Casi siempre elegimos la memoria RAM mirando únicamente dos datos: cuántos gigabytes tiene y qué cifra de velocidad aparece en la caja. Con esos dos números podemos gastar bastante más dinero y terminar con un ordenador que funciona igual o incluso peor.

Para comprar correctamente debemos considerar al menos **la capacidad, el número de módulos, los canales de memoria, la velocidad, las latencias, la plataforma y el uso real del equipo**. Si además vamos a ejecutar modelos de lenguaje en local, la memoria adquiere todavía más importancia: puede determinar qué modelos podemos cargar y cuántos tokens por segundo obtenemos.

En esta guía veremos cómo elegir RAM para un equipo de trabajo, gaming, creación de contenido e inferencia local de LLMs mediante aplicaciones como llama.cpp.

En esta guía

1. Cuánta memoria necesitamos

La capacidad es el primer parámetro que debemos acertar porque ninguna frecuencia ni latencia puede compensar la falta de memoria.

Cuando agotamos la RAM disponible, el sistema operativo empieza a mover páginas de memoria al almacenamiento. Aunque tengamos un SSD NVMe rápido, su latencia y su ancho de banda son muy inferiores a los de la RAM. El resultado son pausas, aplicaciones que tardan en responder y un rendimiento muy irregular.

El caso contrario también es importante: **la RAM que no utilizamos no acelera el ordenador**. Instalar 128 GB no mejora un equipo cuyo consumo máximo nunca supera los 20 GB.

USO PRINCIPAL	CAPACIDAD RECOMENDABLE	CONFIGURACIÓN
Ofimática, navegación y estudio	16–32 GB	2 × 8 GB o 2 × 16 GB
Gaming y uso general exigente	32 GB	2 × 16 GB
Edición de vídeo, fotografía y desarrollo	64 GB	2 × 32 GB
Máquinas virtuales y laboratorios técnicos	64–128 GB	2 × 32 GB, 2 × 48 GB o 2 × 64 GB

USO PRINCIPAL	CAPACIDAD RECOMENDABLE	CONFIGURACIÓN
LLMs pequeños cargados principalmente en GPU	32–64 GB	2 × 16 GB o 2 × 32 GB
LLMs con offload parcial entre CPU y GPU	64–128 GB	2 × 32 GB, 2 × 48 GB o 2 × 64 GB
LLMs grandes mediante CPU	128–256 GB o más	Según los canales y límites de la plataforma

Los 16 GB continúan siendo suficientes para equipos sencillos, pero dejan poco margen cuando abrimos muchas pestañas, utilizamos aplicaciones de comunicación y ejecutamos varias herramientas simultáneamente. Para un equipo nuevo, **32 GB constituyen actualmente el punto de partida más equilibrado**.

Cuando vamos a ejecutar LLMs no debemos calcular la capacidad únicamente a partir del tamaño del archivo del modelo. También necesitamos memoria para el sistema operativo, la aplicación, los búferes de cálculo, el contexto, la caché KV y cualquier programa que permanezca abierto.

> ***Margen recomendado para IA local.*** *El modelo debería caber sin depender de swap y deberíamos conservar varios gigabytes libres. Como orientación práctica, conviene disponer de entre un 20 y un 30 % más de RAM que el consumo previsto del modelo y sus cachés.*

Un modelo GGUF de 40 GB no convierte automáticamente un equipo con 48 GB en una configuración cómoda. Puede llegar a cargar, pero cualquier crecimiento del contexto o consumo adicional del sistema puede provocar paginación y destruir el rendimiento.

2. Módulos, canales y zócalos

La cantidad de módulos importa tanto como su velocidad. Las plataformas domésticas actuales suelen disponer de **dos canales de memoria**. Para utilizar ambos debemos instalar al menos un módulo en cada canal, normalmente en los zócalos A2 y B2 indicados por el fabricante de la placa.

