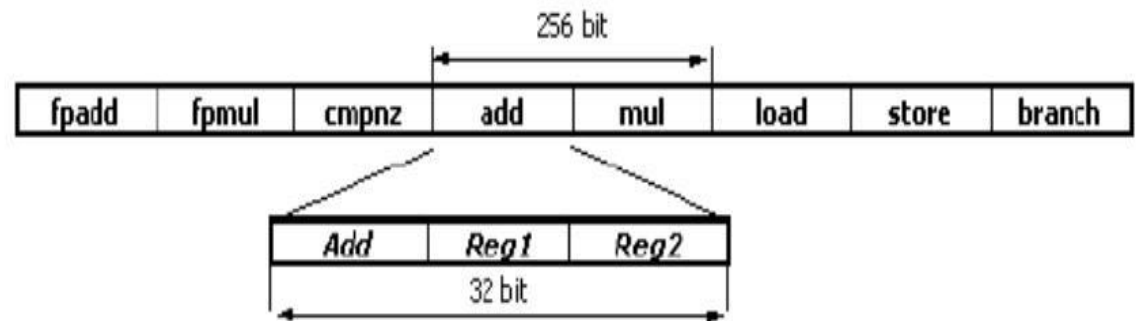
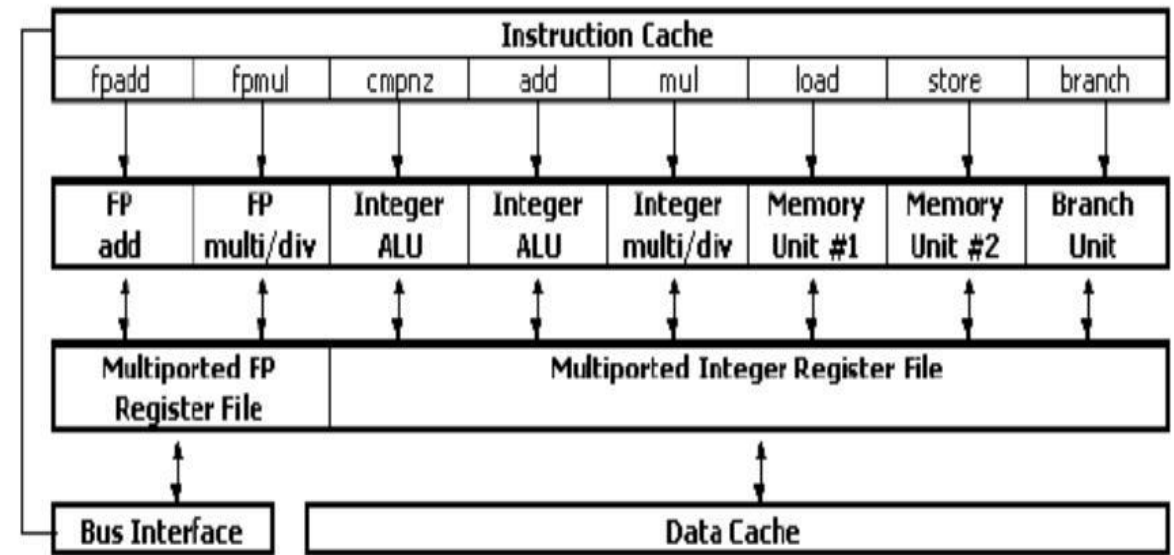
# CONCEPTO

La arquitectura VLIW (Very Long Instruction Word) es una arquitectura de procesador desarrollada por Josh Fisher a principios de 1980 que permite ejecutar múltiples operaciones en paralelo en un solo ciclo, utilizando múltiples unidades funcionales como varias ALU, multiplicadores, entre otras. Esto la hace similar en paralelismo a las arquitecturas superescalares, pero con un enfoque diferente en la planificación y ejecución de instrucciones

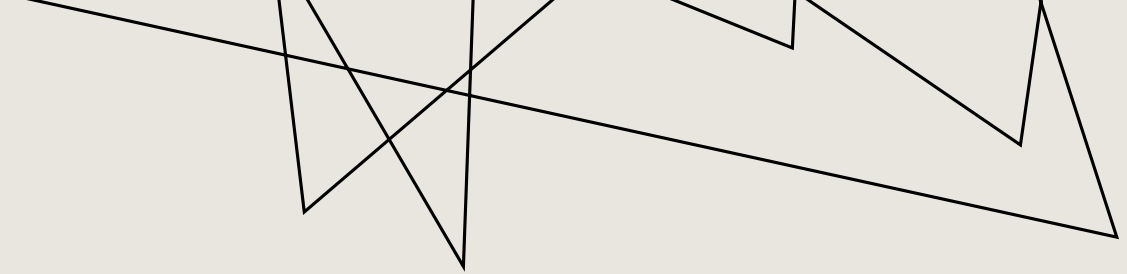


# ALGUNAS CARACTERÍSTICAS

## Generating of VLIW instruction words



A hypothetical VLIW processor architecture

- 
- Las instrucciones en los procesadores VLIW son de gran tamaño porque cada una especifica el estado y la operación de todas las unidades funcionales del sistema, permitiendo ejecutar múltiples operaciones en paralelo y simplificando el diseño del hardware.
  - La planificación del código recae en el compilador, ya que el hardware no se encarga de planificar las instrucciones en tiempo de ejecución, a diferencia de la arquitectura superescalar.

