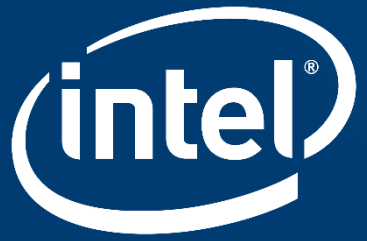


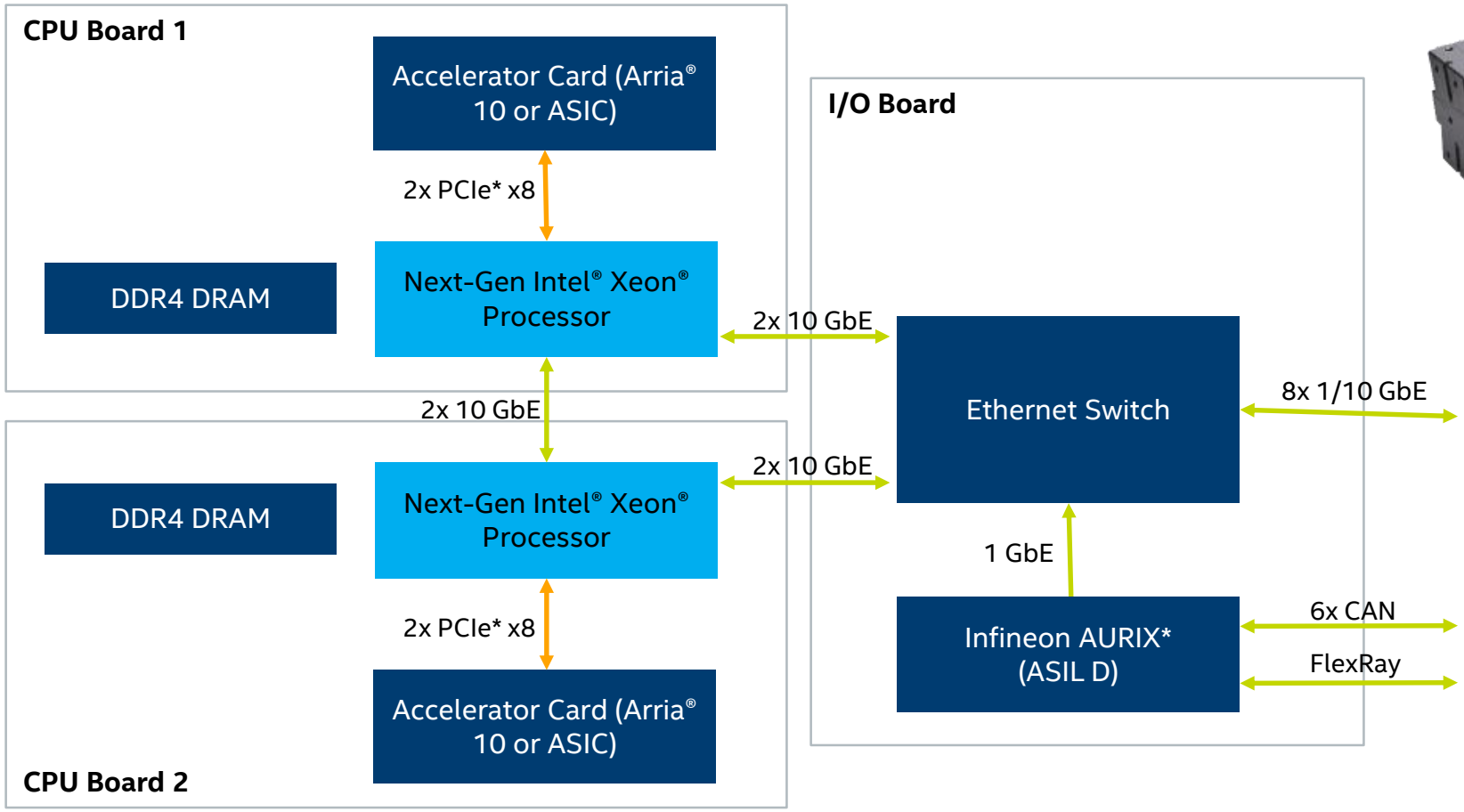
PERFORMANCE MONITORING FOR AUTONOMOUS DRIVING PLATFORMS

Vitaly Slobodskoy

vitaly.slobodskoy@intel.com