Un único módulo de 32 GB deja uno de los canales sin utilizar. Dos módulos de 16 GB permiten trabajar en doble canal y duplican el ancho de bus disponible, aunque la mejora observada en las aplicaciones depende de la carga de trabajo.

> ***Regla general.*** *Para una capacidad determinada, deberíamos comprar dos módulos antes que uno: 2 × 16 GB en lugar de 1 × 32 GB y 2 × 32 GB en lugar de 1 × 64 GB.*

Esta diferencia resulta especialmente importante durante la inferencia mediante CPU o gráficos integrados. La generación de tokens suele estar limitada por la velocidad con la que el procesador puede leer los pesos del modelo desde la memoria. Dejar un canal sin utilizar reduce drásticamente el ancho de banda disponible.

Cuatro zócalos no significan cuatro canales

Una placa con cuatro zócalos DIMM no tiene necesariamente cuatro canales. En una plataforma doméstica habitual seguimos teniendo dos canales, cada uno conectado a dos zócalos.

Al instalar cuatro módulos aumentamos la carga eléctrica soportada por el controlador de memoria. Como consecuencia, puede ser necesario reducir la velocidad, relajar las latencias o aumentar ligeramente el voltaje para conservar la estabilidad.

AMD, por ejemplo, especifica oficialmente DDR5-5600 para determinadas configuraciones de dos módulos en Ryzen 9000, pero reduce la velocidad oficial a DDR5-3600 cuando ocupamos los cuatro zócalos. La velocidad real alcanzable depende de la placa, el procesador, el diseño de los módulos y el firmware utilizado.

Por esta razón, normalmente resulta preferible instalar **2 × 32 GB en lugar de 4 × 16 GB** o **2 × 64 GB en lugar de 4 × 32 GB**. Además de facilitar velocidades superiores, dejamos dos zócalos libres.

Dos subcanales DDR5 no equivalen a doble canal

Cada módulo DDR5 divide internamente su interfaz de datos en dos subcanales independientes de 32 bits. Esto permite gestionar las solicitudes con mayor eficiencia que DDR4, pero no convierte un único módulo en una configuración de doble canal.

Un DIMM DDR5 continúa proporcionando en total 64 bits de datos. Para aprovechar los dos canales de memoria de una plataforma doméstica seguimos necesitando dos módulos correctamente instalados.

3. Qué significa la velocidad de la RAM

Es el número grande que encontramos en la caja: DDR5-5600, DDR5-6000, DDR5-6400 o DDR5-8000.

La unidad correcta son los **MT/s**, megatransferencias por segundo. Aunque muchas tiendas y fabricantes continúan utilizando MHz, DDR5-6000 no funciona con un reloj físico de 6000 MHz.

DDR significa *Double Data Rate*. La memoria transfiere datos en los dos flancos de cada ciclo de reloj. Por tanto, una memoria DDR5-6000 realiza 6000 millones de transferencias por segundo, pero su reloj físico es de aproximadamente 3000 MHz.

Más MT/s proporcionan más ancho de banda. En un canal de 64 bits, equivalente a 8 bytes, podemos calcular el ancho de banda teórico mediante esta fórmula:

```
Ancho de banda teórico = MT/s × 8 bytes × número de canales
```

Una configuración DDR5-6000 de doble canal ofrece, sobre el papel:

```
6000 × 8 × 2 = 96000 MB/s
96000 MB/s = 96 GB/s
```

CONFIGURACIÓN DE DOBLE CANAL	ANCHO DE BANDA TEÓRICO
DDR5-4800	76,8 GB/s
DDR5-5600	89,6 GB/s
DDR5-6000	96 GB/s
DDR5-6400	102,4 GB/s
DDR5-7200	115,2 GB/s
DDR5-8000	128 GB/s

Estas cifras son límites teóricos. El ancho de banda efectivo siempre será inferior debido a la sobrecarga del protocolo, el controlador, los patrones de acceso, la CPU y el software.