# COMPARACIÓN DE ARQUITECTURAS: CISC, RISC Y VLIW

Característica	CISC	RISC	VLIW
Tamaño de instrucción	Variable	Un tamaño, usualmente 32 bits	Variable o fija, según el esquema de codificación de la ISA
Formato de instrucción	La colocación de campos varia	Colocación de campos regular y consistente	
Semántica de instrucción	Varía de simple a complejo; muchas posibles operaciones dependientes por instrucción	Casi siempre una sola operación simple	Muchas operaciones simples e independientes
Registros	Pocos	Muchos, de propósito general	
Referencias de memoria	Incluidas con operaciones	No incluidas en operaciones; arquitectura de load/store	
Enfoque de diseño de hardware	Aprovecha implementaciones con microcódigo	Aprovecha implementaciones con un pipeline	Aprovecha implementaciones con múltiples pipelines

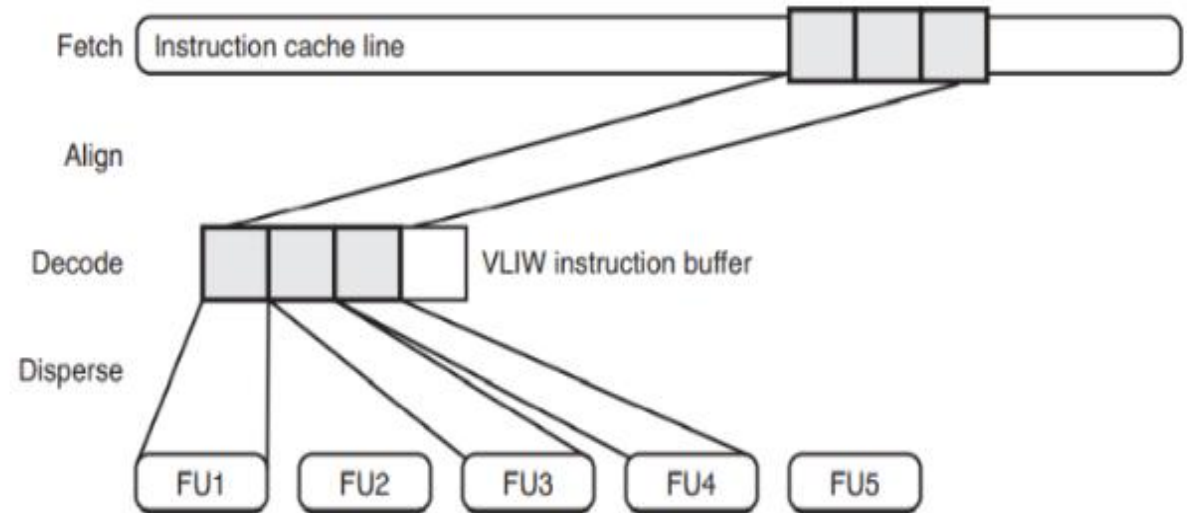
# FUNCIONAMIENTO DE INSTRUCCIONES



# INSTRUCCIÓN FETCH (IF)

Suponiendo un caché de instrucciones estándar, se debe obtener instrucciones del caché de instrucciones en un búfer de instrucciones VLIW, lo que puede incluir la alineación de la instrucción desde su posición de memoria original para que coincida con el diseño del búfer de instrucciones.

Luego, decodifica al menos parcialmente cada operación, lo que proporciona información sobre los requisitos de registro y unidad funcional. Después se dispersan las operaciones a las unidades funcionales apropiadas



# LONGITUD DE INSTRUCCIÓN Y ALINEACIÓN

Una operación VLIW corresponde a una sola operación para una sola unidad funcional; una instrucción VLIW consta de un conjunto de operaciones VLIW. Un paquete es una unidad orientada a la obtención y decodificación, puede incluir instrucciones múltiples, únicas o parciales, y operaciones únicas o múltiples.

El tema fundamental en el paso Align es determinar los límites de instrucción; es decir, dónde comienza y termina cada instrucción: una vez encontrados los límites, la unidad de alineación selecciona una subsecuencia contigua de la línea de caché de instrucción y la coloca en el búfer de instrucción.



