ARQUITECTURA DE COMPUTADORS

Simulació processador Superescalar
Practica 1

UNIVERSITAT ROVIRA I VIRGILI

Universitat Rovira i Virgili
Escola Tècnica Superior d'Enginyeria
Grau d'Enginyeria Informàtica

Tarragona 2024

Albert Coca Rodríguez
Raul Martín Morales

Índex:

1. Intel Core i5-14400

Información general:

1-K via del procesador			
Fetch		8	
Decode		6	
Issue		6	
Commit		12	
2-Tamaño buffers			
ruu		512	
lsq		192 load(ancho 3) i 114 store(ancho 2) => 256	
3-Caches L1 i L2			
	L1I	L1D	L2
Sets	64	64	2048
Mida	32KB	48KB	1280KB
Vias	8	12	10
4-Ancho de banda y latencia			
Latència		16	
Ancho de banda		64bits = 8 bytes	
5-ALU's			
ALU Int		5	
ALU Float		3	
Int mult		2	
Float mult		2	
Puntos de acceso a memoria		5	

Este procesador intel utiliza una arquitectura “Raptor Lake” que utiliza un conjunto de P-cores(6) i E-cores(4) los cuales son muy similares a los de la arquitectura “Golden Cove”.

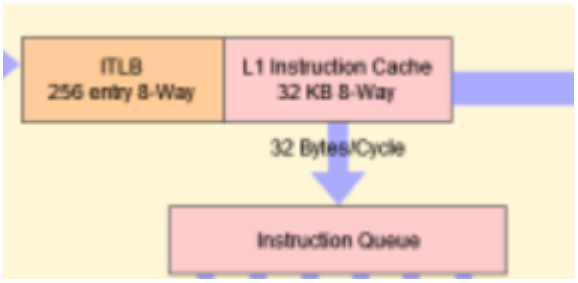
Especificaciones CPU:

1.1 Instruccions per cilce

K-via	Instruccions/cicle
Fetch	8
Decode	6
Issue	6
Commit	12

Fetch:

En esta etapa se reciben las instrucciones, se leen y se guardan en un buffer(Instruction buffer).

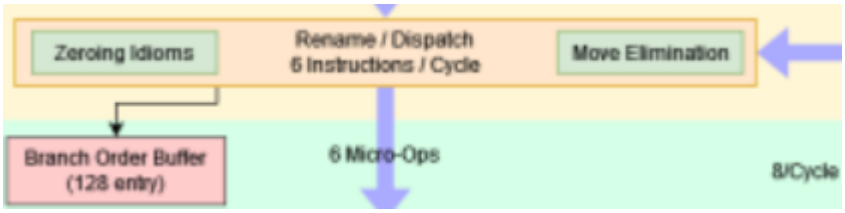


Decode:

En la etapa de decode la CPU determina qué instrucciones se harán. Las instrucciones decodificadas se guardan en un buffer(Dispatch buffer).

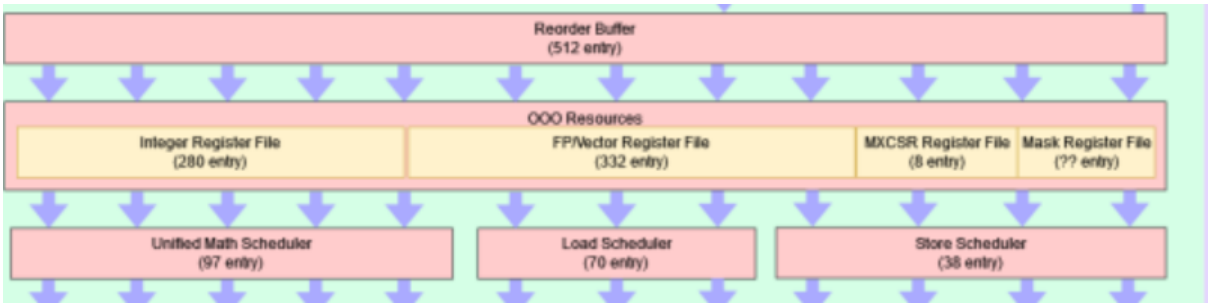
Issue:

En esta etapa las instrucciones se desplazan hasta el ReOrder Buffer donde se enviarán a las ALUs correspondientes para ser ejecutadas.