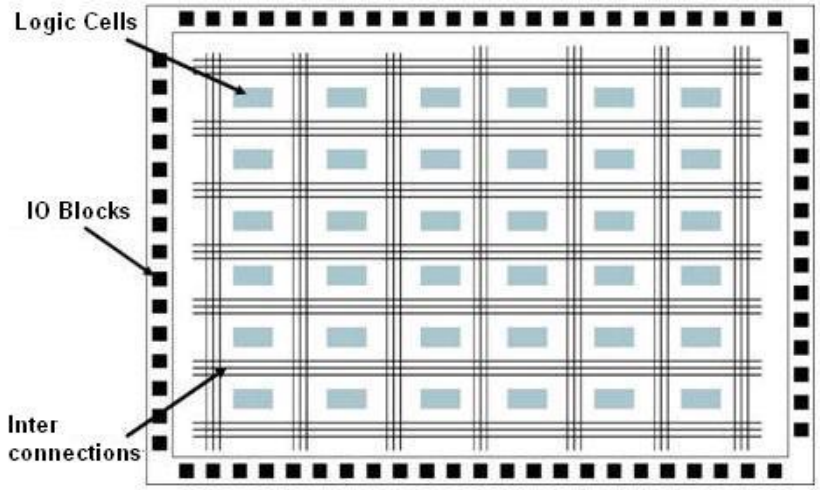
Intel® GO™ Development Platform for AD

Intel® Xeon® processor version



* www.intel.com/automotive

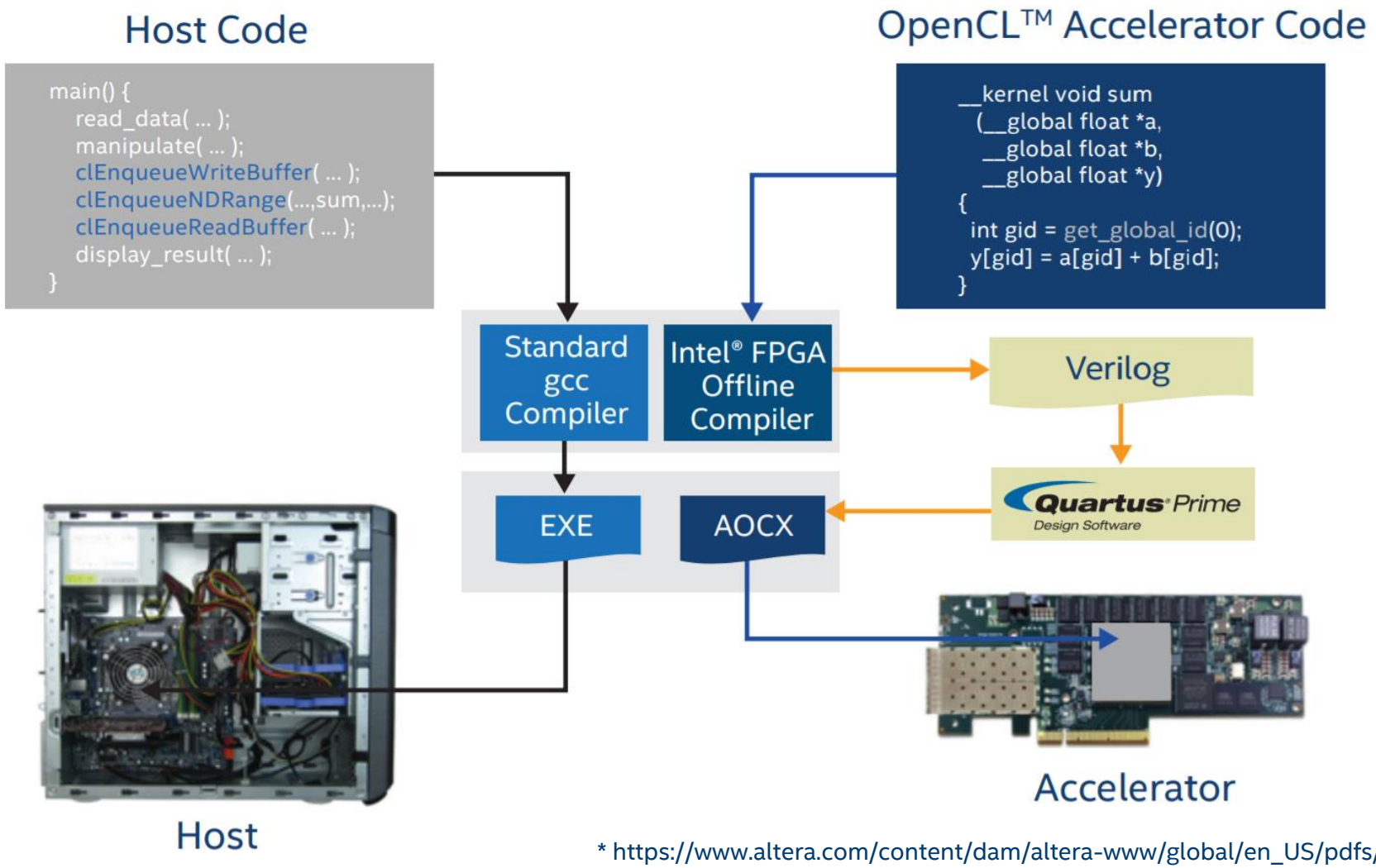
Why FPGA?