La frecuencia no nos indica cuánto tarda la memoria en responder a una solicitud concreta. Para entender esa parte debemos observar las latencias.

4. Qué significa la latencia CAS

En las especificaciones aparece como CL28, CL30, CL32, CL36 o CL40. CAS significa *Column Address Strobe*.

Los datos de la memoria se organizan mediante filas y columnas. Cuando la fila necesaria ya está activa, la latencia CAS indica cuántos ciclos transcurren desde que el controlador solicita una columna hasta que la memoria comienza a entregar el dato.

CL no expresa una cantidad de tiempo, sino una cantidad de ciclos. Como la duración de cada ciclo cambia con la velocidad, no podemos comparar dos memorias de diferente frecuencia mirando únicamente el número CL.

DDR5-6000 CL30 no tiene necesariamente menos latencia que DDR5-6400 CL32. Debemos convertir los ciclos a nanosegundos para saberlo.

CL no representa toda la latencia de la memoria

La latencia CAS es únicamente uno de los tiempos que intervienen en una operación. Los kits suelen mostrar una secuencia como 30-36-36-76, correspondiente a valores como CL, tRCD, tRP y tRAS.

Además de esos tiempos influyen el controlador de memoria, la proporción entre sus relojes, el número de rangos, la topología de la placa, la carga de los zócalos y la arquitectura del procesador.

La fórmula siguiente calcula la **latencia CAS absoluta**. Resulta útil para comparar kits, pero no representa la latencia completa que mediremos desde el procesador.

5. Cómo calcular la latencia CAS real

Podemos convertir la latencia CAS a nanosegundos mediante esta fórmula:

```
Latencia CAS absoluta en ns = CL × 2000 ÷ velocidad en MT/s
```

Para una memoria DDR5-6000 CL30:

```
30 × 2000 ÷ 6000 = 10 ns
```

Para una memoria DDR5-6400 CL32:

```
32 × 2000 ÷ 6400 = 10 ns
```

Los dos kits tienen la misma latencia CAS absoluta. DDR5-6400 proporciona más ancho de banda, pero no responde antes en esta fase concreta de la operación.

Esto no significa que ambos kits rindan exactamente igual. La memoria DDR5-6400 puede ofrecer más rendimiento en cargas sensibles al ancho de banda, siempre que el controlador pueda utilizarla sin introducir penalizaciones adicionales.

6. Tabla comparativa de latencias

MEMORIA	LATENCIA CAS ABSOLUTA	VALORACIÓN
DDR5-4800 CL40	16,7 ns	Configuración básica
DDR5-5600 CL46	16,4 ns	JEDEC con latencias altas
DDR5-5600 CL36	12,9 ns	Gama de entrada razonable
DDR5-6000 CL36	12 ns	Aceptable
DDR5-6000 CL32	10,7 ns	Buena
DDR5-6000 CL30	10 ns	Excelente equilibrio para AM5
DDR5-6400 CL32	10 ns	Más ancho de banda con el mismo CAS absoluto
DDR5-6000 CL28	9,3 ns	Gama alta
DDR5-6400 CL30	9,4 ns	Gama alta
DDR5-7200 CL34	9,4 ns	Entusiasta

MEMORIA	LATENCIA CAS ABSOLUTA	VALORACIÓN
DDR5-8000 CL38	9,5 ns	Entusiasta y muy dependiente de la plataforma
DDR4-3200 CL16	10 ns	Referencia económica en DDR4
DDR4-3600 CL16	8,9 ns	Excelente equilibrio en DDR4

DDR4-3600 CL16 tiene una latencia CAS absoluta inferior a DDR5-6000 CL30. Esto no significa que DDR4 sea más rápida en general. DDR5 proporciona bastante más ancho de banda y puede atender solicitudes con mayor eficiencia gracias a sus dos subcanales internos.