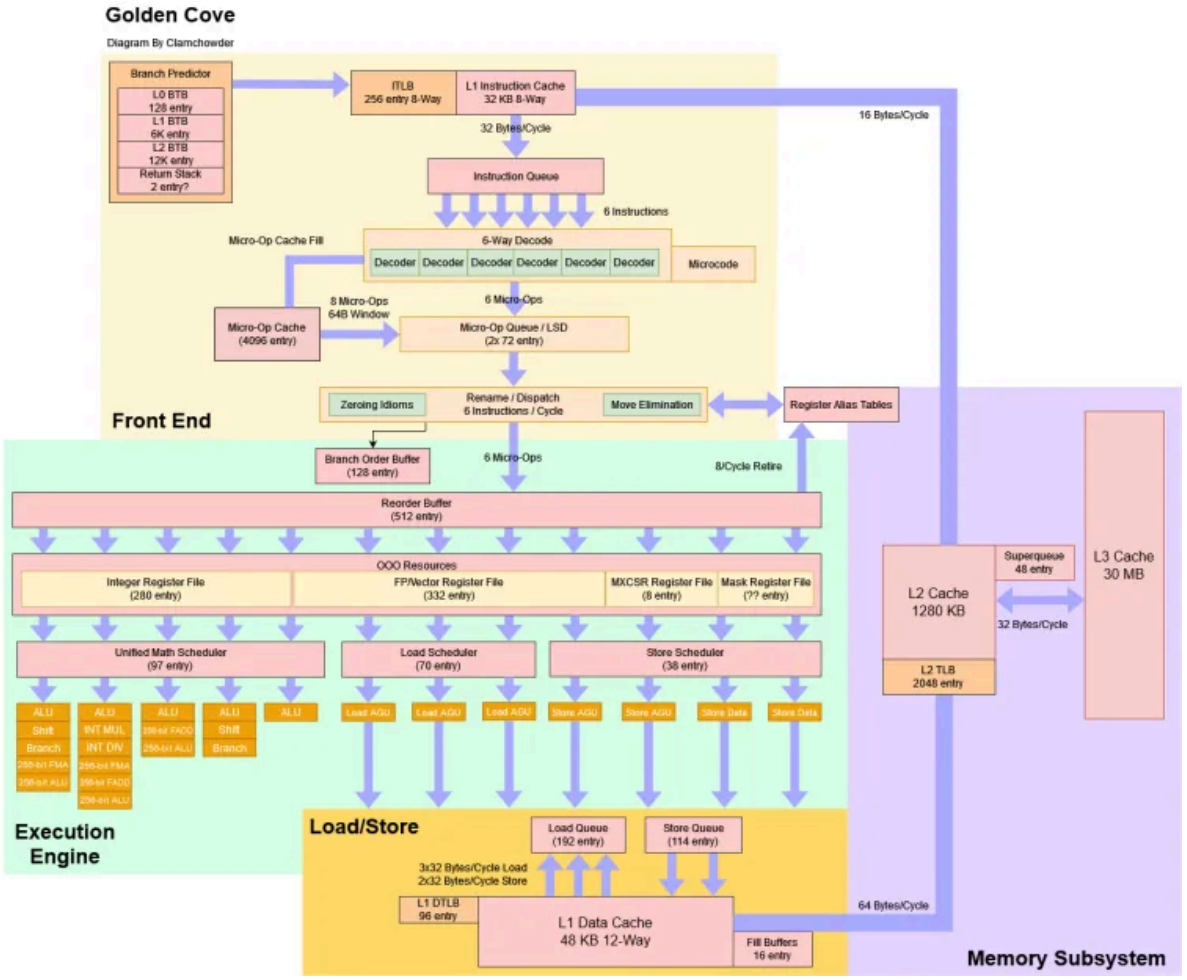


Commit:

En esta etapa se retiran las instrucciones en orden del ReOrder Buffer y se envía una señal para decir que se han ejecutado correctamente. Las instrucciones completas se guardan en el store buffer.

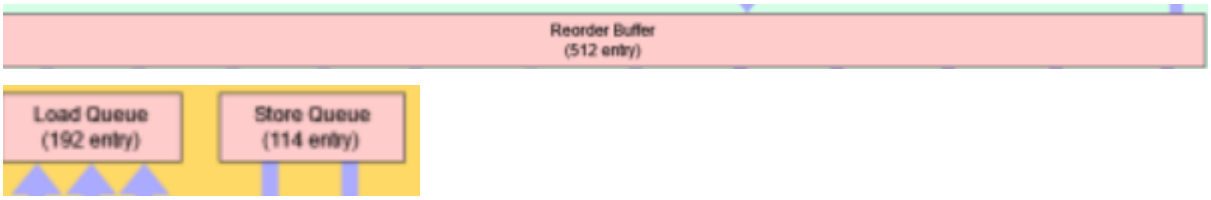


Esquema:


Golden Cove
Diagram By Clamchowder
Branch Predictor
L0 BTB
128 entry
L1 BTB
6K entry
L2 BTB
12K entry
Return Stack
2 entry?
ITLB
256 entry 8-Way
L1 Instruction Cache
32 KB 8-Way
16 Bytes/Cycle
32 Bytes/Cycle
Instruction Queue
6 Instructions
Micro-Op Cache Fill
6-Way Decode
Decoder
Decoder
Decoder
Decoder
Decoder
Decoder
Microcode
8 Micro-Ops
64B Window
6 Micro-Ops
Micro-Op Cache
(4096 entry)
Micro-Op Queue / LSD
(2x 72 entry)
Zeroing Idioms
Rename / Dispatch
6 Instructions / Cycle
Move Elimination
Register Alias Tables
Front End
Branch Order Buffer
(128 entry)
6 Micro-Ops
8/Cycle Retire
Reorder Buffer
(512 entry)
OOO Resources
Integer Register File
(280 entry)
FP/Vector Register File
(332 entry)
MXCSR Register File
(8 entry)
Mask Register File
(?? entry)
Unified Math Scheduler
(97 entry)
Load Scheduler
(70 entry)
Store Scheduler
(38 entry)
ALU
Shift
Branch
INT MUL
INT DIV
Load AGU
Store AGU
Store Data
Execution
Engine
Load/Store
Load Queue
(192 entry)
Store Queue
(114 entry)
3x32 Bytes/Cycle Load
2x32 Bytes/Cycle Store
L1 DTLB
96 entry
L1 Data Cache
48 KB 12-Way
Fill Buffers
16 entry
64 Bytes/Cycle
L2 Cache
1280 KB
Superqueue
48 entry
32 Bytes/Cycle
L2 TLB
2048 entry
L3 Cache
30 MB
Memory Subsystem