# DECODIFICACIÓN Y DISPERSIÓN

Con la instrucción alineada, lo que resta es decodificar parcialmente las operaciones y dispersar las instrucciones a los grupos y unidades funcionales que las ejecutarán. La codificación de operaciones similar a RISC hace que la decodificación sea relativamente simple.

Dado que los clústeres y las unidades funcionales son entidades separadas geográficamente, las conexiones son cables largos que funcionan lentamente debido a la disminución de las capacidades de conducción de los transistores cada vez más pequeños en la tecnología actual.

# UNIDADES FUNCIONALES

Las máquinas modernas incluyen una variedad de tipos de unidades funcionales, aunque la ALU sigue siendo la base de muchas operaciones.

Estas unidades son funcionales en el sentido de que su salida es una función bien definida de sus entradas. Dependiendo de la cantidad de niveles de lógica necesarios para implementar una función en particular, se puede optar por dividir los niveles en múltiples ciclos, pero es posible que los resultados correspondientes no estén disponibles hasta varios ciclos después.

# ESPECULACIÓN DE CONTROL

Consiste en elevar un bloque básico de instrucciones sobre un Branch. Las instrucciones se vuelven especulativas y no pueden comprometerse hasta que se conozca la condición del Branch.

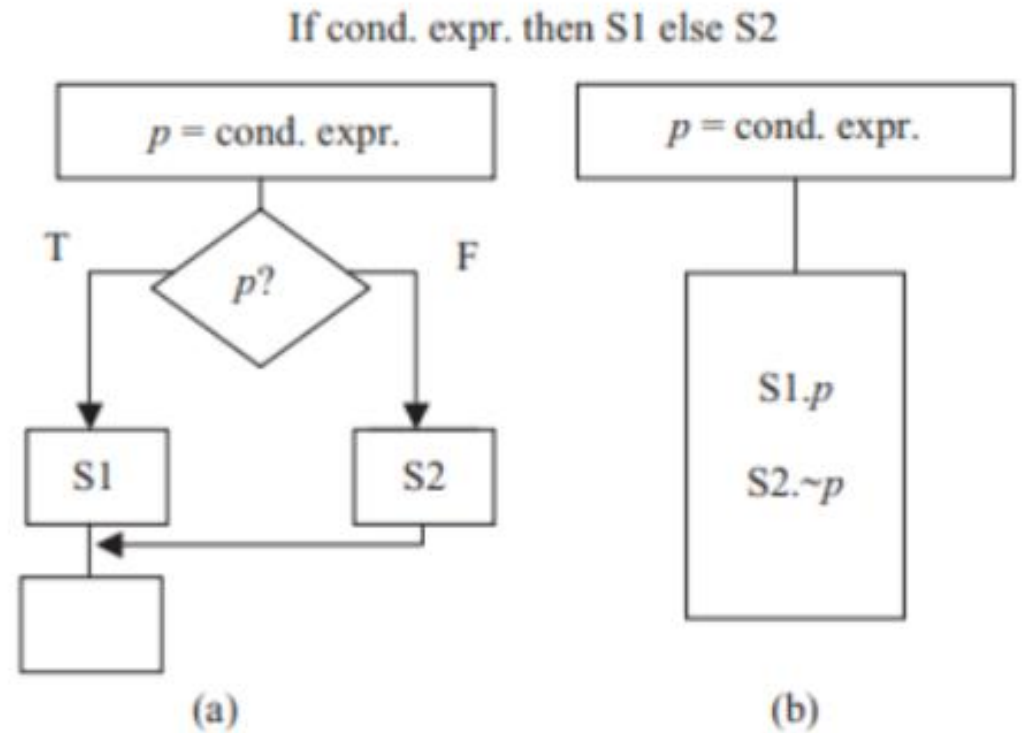
Las ventajas son que los recursos se pueden utilizar mejor en partes de los programas con bajo ILP, y las instrucciones con latencia larga se pueden iniciar antes.

Sin embargo, además de tener que almacenar temporalmente los resultados de las instrucciones especulativas hasta que se resuelva la condición otro posible problema es el de las excepciones.

# PREDICACIÓN

En un procesador VLIW el número de saltos condicionales a ejecutar se puede reducir utilizando la predicación. La forma más simple es utilizando un condicional.

El valor del predicado solo se necesita cuando los resultados deben confirmarse. Además, las declaraciones en S1 y S2 se pueden ejecutar simultáneamente.