La comparación demuestra que no debemos utilizar un único parámetro para evaluar toda la memoria. **Necesitamos capacidad suficiente, ancho de banda adecuado y latencias razonables.**

7. Cómo afecta la RAM a la inferencia de LLMs

La compra cambia sustancialmente cuando el equipo se utilizará para ejecutar modelos de lenguaje en local.

La inferencia de un LLM autorregresivo consta de dos fases principales. Durante el procesamiento inicial del prompt podemos evaluar muchos tokens en paralelo, por lo que la capacidad de cálculo tiene gran importancia. Durante la generación, el modelo produce normalmente un token cada vez y debe recorrer repetidamente sus pesos.

Los desarrolladores de llama.cpp describen el procesamiento del prompt como una carga principalmente limitada por cálculo y la generación de tokens como una carga especialmente limitada por el ancho de banda de memoria.

Inferencia completamente cargada en la GPU

Cuando todo el modelo, la caché KV y los búferes necesarios caben en la VRAM, la velocidad de la RAM del sistema tiene una influencia limitada después de cargar el modelo.

En este escenario importan principalmente:

- Que tengamos suficiente RAM para el sistema y para cargar el modelo sin paginación.
- Que el almacenamiento permita leer rápidamente el archivo inicial.
- Que la GPU tenga suficiente VRAM y ancho de banda.
- Que no necesitemos mantener una copia adicional del modelo por el funcionamiento concreto del backend.

Para una estación con una GPU de 16 o 24 GB, normalmente resultan razonables 32 o 64 GB de RAM. Instalar DDR5-8000 en lugar de DDR5-6000 no transformará la velocidad de generación si todo el cálculo se realiza en la GPU.

Offload parcial entre RAM y VRAM

llama.cpp permite repartir el modelo entre CPU y GPU, lo que hace posible ejecutar modelos mayores que la VRAM disponible. El proyecto admite inferencia híbrida y permite controlar cuántas capas o tensores se almacenan en la GPU.

En este escenario la RAM deja de ser un simple espacio de carga. Las capas que permanecen en CPU deben leerse continuamente desde la memoria del sistema, por lo que el ancho de banda adquiere mucha importancia.

Si utilizamos una GPU de 16 GB para ejecutar un modelo de 35 o 70 mil millones de parámetros parcialmente alojado en RAM, debemos priorizar:

1. Capacidad suficiente para evitar swap.
2. Dos módulos para utilizar ambos canales.

3. La mayor velocidad estable que admita la plataforma.
4. Latencias razonables, sin sacrificar estabilidad.

En esta situación, pasar de un único módulo a una configuración de doble canal puede ser mucho más importante que reducir CL32 a CL30.

Inferencia completamente mediante CPU

Cuando todo el modelo se ejecuta en CPU, el procesador debe leer una gran cantidad de pesos desde la RAM para generar cada token. En muchos modelos cuantizados, añadir más núcleos deja de mejorar el rendimiento cuando ya hemos saturado los canales de memoria.

Podemos utilizar la siguiente aproximación para entender el límite físico:

```
Techo ideal de tokens/s ≈ ancho de banda efectivo ÷ datos de pesos leídos por token
```

Supongamos que obtenemos 80 GB/s efectivos y que cada token requiere recorrer aproximadamente 40 GB de pesos:

```
80 GB/s ÷ 40 GB = 2 tokens/s
```

Es únicamente una aproximación. La arquitectura del modelo, la cuantización, las capas activas, la caché, el contexto, los kernels y el procesador alteran el resultado. En los modelos MoE tampoco tienen por qué utilizarse todos los parámetros para cada token.

Aun así, la fórmula explica por qué el ancho de banda resulta tan importante: para inferencia CPU, subir de DDR5-5600 a DDR5-6400 puede ser más útil que reducir ligeramente la latencia CAS.