- Powerful, cost-effective, scalable design platform allows automakers and suppliers to get designs up and running without full hardware development of their own.
- FPGAs can be deployed in production systems to provide the combined benefits of hardware acceleration and field programmability.
- FPGAs allow optimization of system performance by developing hardware parallel processing engines using FPGA logic and integrating with software algorithms
- Intel® Arria® 10 FPGAs feature hard floating-point DSP blocks with speeds up to 1500 GFLOPS.

* http://fpgacenter.com/fpga/fpga_arch.php

Intel® FPGA SDK for OpenCL™ Use Model

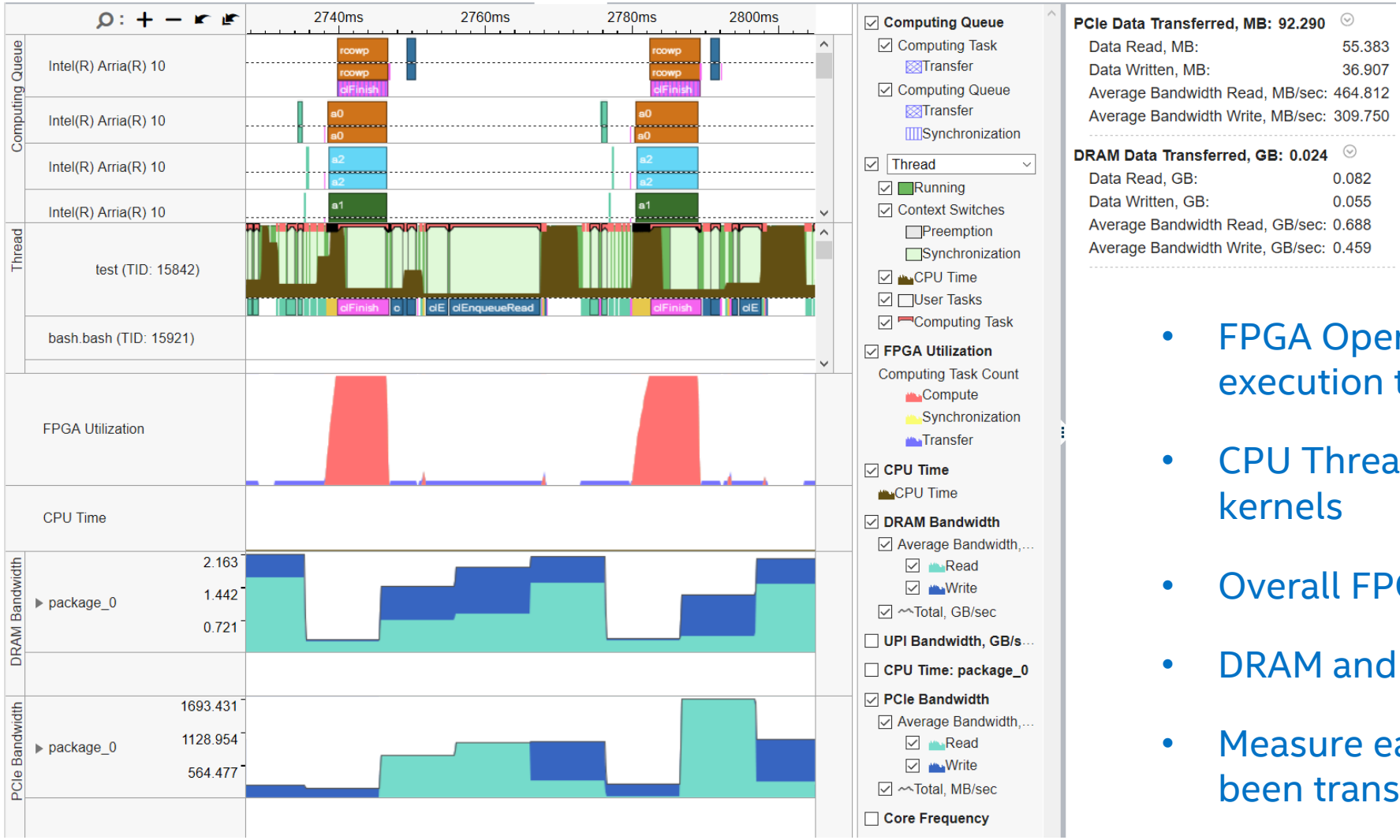


* https://www.altera.com/content/dam/altera-www/global/en_US/pdfs/literature/po/ps-opencl.pdf

CPU/FPGA performance analysis use cases

- Determine if application is CPU or FPGA bound
- Identify the places where CPU or FPGA are just waiting for each other
- Analyze the timings for CPU<->FPGA memory transfers

Intel® VTune™ Amplifier: CPU/FPGA interaction analysis



- FPGA OpenCL Queues with kernel execution timeframe
- CPU Threads originating OpenCL kernels
- Overall FPGA Utilization
- DRAM and PCIe bandwidth
- Measure easily how much data has been transferred over PCIe/DRAM

OpenCL profiling API

clCreateCommandQueue

Create a command-queue on a specific device.

```
cl_command_queue clCreateCommandQueue(cl_context context,
                                      cl_device_id device,
                                      cl_command_queue_properties properties,
                                      cl_int *errcode_ret)
```

CL_QUEUE_PROFILING_ENABLE	Enable or disable profiling of commands in the command-queue. If set, the profiling of commands is enabled. Otherwise profiling of commands is disabled. See clGetEventProfilingInfo for more information.

clEnqueueTask

Enqueues a command to execute a kernel on a device.

```
cl_int clEnqueueTask (cl_command_queue command_queue,
                      cl_kernel kernel,
                      cl_uint num_events_in_wait_list,
                      const cl_event *event_wait_list,
                      cl_event *event)
```

clGetEventProfilingInfo

Returns profiling information for the command associated with event if profiling is enabled.

```
cl_int clGetEventProfilingInfo (cl_event event,
                                cl_profiling_info param_name,
                                size_t param_value_size,
                                void *param_value,
                                size_t *param_value_size_ret)
```

