acontis technologies GmbH

SOFTWARE

Cache-Allocation-Technology-Guide

Cache-Allocation-Technology Guide

Version 1.0

Edition: November 21, 2023

© Copyright **acontis technologies GmbH**

Neither this document nor excerpts therefrom may be reproduced, transmitted, or conveyed to third parties by any means whatever without the express permission of the publisher. At the time of publication, the functions described in this document and those implemented in the corresponding hardware and/or software were carefully verified; nonetheless, for technical reasons, it cannot be guaranteed that no discrepancies exist. This document will be regularly examined so that corrections can be made in subsequent editions. Note: Although a product may include undocumented features, such features are not considered to be part of the product, and their functionality is therefore not subject to any form of support or guarantee.

Table of Contents

1 Manual CAT Optimization **4**
1.1 Default configuration 4
1.2 Manual configuration 4
1.3 Optimization 5
1.4 Counters 5
1.5 Resources 6

1 Manual CAT Optimization

1.1 Default configuration

On default CAT is configured to be shared equaly between all OS. Since version 8.0.01.03 the cache architecture (which cores are sharing which cache) is taken into calculation.

For example an x5-E3940 with 4 cores and 2 separated L2 caches:

```
L2-1 is shared between core index 0 and 1
L2-2 is shared between core index 2 and 3.
```

The architecture ist coded into CPU-ID information (see Intel Software Developer Manual, Volume 2, assembler directive 'cpuid' for details). For analysis reports RtosUpload can display such information using:

```
RtosUpload.exe -nosleep -nowait -idshow 29,0,1
RtosUpload.exe -nosleep -nowait -idshow 29,0,3
```

(The exact idshow value might change without notice but using 0,0,0 gives a hint where to start, so does 29,0,0)

Automatic configuration examples (since 8.0.01.03)

```
CPU- MASK
Host RTOS Caches
0x01 0x0E L2-1 is shared 50%; L2-2 is not shared
0x03 0x0C L2-1 and L2-2 are not shared
0x07 0x08 L2-1 is not shared; L2-2 is shared 50%
```

1.2 Manual configuration

In "RtosVM-UserManual.pdf" chapter "CAT Config-File settings" describes how to manually configure CAT. This is an example to modify the previous example (CPU masks 0x07 and 0x08) to assign 3/4 of L2-2 to RTOS:

```
[Vmf\RDT]
  "CatMaskL2Cos0Cpu2"=dword:C0
  "CatMaskL2Cos1Cpu2"=dword:3F
```

Hint: The higher bits should be used for the lower prio OS because they might be shared with an onboard GPU.

Additional information can be found in SDM vol. 3 chapter "17.19 Intel(R) Resource Director Technology Allocation Features"
https://www.intel.com/content/www/us/en/developer/articles/technical/intel-sdm.html

1.3 Optimization

It is possible to measure the effects of CAT using the processor performance counters to measure the L1/L2/L3 cache miss. This can be used to optimize the cache allocation mask. An example of how to measure this is included into our RTOS-32 RealtimeDemo. The demo has to be recompiled with `RT-MEASURE_PMC` being defined. Of course the CPU needs to support performance-counters and the used IDs.

Concept:

```
Configuration
- Disable all performance-counter (PMC) at MSR IA32_PERF_GLOBAL_CTRL
- Configure counter x at MSR IA32_PERFEVTSEL0, Reset counter x at MSR IA32_
 ↪PMC0
- Configure counter x+1 at MSR IA32_PERFEVTSEL0+1, Reset counter x+1 at␣
 ↪MSR IA32_PMC0+1
...
- Enable configured PMC at MSR IA32_PERF_GLOBAL_CTRL

Read counter
- Read PMC(X) using intrinsic ``__readpmc(x)`` ASM ``RDPMC``
```

Additional information can be found in SDM vol. 3 chapter “19.2 ARCHITECTURAL PERFORMANCE MONITORING”
https://www.intel.com/content/www/us/en/developer/articles/technical/intel-sdm.html

1.4 Counters

In our RealtimeDemo the `event_selector` for L1-miss is 0x00000000004308D1 which will be written to MSR IA32_PERFEVTSEL0. According to chapter “19.2.1.1 Architectural Performance Monitoring Version 1 Facilities” in SDM this means:

```
EN = 1              // Enable counters
OS = 1              // Count at priviledge level 0
USR = 1             // Count at priviledge level 1,2,3
Unit Mask = 0x08    // UMask=08H
Event Select = 0xD1 // EventSel=D1H
```

“Unit Mask” and “Event Select” can be found on https://perfmon-events.intel.com/ (EventSel=D1H UMask=08H). The event name is “MEM_LOAD_RETIRED.L1_MISS”. Modifying those values allows measuring any of the events supported by the CPU.

Additional information can be found in SDM vol. 3 chapter “19.2.1.1 Architectural Performance Monitoring Version 1 Facilities”
https://www.intel.com/content/www/us/en/developer/articles/technical/intel-sdm.html

List of performance monitoring events:
https://perfmon-events.intel.com/

1.5 Resources

Additional information about Resource Director Technologies (RDT):
https://www.intel.com/content/www/us/en/search.html?ws=idsa-default#q=Resource%20Director%20Technology&sort=relevancy&f:@tabfilter={[}Developers]

Additional information about Cache Allocation Technology (CAT):
https://www.intel.com/content/www/us/en/search.html?ws=idsa-default#q=Cache%20Allocation%20Technology&sort=relevancy&f:@tabfilter={[}Developers]

Depending on the processor, generation etc.
https://www.intel.com/content/www/us/en/developer/videos/time-coordinated-computing-cache-allocation.html?wapkw=Cache%20Allocation%20Technology