Gráficos integrados y memoria unificada

Una GPU integrada no dispone normalmente de VRAM dedicada y utiliza la RAM del sistema. En este caso la CPU y la GPU comparten tanto la capacidad como el ancho de banda.

Para inferencia mediante una GPU integrada debemos evitar un único módulo y priorizar la mayor velocidad estable posible. También debemos instalar capacidad adicional porque una parte de la RAM será utilizada por el sistema y otra por el backend gráfico.

El archivo del modelo no es todo el consumo

Para ejecutar un modelo con fluidez necesitamos que permanezcan en memoria los pesos utilizados durante la inferencia. El mapeo de archivos puede hacer que determinadas herramientas no atribuyan toda esa memoria directamente al proceso, pero las páginas siguen ocupando RAM o caché del sistema cuando son necesarias.

llama.cpp utiliza mapeo de memoria de forma predeterminada y ofrece opciones como mlock para intentar mantener el modelo en RAM. Si el sistema empieza a descartar páginas y releerlas constantemente desde el disco, la generación puede volverse extremadamente lenta.

También debemos reservar espacio para la caché KV. Su consumo aumenta con la longitud del contexto, el número de secuencias paralelas, la arquitectura del modelo y el tipo de datos utilizado para almacenarla.

> ***Para LLMs debemos priorizar en este orden:*** *capacidad suficiente, número de canales, estabilidad, ancho de banda y, finalmente, latencias más ajustadas.*

8. Qué velocidad elegir en AMD e Intel

AMD con socket AM5

DDR5-6000 con latencias cercanas a CL30 continúa siendo una recomendación práctica muy equilibrada para muchos equipos AM5. Existe una amplia disponibilidad de kits con perfil EXPO y suele ser una velocidad alcanzable mediante dos módulos en placas y procesadores adecuados.

Sin embargo, debemos distinguir entre la velocidad oficial y el overclocking. Por ejemplo, AMD especifica DDR5-5600 como velocidad máxima oficial en configuraciones de dos módulos para varios Ryzen 9000. DDR5-6000 mediante EXPO funciona fuera de esas especificaciones oficiales y no está garantizada en todas las combinaciones.

La configuración recomendada para la mayoría de equipos AM5 es:

```
32 GB: 2 × 16 GB DDR5-6000 CL30 EXPO
64 GB: 2 × 32 GB DDR5-6000 CL30 o CL32 EXPO
96 GB: 2 × 48 GB DDR5-6000 si aparece validado para la placa
128 GB: 2 × 64 GB a la mayor velocidad estable validada
```

Al aumentar la capacidad puede ser necesario bajar a 5600 MT/s o relajar las latencias. Para inferencia de LLMs debemos preferir 128 GB estables a 5600 antes que 96 o 128 GB inestables intentando alcanzar 6000.

No debemos comprar DDR5-7200 para AM5 suponiendo que será automáticamente más rápida. Dependiendo del procesador y sus relojes internos, una frecuencia superior puede obligar a utilizar relaciones menos favorables en el controlador y aumentar la latencia total.

Intel LGA1700

Las plataformas Intel Core de 12.ª, 13.ª y 14.ª generación pueden utilizar DDR4 o DDR5 dependiendo de la placa base. No podemos instalar ambos tipos en la misma placa.

Con DDR5, los kits de 6000 a 6400 MT/s suelen ofrecer un equilibrio razonable entre rendimiento, compatibilidad y precio. Las frecuencias superiores dependen mucho más de la placa, el procesador, el número de módulos y la calidad del controlador de memoria.

Intel Core Ultra 200S y 200S Plus

Las plataformas Intel Core Ultra 200S para escritorio admiten velocidades oficiales superiores a las de muchas generaciones anteriores. Intel especificó DDR5-6400 para los modelos 200S originales y DDR5-7200 para determinados modelos 200S Plus presentados en 2026.