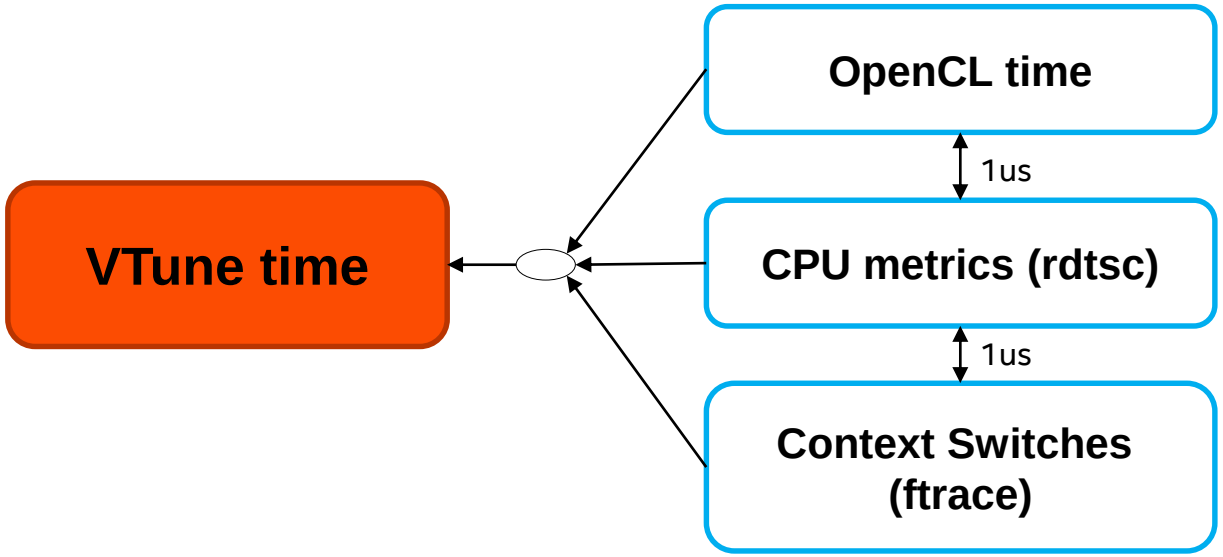
The following is a table of **clGetEventProfilingInfo** parameter queries

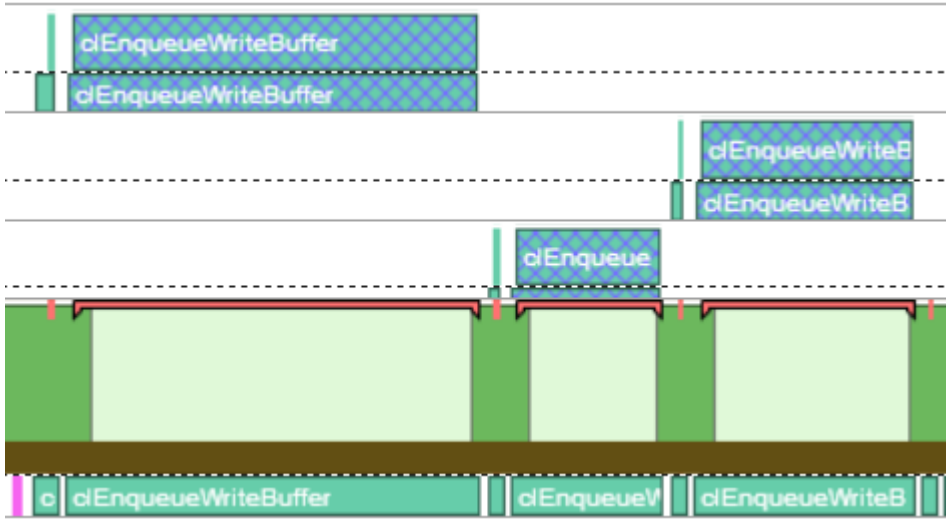
cl_profiling_info	Return Type	Info. returned in *param_value*
CL_PROFILING_COMMAND_QUEUED	cl_ulong	A 64-bit value that describes the current device time counter in nanoseconds when the command identified by *event* is enqueued in a command-queue by the host.
CL_PROFILING_COMMAND_SUBMIT	cl_ulong	A 64-bit value that describes the current device time counter in nanoseconds when the command identified by *event* that has been enqueued is submitted by the host to the device associated with the commandqueue.
CL_PROFILING_COMMAND_START	cl_ulong	A 64-bit value that describes the current device time counter in nanoseconds when the command identified by *event* starts execution on the device.
CL_PROFILING_COMMAND_END	cl_ulong	A 64-bit value that describes the current device time counter in nanoseconds when the command identified by *event* has finished execution on the device.

