

Elapsed Time: 0.035s

Application execution time is too short. Metrics data may be unreliable. Consider reducing the sampling interval or increasing your application execution time.

Clockticks: 136,620,000
Instructions Retired: 118,800,000
CPI Rate: 1.150

The CPI may be too high. This could be caused by issues such as memory stalls, instruction starvation, branch misprediction or long latency instructions. Explore the other hardware-related metrics to identify what is causing high CPI.

MUX Reliability: 0.767
Retiring: 41.5% of Pipeline Slots
 Light Operations: 32.6% of Pipeline Slots
 FP Arithmetic: 0.0% of uOps
 FP x87: 0.0% of uOps
 FP Scalar: 0.0% of uOps
 FP Vector: 0.0% of uOps
 Other: 100.0% of uOps
 Heavy Operations: 8.9% of Pipeline Slots
 Microcode Sequencer: 5.2% of Pipeline Slots
 Assists: 0.0% of Pipeline Slots
Front-End Bound: 41.5% of Pipeline Slots

Issue: A significant portion of Pipeline Slots is remaining empty due to issues in the Front-End.

Tips: Make sure the code working size is not too large, the code layout does not require too many memory accesses per cycle to get enough instructions for filling four pipeline slots, or check for microcode assists.

Front-End Latency:	11.9% of Pipeline Slots
ICache Misses:	0.0% of Clockticks
ITLB Overhead:	2.0% of Clockticks
Branch Resteers:	3.6% of Clockticks
Mispredicts Resteers:	0.0% of Clockticks
Clears Resteers:	0.0% of Clockticks
Unknown Branches:	3.6% of Clockticks
DSB Switches:	0.0% of Clockticks
Length Changing Prefixes:	0.0% of Clockticks
MS Switches:	0.0% of Clockticks
Front-End Bandwidth:	29.6% of Pipeline Slots

This metric represents a fraction of slots during which CPU was stalled due to front-end bandwidth issues, such as inefficiencies in the instruction decoders or code restrictions for caching in the DSB (decoded uOps cache). In such cases, the front-end typically delivers a non-optimal amount of uOps to the back-end.

Front-End Bandwidth MITE: 35.6% of Clockticks

This metric represents a fraction of cycles during which CPU was stalled due to the MITE fetch pipeline issues, such as inefficiencies in the instruction decoders.

Front-End Bandwidth DSB: 0.0% of Clockticks
(Info) DSB Coverage: 25.0%

Issue: A significant fraction of uOps was not delivered by the DSB (known as Decoded ICache or uOp Cache). This may happen if a hot code region is too large to fit into the DSB.

Tips: Consider changing the code layout (for example, via profile-guided optimization) to help your hot regions fit into the DSB.

See the "Optimization for Decoded ICache" section in the Intel 64 and IA-32 Architectures Optimization Reference Manual.

Bad Speculation:	5.9% of Pipeline Slots
Branch Mispredict:	0.0% of Pipeline Slots
Machine Clears:	5.9% of Pipeline Slots

Back-End Bound:	11.1% of Pipeline Slots
Memory Bound:	7.9% of Pipeline Slots
L1 Bound:	23.7% of Clockticks
DTLB Overhead:	1.2% of Clockticks
Load STLH Hit:	0.0% of Clockticks
Load STLH Miss:	1.2% of Clockticks
Loads Blocked by Store Forwarding:	0.0% of Clockticks
Lock Latency:	0.0% of Clockticks
Split Loads:	0.0% of Clockticks
4K Aliasing:	0.0% of Clockticks
FB Full:	0.0% of Clockticks
L2 Bound:	0.0% of Clockticks
L3 Bound:	7.9% of Clockticks
Contested Accesses:	0.0% of Clockticks
Data Sharing:	0.0% of Clockticks
L3 Latency:	0.0% of Clockticks
SQ Full:	0.0% of Clockticks
DRAM Bound:	0.0% of Clockticks
Memory Bandwidth:	7.9% of Clockticks
Memory Latency:	11.9% of Clockticks
Store Bound:	0.0% of Clockticks
Store Latency:	11.9% of Clockticks
False Sharing:	0.0% of Clockticks
Split Stores:	0.0% of Clockticks
DTLB Store Overhead:	1.2% of Clockticks
Store STLH Hit:	0.0% of Clockticks
Store STLH Hit:	1.2% of Clockticks
Core Bound:	3.2% of Pipeline Slots
Divider:	0.0% of Clockticks
Port Utilization:	11.2% of Clockticks
Cycles of 0 Ports Utilized:	32.6% of Clockticks
Serializing Operations:	19.8% of Clockticks
Mixing Vectors:	0.0% of uOps
Cycles of 1 Port Utilized:	11.9% of Clockticks
Cycles of 2 Ports Utilized:	5.9% of Clockticks
Cycles of 3+ Ports Utilized:	11.9% of Clockticks
ALU Operation Utilization:	8.9% of Clockticks
Port 0:	5.9% of Clockticks
Port 1:	0.0% of Clockticks
Port 5:	5.9% of Clockticks
Port 6:	23.7% of Clockticks
Load Operation Utilization:	3.0% of Clockticks
Port 2:	0.0% of Clockticks
Port 3:	5.9% of Clockticks
Store Operation Utilization:	0.0% of Clockticks
Port 4:	0.0% of Clockticks
Port 7:	0.0% of Clockticks
Vector Capacity Usage (FPU):	0.0%
Average CPU Frequency:	1.533 GHz
Total Thread Count:	9

Paused Time: 0s

Effective Physical Core Utilization: 42.0% (1.679 out of 4)

The metric value is low, which may signal a poor physical CPU cores utilization caused by:

- load imbalance
- threading runtime overhead
- contended synchronization
- thread/process underutilization
- incorrect affinity that utilizes logical cores instead

of physical cores

Explore sub-metrics to estimate the efficiency of MPI and OpenMP parallelism or run the Locks and Waits analysis to identify parallel bottlenecks for other parallel runtimes.

Effective Logical Core Utilization: 31.5% (2.518 out of 8)

The metric value is low, which may signal a poor logical CPU cores utilization. Consider improving physical core utilization as the first step and then look at opportunities to utilize logical cores, which in some cases can improve processor throughput and overall performance of multi-threaded applications.

Collection and Platform Info:

Application Command Line: ./codecs/HHI-VVC/decoder/vvdecapp "-b" ".bin/HHI-VVC/randomaccess_faster.cfg/CLASS_C/RaceHorses_416x240_30_QP_32_HHI-VVC.bin"

User Name: root

Operating System: 5.4.0-72-generic DISTRIB_ID=Ubuntu
DISTRIB_RELEASE=18.04 DISTRIB_CODENAME=bionic
DISTRIB_DESCRIPTION="Ubuntu 18.04.5 LTS"

Computer Name: eimon

Result Size: 13.7 MB

Collection start time: 22:34:33 18/04/2021 UTC

Collection stop time: 22:34:33 18/04/2021 UTC

Collector Type: Event-based sampling driver

CPU:	
Name: ULX	Intel(R) Processor code named Kabylake
Frequency:	1.992 GHz
Logical CPU Count:	8
Cache Allocation Technology:	
Level 2 capability:	not detected
Level 3 capability:	not detected