Intel también ofrece compatibilidad con perfiles de overclocking de 8000 MT/s en determinadas configuraciones y soporte temprano para módulos CUDIMM de cuatro rangos y gran capacidad en placas seleccionadas de la serie 800. Esto no significa que cualquier placa, procesador o kit pueda funcionar automáticamente a esas velocidades.

Para un equipo Intel convencional podemos elegir:

```
32 GB: 2 × 16 GB DDR5-6400 CL32
48 GB: 2 × 24 GB DDR5-6400 CL32
64 GB: 2 × 32 GB DDR5-6400 CL32
96 GB o más: máxima capacidad validada a una velocidad estable
```

Las memorias de 7200, 8000 MT/s o superiores tienen sentido en sistemas entusiastas, con placas diseñadas expresamente para ellas y cuando el incremento de precio no nos obliga a reducir capacidad.

Qué es CUDIMM

Los módulos CUDIMM incorporan un controlador de reloj en el propio módulo. Su función es regenerar la señal de reloj y facilitar velocidades elevadas manteniendo la integridad de la señal.

CUDIMM no garantiza por sí solo que podamos utilizar una velocidad concreta. Necesitamos compatibilidad del procesador, la placa, la BIOS y el kit. Tampoco debemos asumir que toda DDR5-8000 requiere obligatoriamente CUDIMM, ya que existen configuraciones UDIMM convencionales capaces de alcanzar esas velocidades mediante overclocking.

Servidores y estaciones de trabajo

Cuando la prioridad es la inferencia CPU de modelos grandes, una plataforma doméstica de dos canales puede quedarse corta aunque acepte 192 o 256 GB.

Las plataformas HEDT y de servidor pueden disponer de cuatro, ocho, doce o más canales de memoria. Esto permite aumentar notablemente el ancho de banda agregado y resulta mucho más relevante para los tokens por segundo que una pequeña diferencia de CL.

No obstante, en sistemas con varios sockets debemos tener en cuenta NUMA. Acceder a la memoria conectada al otro procesador introduce latencia adicional y puede impedir que el rendimiento escale de forma lineal. llama.cpp incorpora estrategias NUMA, pero el resultado depende de la plataforma y la distribución real de las páginas de memoria.

9. Perfiles EXPO y XMP

Cuando compramos un kit DDR5-6000 CL30, el sistema no tiene por qué arrancar automáticamente a 6000 MT/s con esas latencias.

El módulo contiene valores JEDEC conservadores para maximizar la compatibilidad y uno o más perfiles de overclocking configurados por el fabricante:

- **AMD EXPO:** perfiles diseñados para plataformas AMD AM5.
- **Intel XMP:** perfiles de overclocking para plataformas Intel y muchas placas AMD compatibles.

AMD describe EXPO como una tecnología de perfiles optimizados para facilitar el overclocking de DDR5 en procesadores Ryzen para socket AM5. Intel XMP almacena combinaciones predefinidas de velocidad, latencias y voltaje que activamos desde la BIOS.

Después de montar el equipo debemos entrar en la BIOS, activar el perfil correspondiente y comprobar desde el sistema operativo la velocidad aplicada.

> ***Activar el perfil no garantiza estabilidad.*** *EXPO y XMP son formas sencillas de aplicar overclocking de memoria. La estabilidad final depende del kit, la placa, la BIOS, el controlador de memoria integrado en el procesador y el número de módulos.*

Para un servidor de LLM que funcionará durante días o semanas, la estabilidad es más importante que obtener unos pocos puntos adicionales de ancho de banda. Debemos probar la memoria después de activar el perfil y reducir la velocidad si aparecen errores, bloqueos o resultados corruptos.

10. ECC interno y ECC real

Todos los chips DDR5 incorporan mecanismos internos de corrección de errores denominados *on-die ECC*. Estos mecanismos ayudan a corregir errores producidos dentro de la propia matriz de memoria.

