

# **Entregable 1 AC**

**Alejandro Luque Villegas**

Análisis del procesador de mi portátil donde se explorara la micro-arquitectura con la mayor precisión posible e incidiendo en aquellos aspectos que tengan que ver con ILP y los contenidos dados en la asignatura. El sistema operativo que se esta usando es Arch Linux así que todos los comandos correspondientes serán específicos para este sistema.

May 05, 2025

)

## 1. Identificación del procesador

Para identificar el procesador se ha usado el comando dado en la entrega del ejercicio:

```
$ lscpu
```

```
LINUC → ~ lscpu
Architecture:          x86_64
CPU op-mode(s):       32-bit, 64-bit
Address sizes:        39 bits physical, 48 bits virtual
Byte Order:           Little Endian
CPU(s):               8
On-line CPU(s) list: 0-7
Vendor ID:            GenuineIntel
Model name:           11th Gen Intel(R) Core(TM) i7-1165G7 @ 2.80GHz
CPU family:          6
Model:                140
Thread(s) per core:  2
Core(s) per socket:  4
Socket(s):           1
Stepping:             1
CPU(s) scaling MHz: 23%
CPU max MHz:         4700.0000
CPU min MHz:         400.0000
BogoMIPS:             5600.40
Flags:                fpv vme de pse tsc msr pae mce cx8 apic sep mtrr pge mca cmov pat pse36 clflush dts acpi mmx fxsr sse sse2 ss ht tm pbe syscall nx pdpe1gb rdtscp lm constant_tsc art arch_perfmon pebs bits rep_good nopl topstop_tsc cpuid aperf mperf tsc_known_freq pn1 pmlnqdq dtes64 monitor ds_cpl vnx est tm2 sse3 sdbg fma cx16 xtrp pdcm pcid sse4_1 sse4_2 x2apic movebe pdcmtsc_desceline_timer aes xsave avx f1co rorand lm abm Jndowprefetch cpuid,fault epb cpt_l2 cdp_l2 ssd lrs lbp stibp lrs_enhanced tpr_shadow flexpriority ept vpid ept_ad fsgsb64 tsc_offset bell avx2 steep btrm errm andtrotl avx512f avx512dq -based edx smp avx512fma clflushopt clflushopt_pt avx512cd sha_ni avx512bw avx512vl xsaveopt xsaves xgetbv1 xswaves xgetbv2 user-shutdown ida eras ptw hwp hwp_notify hwp_actwindow hwp_opp hwp_psd_rqq vmem avx512vnni umip pkru oskpe avx512_vnni gfnl vmem vpcimnlqdq avx512_vnni
Virtualization features:
Virtualization:        VT-x
Caches (use of all):
L1d:                  192 KiB (4 instances)
L1i:                  128 KiB (4 instances)
L2:                   5 MiB (4 instances)
L3:                   12 MiB (1 instance)
NUMA:
NUMA node(s):          1
NUMA node0 CPU(s):     0-7
Vulnerabilities:
Gather data sampling: Mitigation; Microcode
Ghostwrite:             Not affected
Itdb multithit:        Not affected
L1i cache sampling:    Not affected
Mdts:                  Not affected
Meltdown:              Not affected
Mmio stale data:       Not affected
Reg file data sampling: Not affected
Retired:               Not affected
Spec store bypass:      Mitigation; Speculative Store Bypass disabled via prctl
Spectre v1:              Mitigation; usercopy/swaps barriers and __user pointer sanitization
Spectre v2:              Mitigation; Enhanced / Automatic IBRS; IBPB conditional; RSB filling; PBRSB-eIBRS SW sequence; BHI SW loop, KVM SW loop
SMBds:                 Not affected
Txz async abort:        Not affected
```

Figura 1: Salida del comando lscpu de mi maquina.

En la imagen se puede observar información relevante que se comentara mas adelante. Cabe destacar que hay algunas curiosidades que no creo que merezcan mayor exploración pero me gustaría comentar aquí brevemente. Se muestra que tipo de Endianness usa el procesador, en nuestro caso Little Endian. También se muestra el tamaño de direcciones, tanto virtual como física.

## 2. Datos relevantes

- Nombre del CPU:** Intel Core i7-1165G7 @ 2.80GHz
- Cores (núcleos):** 4
- Threads (hilos):** 8 (Hyper-Threading)
- CPU(s) disponibles:** 8 (0-7)
- Familia:** 6
- Frecuencia base:** 2.80 GHz
- Frecuencia máxima (turbo):** 4.70 GHz
- L1d (datos):** 128 KiB (4× 32 KiB)
- L1i (instrucciones):** 192 KiB (4× 48 KiB)
- L2:** 5 MiB (4× 1.25 MiB)
- L3:** 12 MiB (compartido)

Al tener 4 cores y que cada core sea capaz de llevar 2 hilos, las CPUs disponibles son 8, ya que estas son CPUs **lógicas** disponibles.

### **3. Cache**

En cuanto a las unidades de memoria caché cabe destacar lo siguiente. La caché de Nivel 1 (L1), con 80 KB por núcleo (48 KB para datos y 32 KB para instrucciones), es la que ofrece un acceso más rápido y cada núcleo tiene su propia caché L1. Esta caché almacena la información que el núcleo está utilizando de manera más inmediata. La caché de Nivel 2 (L2) de 1.25 MB por núcleo y es más lenta que la L1, pero más rápida que la L3 y actúa como buffer entre la caché L1 y memoria principal. Por último, la caché de Nivel 3 (L3) es un bloque de 12 MB compartido por todos los núcleos del procesador. Su propósito principal es mejorar la coherencia de los datos que comparten los diferentes núcleos y facilitar una comunicación más eficiente entre ellos, optimizando el rendimiento en tareas multihilo.

#### **3.1. Coherencia**

En cuanto a los algoritmos para mantener coherencia, no se ha podido encontrar con exactitud qué algoritmo usa esta familia de procesadores, pero se puede asumir que entre las cachés L1 y L2 se utiliza algún algoritmo de snooping. También es posible que se utilice algún mecanismo parecido a MESIF [1] que es una variación del protocolo MESI.

### **4. Interconexión**

En cuanto a interconexión, en la familia de TigerLake se utiliza un *dual ring bus* que conecta los núcleos de la CPU, los gráficos integrados, la caché de último nivel (LLC), el controlador de memoria y el agente del sistema. Cada uno de estos componentes («agentes») tiene su propia interfaz con el anillo, permitiendo una comunicación eficiente.[2]

## Bibliografía

- [1] D. Kanter, «Common System Interface (CSI) Part 5: Directory Snooping». Accedido: 5 de mayo de 2025. [En línea]. Disponible en: <https://www.realworldtech.com/common-system-interface/5/>
- [2] I. Corporation, «The Architecture of Intel Processor Graphics Gen11», ago. 2019. Accedido: 5 de mayo de 2025. [En línea]. Disponible en: <https://www.intel.com/content/dam/develop/external/us/en/documents/the-architecture-of-intel-processor-graphics-gen11-r1new.pdf>