1.2 Tamaño buffer

ruu	512
lsq	192 load(ancho 3) i 114 store(ancho 2)



1.3 Caches

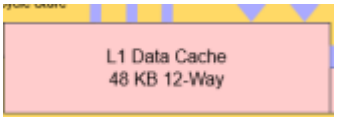
	L1I	**L1D**	**L2**
Sets	64	64	2048
Mida	32KB	48KB	1280KB
Vias	8	12	10

Tamaño Caché(bytes) = SETS * VIAS * 64

Cache L1I:

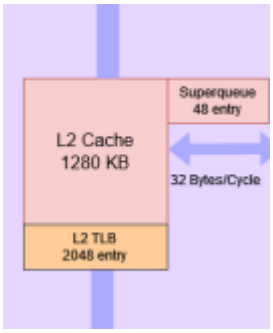
32*1024 = 64 * 8 * 64

Cache L1D:



48*1024 = 64 * 12 * 64

Cache L2:



1280*1024 = 2048 * 10 * 64

1.4 Ancho de banda i latencia

Latencia	16
Ancho de banda	64 bits => 8 bytes

First chunk = (CAS + tRCD) × (CPU Clock / Memory Clock)

Inter chunk = CPU_Clock/ Memory_Data_Rate

fent servir DDR4
CAS = 24
tRCD = 24
Frecuencia del procesador = 2500 MHz GHz
Frecuencia de la memoria = 3200 MHz -> 1600 efectius

First chunk = (24 + 24) * (4600/1600) = 75
Inter chunk = 2500 / 3200 = 0,78125 => 1

1.5 ALU's

ALU Int	5
ALU Float	3
Int mult	2
Float mult	2
Puntos de acceso a memoria	5

2. AMD Ryzen 57600X

Información general:

1-K via del procesador			
Fetch		8	
Decode		4	
Issue		6	
Commit		9	
2-Tamaño buffers			
ruu		320 => 256	
lsq		136 load(amplada 3) i 64 store(amplada 2) => 256	
3-Caches L1 i L2			
	L1I	L1D	L2
Sets	64	64	2048
Mida	32KB	32KB	1MB
Vias	8	8	8
4-Ancho de banda y latencia			
Latencia		16	
Ancho de banda		64 bits => 8 bytes	
5-ALU's			
ALU Int		4	
ALU Float		6	
Int mult		1	
Float mult		2	
Puntos de acceso a memoria		3	

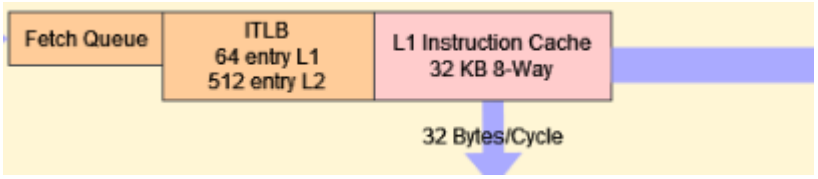
Este procesador usa la arquitectura Zen 4 que usa 6 cores AMD.

Especificaciones CPU:

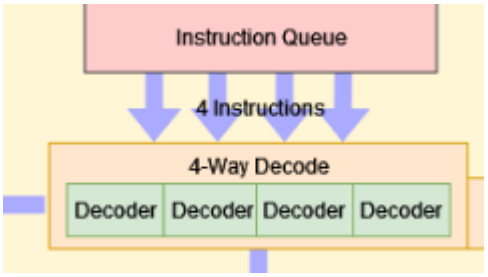
2.1 Instrucciones por ciclo

K-via	Instrucciones/ciclo
Fetch	8
Decode	4
Issue	6
Commit	9

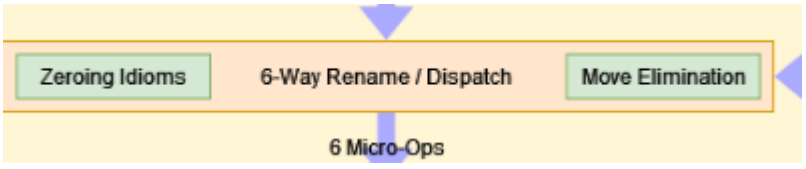
Fetch:



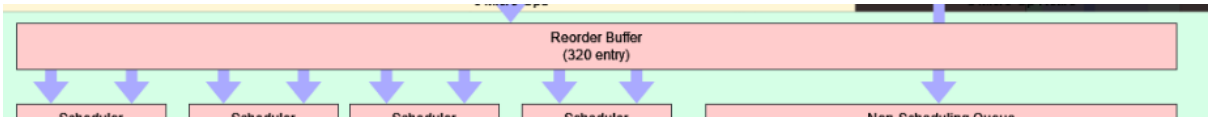
Decode:



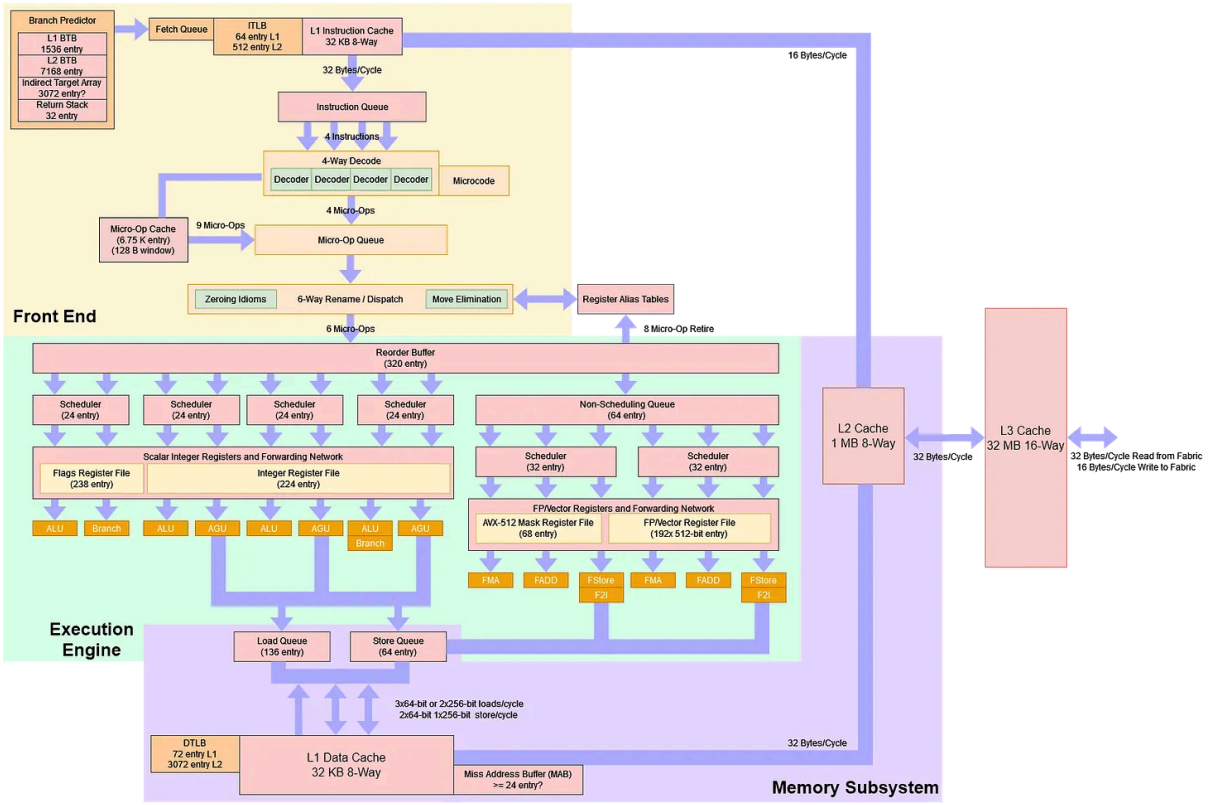
Issue:



Commit:

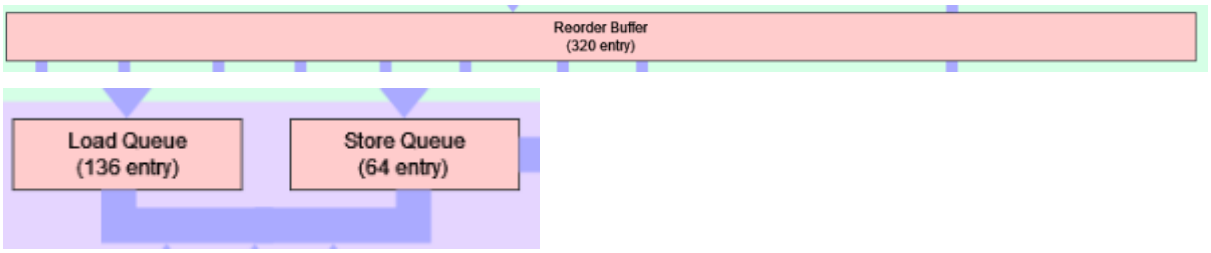


Esquema:



2.2 Tamaño buffer

ruu	320
lsq	136 load(ancho 3) i 64 store(ancho 2)

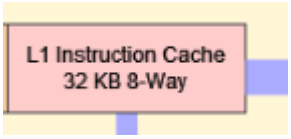


2.3 Caches

	L1I	L1D	L2
Sets	64	64	2048
Mida	32KB	32KB	1MB
Vias	8	8	8

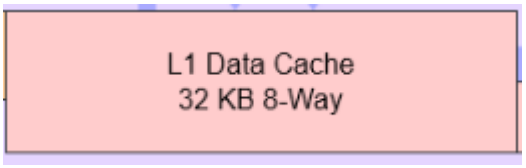
Tamaño Caché(bytes) = SETS * VIAS * 64

Cache L1I:



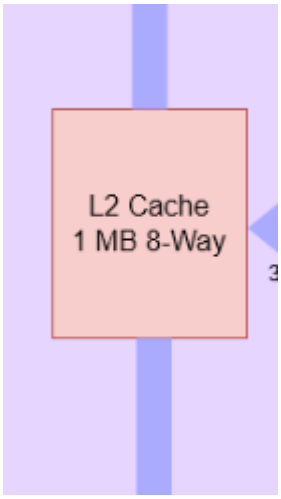
32*1024 = 64 * 8 * 64

Cache L1D:



32*1024 = 64 * 8 * 64

Cache L2:



1024*1024 = 2048 * 8 * 64

2.4 Ancho de banda i latencia

Latencia	16
Ancho de banda	64 bits => 8 bytes

First chunk = (CAS + tRCD) × (CPU Clock / Memory Clock)
Inter chunk = CPU_Clock/ Memory_Data_Rate
fent servir DDR5
CAS = 24
tRCD = 24
Frecuencia del procesador = 3400MHz
Frecuencia de la memoria = 3200 MHz -> 1600 efectivos

First chunk = (24+24) *(3400/1600) = 102=>100
Inter chunk = 3400/3200 = 1,0625 => 1

2.5 ALU's

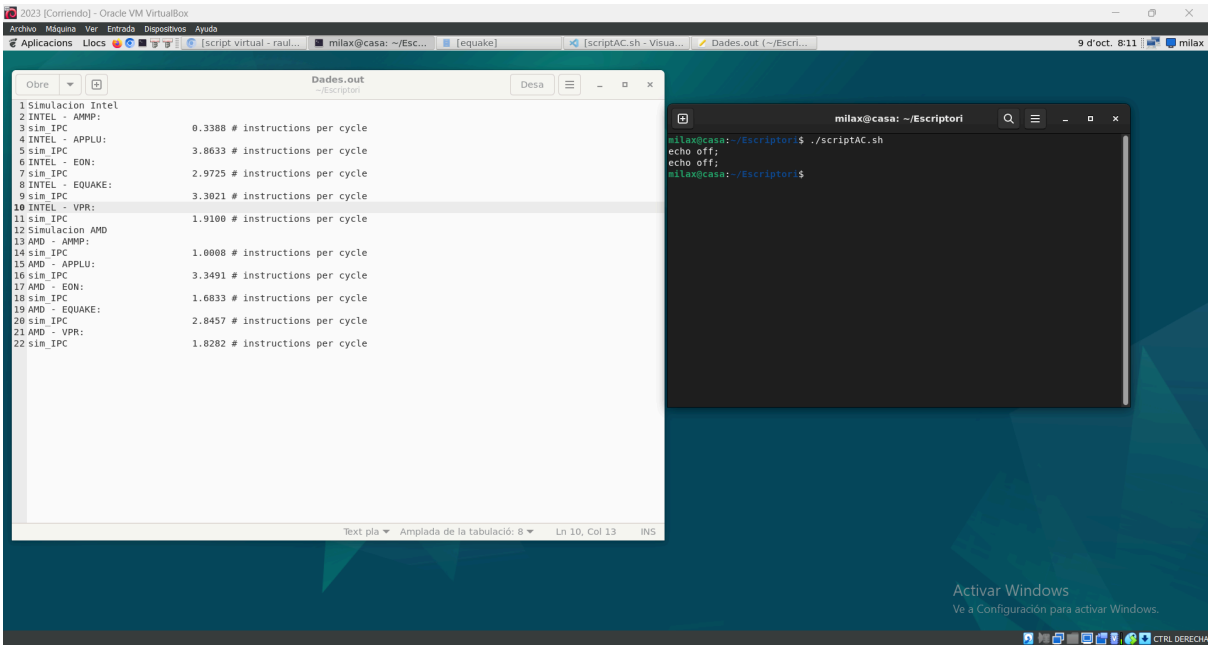
ALU Int	4
ALU Float	6
Int mult	1
Float mult	2
Puntos de acceso a memoria	3

3. Simulación

Paràmetros:

Así es como hemos configurado los dos procesadores en el simplescalar:

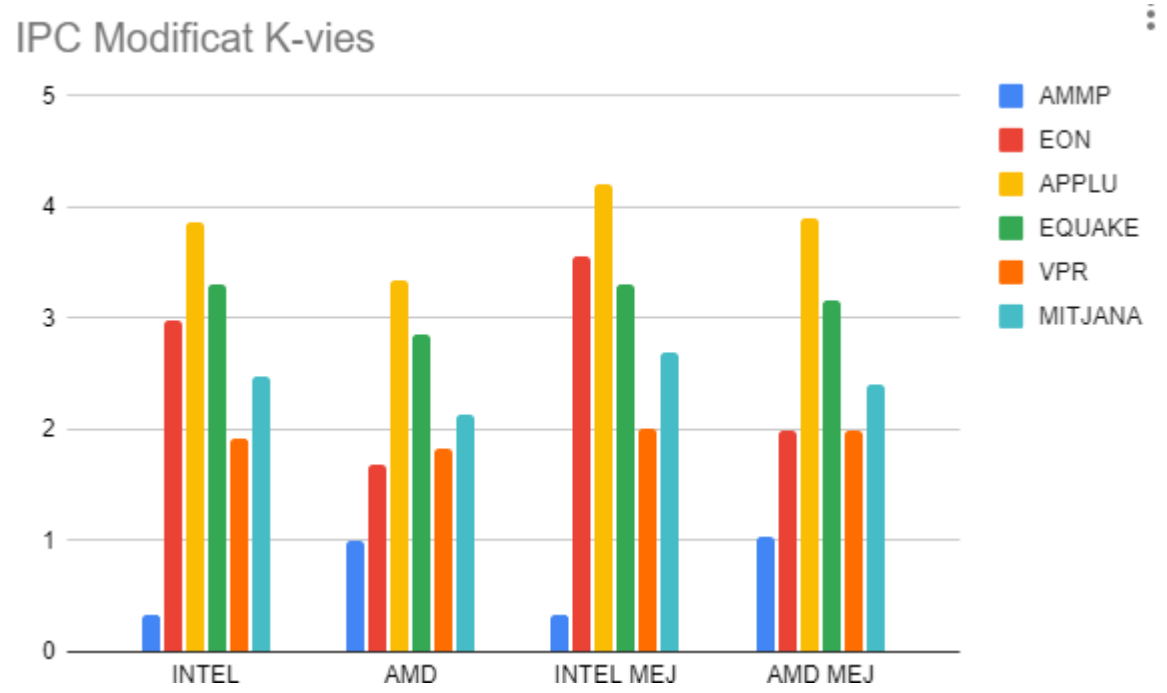
Parámetros Intel	Parámetros AMD
-fetch:ifqsize 8 -decode:width 6 -issue:width 6 -commit:width 16 (12) -ruu:size 512 -lsq:size 256 -mem:lat 75 1 -mem:width 64 -res:memport 5 -res:ialu 5 -res:imult 2 -res:fpalu 3 -res:fpmult 2 -cache:dl1 dl1:64:64:8:l -cache:il1 il1:64:64:8:l -cache:dl2 ul2:2048:64:8:l	-fetch:ifqsize 8 -decode:width 4 -issue:width 6 -commit:width 9 -ruu:size 256 -lsq:size 256 -mem:lat 100 1 -mem:width 64 -res:memport 3 -res:ialu 4 -res:imult 1 -res:fpalu 6 -res:fpmult 2 -cache:dl1 dl1:64:32:8:l -cache:il1 il1:64:32:8:l -cache:dl2 ul2:2048:1024:8:l



Millores proposades:

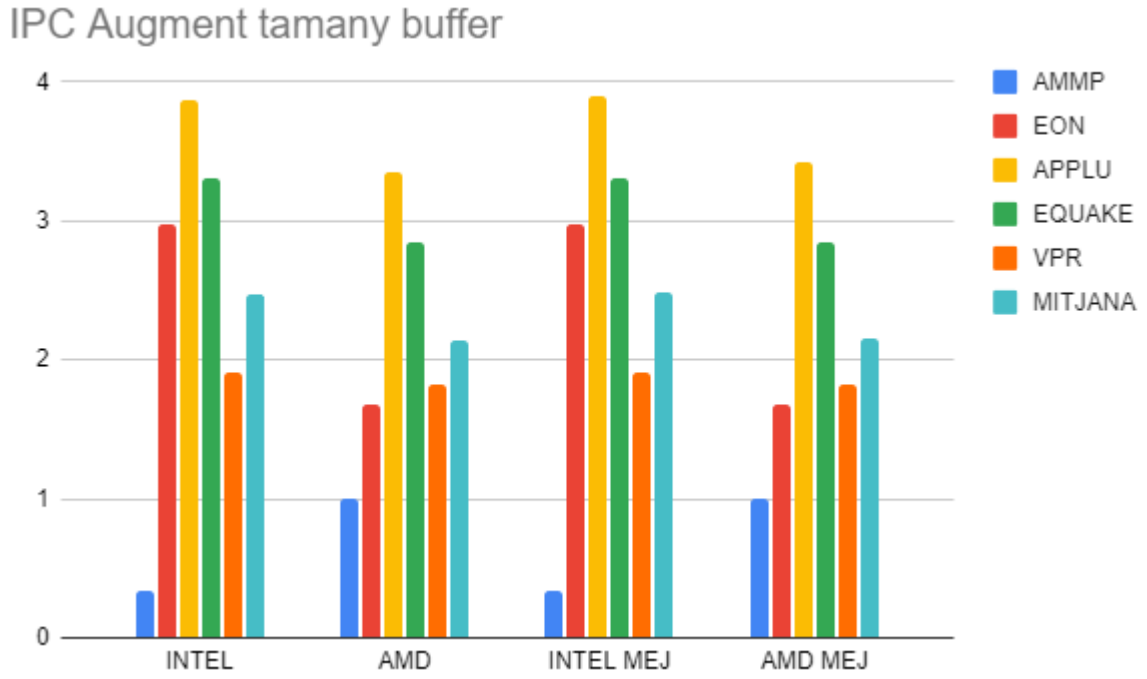
3.1 Modificación de las k-vias:

Aumentar el tamaño del buffer de búsqueda de instrucciones (fetch:ifqsize):
Cambiar de 8 a 16 para permitir más instrucciones en cola, y permitir una búsqueda más rápida de instrucciones.
Aumentar el ancho del decode, issue y commit, esto permitirá que el procesador maneje más instrucciones por ciclo.
Le sumamos 4 al valor por defecto del -decode:width y -issue:width.
Y también aumentaremos el -commit:width de 16 a 20 en Intel, de 9 a 15 en el AMD.