No equivalen a la memoria ECC utilizada en servidores y estaciones de trabajo. El ECC real protege también los datos almacenados y transferidos a través de la interfaz de memoria, siempre que el módulo, el procesador y la placa sean

compatibles.

Si necesitamos ECC para una estación de trabajo, virtualización, cálculo prolongado o inferencia crítica, debemos comprobar expresamente:

- Que los módulos sean ECC UDIMM o RDIMM, según corresponda.
- Que el procesador admita ese tipo de memoria.
- Que la placa implemente y active ECC.
- Que el firmware y el sistema operativo informen correctamente de los errores.

AMD indica compatibilidad ECC en determinados Ryzen 9000, pero condicionada al soporte proporcionado por la placa base.

11. Errores habituales al comprar RAM

1. **Mirar únicamente los MT/s.**
   Una frecuencia elevada acompañada por latencias muy relajadas puede aportar menos de lo que parece.
2. **Comprar un único módulo.**
   Dejamos uno de los canales sin utilizar y reducimos mucho el ancho de banda, especialmente perjudicial para gráficos integrados e inferencia CPU.
3. **Comprar la capacidad justa para un LLM.**
   El archivo del modelo no incluye todo el consumo de la aplicación, el contexto, la caché KV y el sistema operativo.
4. **Confundir cuatro zócalos con cuatro canales.**
   En una plataforma doméstica seguimos teniendo normalmente dos canales y cuatro módulos pueden obligarnos a reducir la velocidad.
5. **Mezclar kits separados.**
   Aunque tengan la misma referencia comercial, pueden utilizar chips o revisiones diferentes. El fabricante solo garantiza normalmente el funcionamiento conjunto de los módulos incluidos en el mismo kit.
6. **Comprar una frecuencia excesiva para AM5.**
   Una DDR5-7200 no tiene por qué superar a una DDR5-6000 bien configurada si el controlador debe utilizar una relación menos favorable.
7. **Pagar por latencias extremas cuando necesitamos capacidad.**
   Para LLMs es preferible disponer de 128 GB CL36 que quedarnos en 64 GB CL28 y no poder cargar el modelo.
8. **No activar EXPO o XMP.**
   El kit puede funcionar durante años con los valores JEDEC lentos sin que nos demos cuenta.
9. **No comprobar la QVL.**
   La lista del fabricante no contiene todos los kits que pueden funcionar, pero elegir uno validado reduce el riesgo de incompatibilidades.
10. **Suponer que on-die ECC es ECC completo.**
    La corrección interna de DDR5 no sustituye a una plataforma ECC.
11. **Olvidar la altura de los módulos.**
    Los disipadores altos pueden chocar con determinados disipadores de CPU.
12. **No probar la estabilidad.**
    Un perfil que permite iniciar el sistema puede seguir produciendo errores bajo carga prolongada.

12. Configuraciones recomendadas

Equipo básico de trabajo y estudio

```
32 GB
2 × 16 GB
DDR5-5600 CL36
```

Obtendremos capacidad suficiente para navegación intensiva, ofimática, videollamadas y multitarea. En estas cargas no suele compensar pagar mucho más por una memoria extrema.

Gaming con AMD AM5

```
32 GB
2 × 16 GB
DDR5-6000 CL30
Perfil EXPO
```

Es la configuración de referencia para un equipo AM5 orientado a juegos. Si además realizamos edición, virtualización o IA local, podemos pasar a 64 GB mediante 2 × 32 GB.

Gaming con Intel

```
32 o 48 GB
2 × 16 GB o 2 × 24 GB
DDR5-6400 CL32
Perfil XMP
```

Las velocidades superiores pueden aportar rendimiento adicional, pero requieren revisar la compatibilidad concreta del procesador y la placa.

Creación de contenido y máquinas virtuales

```
64 GB
2 × 32 GB
DDR5-6000 CL30 o CL32
```

En edición 4K, escenas 3D, compilaciones y virtualización, la capacidad suele importar más que una pequeña diferencia de latencia.