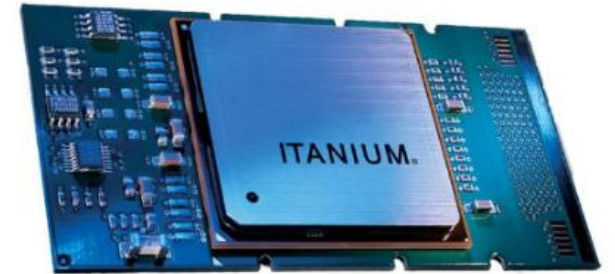


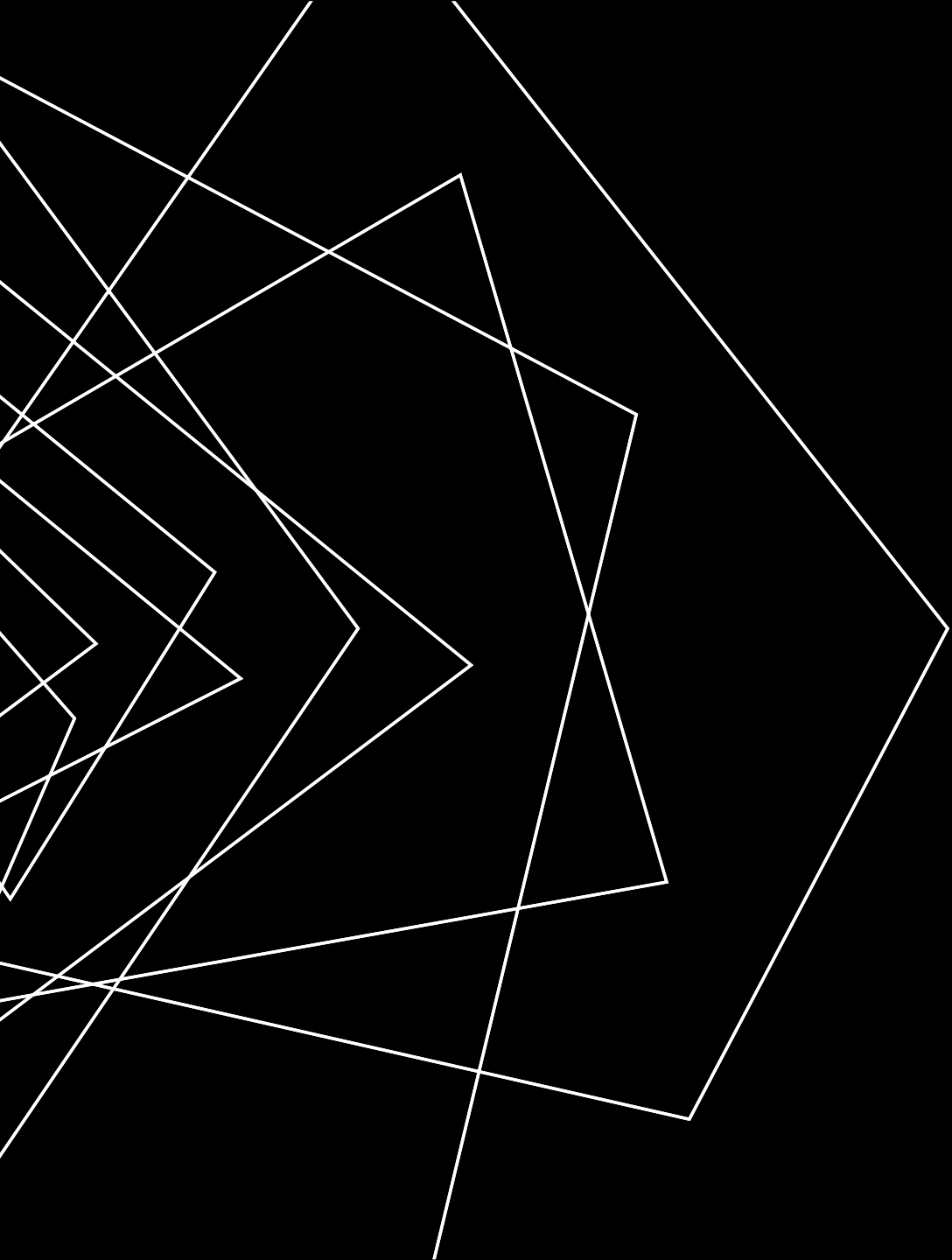
(a) Código usual, (b) Código predicado

# IMPLEMENTACIONES

A principios de 1990, Intel presentó el microprocesador i860 RISC, el cual tenía dos modos de operación: uno escalar y uno VLIW. Con esto se implementó una forma cruda de compresión de código, pero sus compiladores eran de calidad insuficiente para satisfacer el mercado.

Algunos ejemplos contemporáneos incluyen los procesadores de medios TriMedia de NX el DSP Super Harvard Architecture Single-Chip Computer de Analog Devices y el FR-V de Fujitsu, los cuales tienen éxito como procesadores de medios integrados para dispositivos electrónicos de consumo.





**GRACIAS**