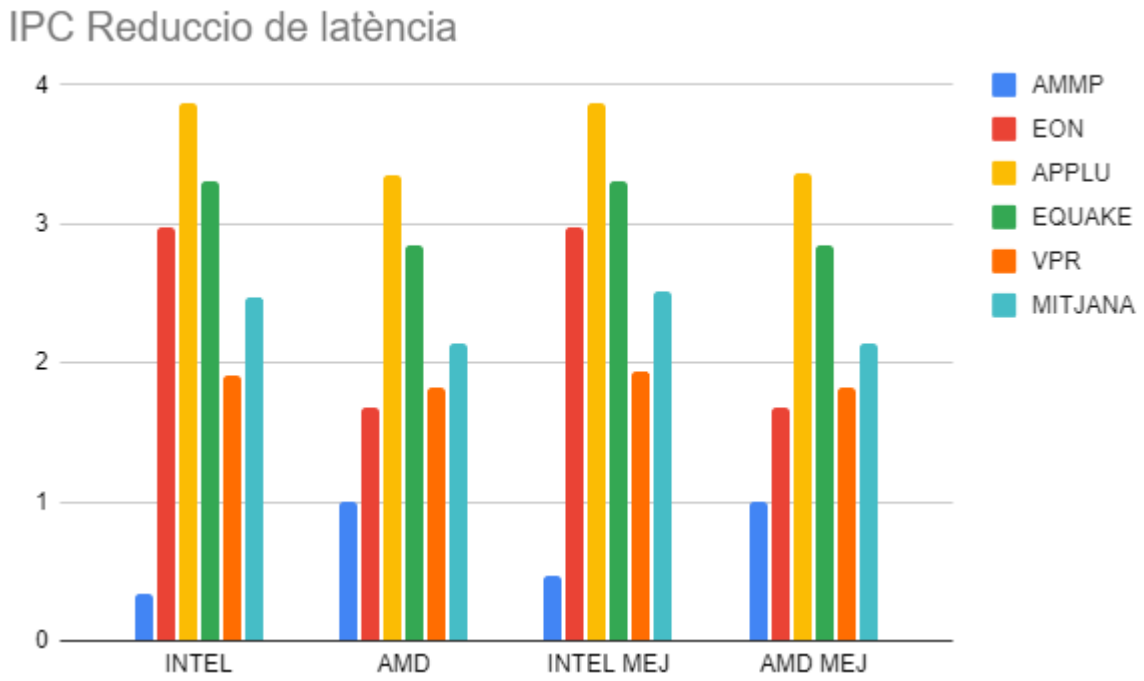
3.2 Aumentar tamaño del buffer RUU y LSQ:

Aumentar el tamaño del RUU y LSQ permite al procesador mantener más instrucciones y operaciones en espera, lo que puede reducir las ineficiencias debidas a dependencias de datos o latencias de memoria.
Pero este aumento del buffer puede incrementar la complejidad del procesador y también el consumo.
En el caso del Intel -ruu:size de 512 a 1024 i el tamaño de -lsq:size de 256 a 512.
En el caso del AMD-ruu:size de 256 a 512 i el tamaño de -lsq:size de 256 a 512.



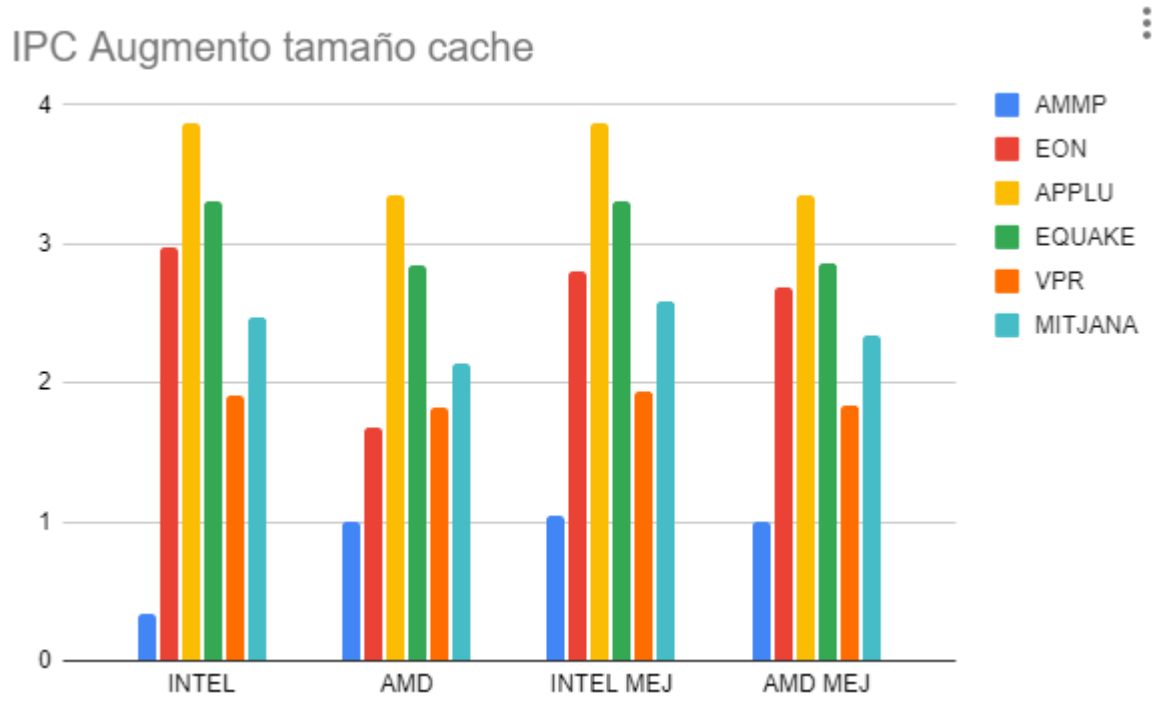
3.3 Reducir la latencia de la memoria:

Reducir -mem:lat a 50. en el Intel i en el de AMD lo reduciremos a 75.
Reducir esta latencia creemos que debería mejorar el rendimiento.

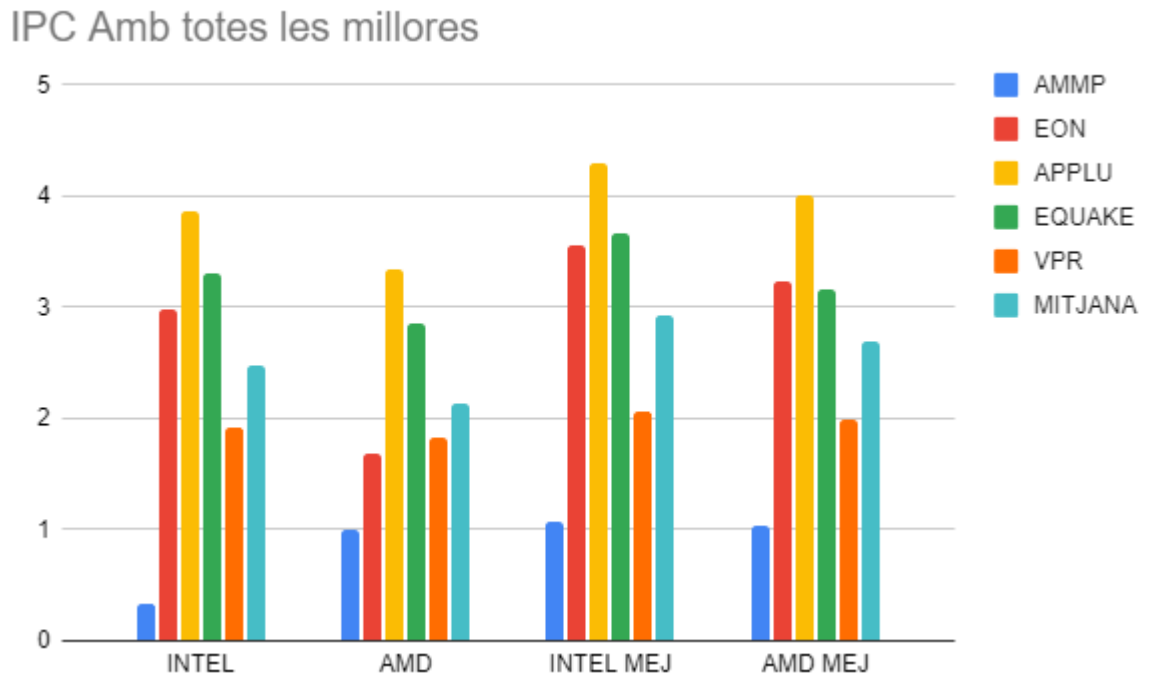


3.4 Aumentar el tamaño de la cachés L1 i L2:

Actualmente tenemos una caché L1 de 64 bloques con 64 bytes por bloque en el Intel. Probamos a aumentar el tamaño a 128 bloques en el L1 para mejorar el acceso rápido a los datos más frecuentemente utilizados. En la caché L2 también duplicaremos su tamaño. En los caches de AMD haremos lo mismo, aumentaremos también los tamaños de bloque.



3.5. Amb tots els canvis:



4. Comparativa de AMD y Intel al Gaming:

En cuanto a datos e información se ha podido ver que Ryzen 7 5700X3D proporciona un rendimiento mayor que el i5-14400, haciéndolo así al procesador de AMD mucho más atractivo para los amantes de los videojuegos ya que su CPU ofrece más rapidez en los videojuegos, aun así también es productivo para otras aplicaciones. En cuanto a rendimiento y precio el de AMD está superando a Intel en el entorno del gaming y tecnología avanzada, pero con un presupuesto más ajustado, el de Intel sigue siendo una opción válida, aun teniendo un rendimiento inferior.

5. Bibliografía

https://chipsandcheese.com/p/popping-the-hood-on-golden-cove

https://chipsandcheese.com/p/going-armchair-quarterback-on-golden-coves-caches

https://chipsandcheese.com/p/amds-zen-4-part-1-frontend-and-execution-engine

https://www.anandtech.com/show/17585/amd-zen-4-ryzen-9-7950x-and-ryzen-5-7600x-review-retaking-the-high-end/8

https://www.anandtech.com/show/11170/the-amd-zen-and-ryzen-7-review-a-deep-dive-on-1800x-1700x-and-1700/8

https://www.quora.com/How-many-ALU-and-FPU-are-in-the-Cori7-9th-generation-processor

https://www.agner.org/optimize/microarchitecture.pdf