LLMs cargados completamente en una GPU

```
64 GB
2 × 32 GB
DDR5-5600 o DDR5-6000
```

Cuando el modelo cabe completamente en VRAM no necesitamos perseguir la RAM más rápida. Debemos reservar capacidad suficiente para el sistema, cargar modelos grandes y mantener otras aplicaciones abiertas.

LLMs con offload parcial en una plataforma AM5

```
96 o 128 GB
2 × 48 GB o 2 × 64 GB
```

```
DDR5-5600 o DDR5-6000
La mayor velocidad completamente estable
```

Aquí sí aprovecharemos el ancho de banda adicional porque parte del modelo será procesada mediante CPU. Debemos evitar cuatro módulos cuando podamos alcanzar la misma capacidad mediante dos.

LLMs con offload parcial en Intel

```
96 o 128 GB
Dos módulos de alta capacidad
DDR5-6000 o DDR5-6400
Perfil XMP validado para la placa
```

Podemos utilizar frecuencias superiores si el kit, la CPU y la placa aparecen validados, pero no debemos sacrificar capacidad ni estabilidad para obtener una cifra mayor.

Inferencia CPU de modelos grandes

```
128–256 GB o más
Todos los canales de memoria disponibles
La mayor velocidad estable
Preferiblemente ECC para cargas críticas
```

Para esta carga debemos elegir primero la plataforma y después la RAM. Una plataforma de ocho canales puede superar ampliamente a un equipo doméstico de dos canales aunque utilice módulos con menor frecuencia o CL más alto.

Inferencia mediante GPU integrada

```
64–128 GB
Dos módulos
La mayor velocidad estable admitida
Nunca un único módulo
```

La GPU integrada comparte la memoria con la CPU. Tanto la capacidad como el ancho de banda afectan directamente al modelo que podemos cargar y a su rendimiento.

Ampliación de un equipo DDR4

```
Uso general:
32 GB, 2 × 16 GB, DDR4-3200 CL16 o DDR4-3600 CL16

IA local o virtualización:
64–128 GB, preferiblemente mediante dos módulos si la plataforma lo permite
```

No necesitamos cambiar de plataforma únicamente para obtener DDR5. Si el procesador sigue siendo suficiente, ampliar la capacidad DDR4 puede resultar mucho más rentable.

13. Checklist antes de comprar

- ¿Nuestra placa utiliza DDR4 o DDR5?
- ¿Cuál es la capacidad máxima admitida por la placa y el procesador?

- ¿El kit contiene dos módulos?
- ¿Podemos alcanzar la capacidad necesaria sin ocupar los cuatro zócalos?
- ¿Estamos priorizando capacidad sobre latencias extremas para los LLMs?
- ¿Hemos calculado la latencia CAS absoluta en lugar de mirar únicamente CL?
- ¿La velocidad encaja con nuestra plataforma?
- ¿El kit utiliza EXPO, XMP o ambos?
- ¿Aparece en la QVL de la placa o en la lista de compatibilidad del fabricante?
- ¿La BIOS instalada reconoce módulos de esa capacidad?
- ¿Los módulos caben debajo del disipador del procesador?
- ¿Necesitamos ECC real?
- ¿El modelo de IA, la caché KV y el sistema cabrán sin utilizar swap?
- ¿Vamos a ejecutar el modelo completamente en GPU, parcialmente en CPU o completamente en CPU?
- ¿Hemos activado EXPO o XMP después del montaje?
- ¿Hemos realizado una prueba prolongada de estabilidad?

Guía actualizada en agosto de 2026 para plataformas DDR5 AMD AM5, Intel LGA1700 e Intel LGA1851, con recomendaciones adicionales para ampliaciones DDR4 e inferencia local de modelos mediante CPU, GPU, gráficos integrados y offload híbrido.