* https://www.khronos.org/registry/OpenCL

Time synchronization

- Consuming data from various sources we're correlating them with 1us precision (if possible)





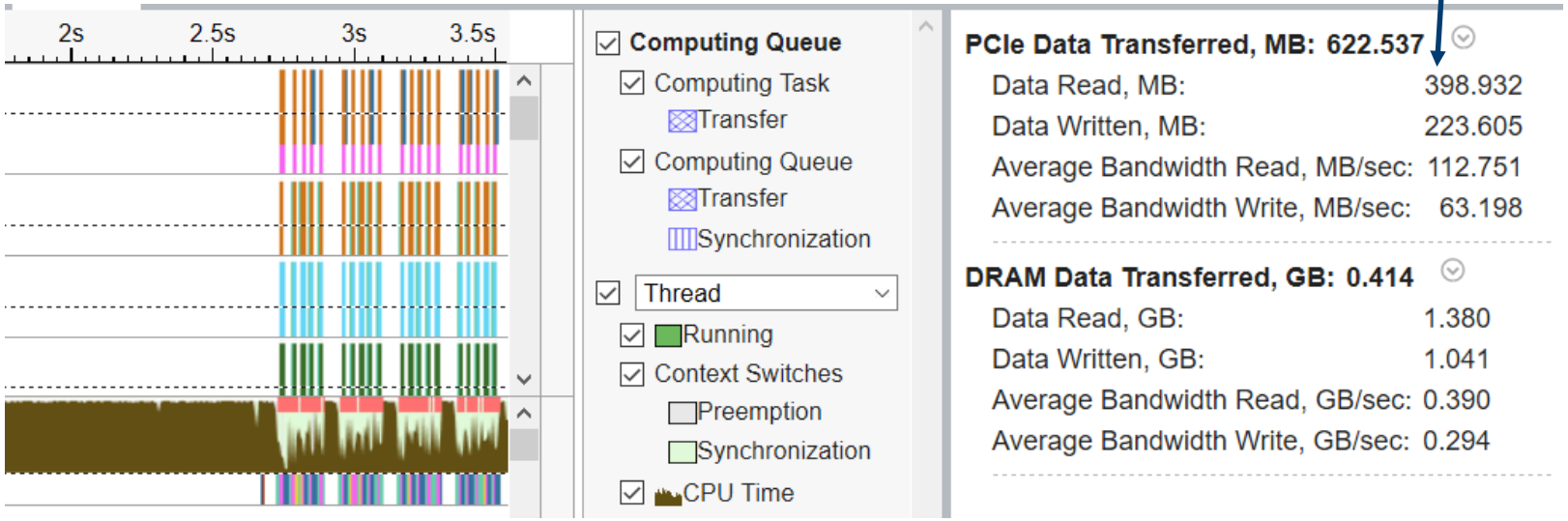
Monitoring PCIe

Grouping: Computing Task Purpose / Computing Task (FPGA) / Instance

Computing Task Purpose / Computing Task (FPGA) / Instance	Work Size		Computing Task			Data Transferred	
	Global	Local	Total Time	Average Time	Instance Count	Size ▼	Total, GB/sec
Transfer			0.275s	0.000s	616	588 MB	2.246
clEnqueueReadBuffer			0.139s	0.001s	140	380 MB	2.878
clEnqueueWriteBuffer			0.136s	0.000s	476	208 MB	1.604
Compute			3.811s	0.007s	580		0.000

OpenCL runtime provided

PMU Uncore counters



- PMU Uncore counters to measure PCIe memory transfers
- MUX free collection with PCIe FPGA slot auto-detection

Conclusion

- Use Intel® VTune™ Amplifier for heterogeneous monitoring of CPU+FPGA platforms:
  - Explore FPGA utilization with OpenCL kernels easily
  - Determine who is the bottleneck – CPU or FPGA
  - Find the places where CPU and FPGA are not executing simultaneously
  - Inspect memory transfers to FPGA
- https://software.intel.com/en-us/intel-vtune-amplifier-xe