Vortex Lite Whitepaper

Performance and Footprint Characteristics

www.adlinktech.com
2017

Table of Contents

1. Introduction

This document provides benchmark data illustrating the performance and memory footprint characteristics of Vortex Lite for a number of different platform configurations and test scenarios

2 Test Scenarios

2.1 Latency

This test measures the time it takes to deliver messages of varying sizes (measured in Bytes) between two nodes on the same Gigabit Ethernet network. The test was repeated using two different hardware configurations. For the first configuration, the Publisher and Subscriber nodes were based on typical multicore server hardware (3.5 GHz, x86 processor). The second configuration was based on hardware more typically used in resource constrained embedded IoT systems (1 GHz BeagleBone Black ARM processor boards). For comparative purposes the test were repeated for both Vortex Lite and Vortex OpenSplice.

2.2 Throughput

This test measures the amount of data (measured in Bytes per second) that can be delivered between two nodes on the same Gigabit Ethernet network for varying message sizes (measured in bytes). The test was repeated using two the different hardware configurations described previously.

2.3 Footprint

This test measures the static and dynamic memory footprint required to run a basic Vortex Lite publisher and subscriber application. For comparative purposes the test was repeated using the same applications using Vortex OpenSplice.

3. Test Environment

3.1 Test Environment 1

- Software Under Test
  - Vortex Lite v1.0
  - Vortex OpenSplice Enterprise v6.5
- 2 * Intel(R) Xeon(R) CPU E3-1270 V2 @ 3.50GHz (4 cores with hyperthreading)
  - 16 GB RAM
  - Disk: WDC WD5003ABYX-1 Rev 01.0
- Gigabit Ethernet
  - Switch: Dell PowerConnect 2816
- OS
  - 64-bit linux
  - Linux 3.8.13-rt14.20.el6rt.x86_64 #1 SMP PREEMPT RT Mon Aug 19 23:09:43 EDT 2013 x86_64 x86_64 x86_64 GNU/Linux

3.2 Test Environment 2

- Software Under Test
  - Vortex Lite v1.0

- Vortex OpenSplice Enterprise v6.5

- 2 x BeageBone Black rev C baords - Sitara ARM Cortex-A8 processor running at 1 GHz
  - 512 MB RAM
  - 4 GB of 8-bit eMMC flash memory

- Gigabit Ethernet
  - Switch: Dell PowerConnect 2816

- OS
  - 32-bit Linux
  - Ubuntu Linux v14.04 (Linux kernel 3.16)

4. Test Results

4.1 Latency

Figure 1 illustrates the latency measured in micro seconds when sending a message between a DDS Data Writer (Publisher) on one node and a DDS Data Reader (Subscriber) on another node. In order to avoid time synchronization issues, latency was measured by sending the message between the nodes and then echoing it immediately back. One-way latency was then calculated by dividing the roundtrip time by two. The test was repeated up to a maximum payload of 8 Kbytes. In this test the C API was used with the standard DDSI interoperability protocol and the message exchanges were reliable. For comparison purposes the test was repeated with both Vortex Lite and Vortex OpenSplice.

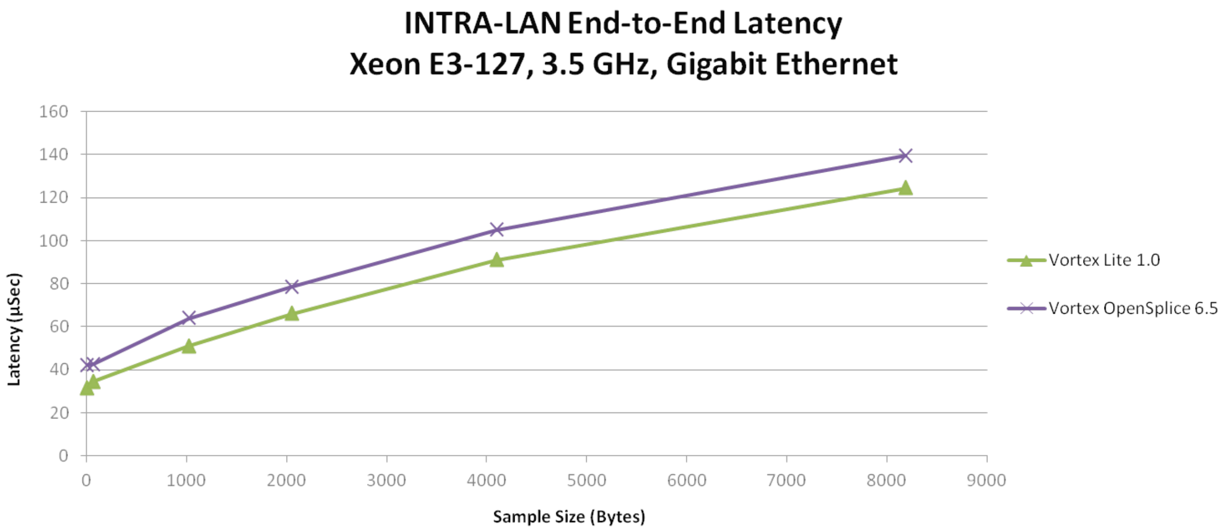

Figure 1 – Vortex Lite Latency Measurements, Multicore

The curves shows excellent latency can be achieved even as the payload size increases. The Vortex Lite results show that for small payloads < 1 Kbytes then a latency of less than 50 µSecs can be achieved. For payloads up to 5 Kbytes then the latency is still less than 100 micro seconds. Vortex Lite provides better end-to-end latency across the data range than Vortex OpenSplice.

Figure 2 shows the latency measurements when the same test was repeated on the embedded BeagleBone hardware (1 GHz ARM). Even with a much more limited processor, for small payloads, latencies measured in few hundreds of micro seconds can be achieved. As the message size increases, latency degradation is close to linear.

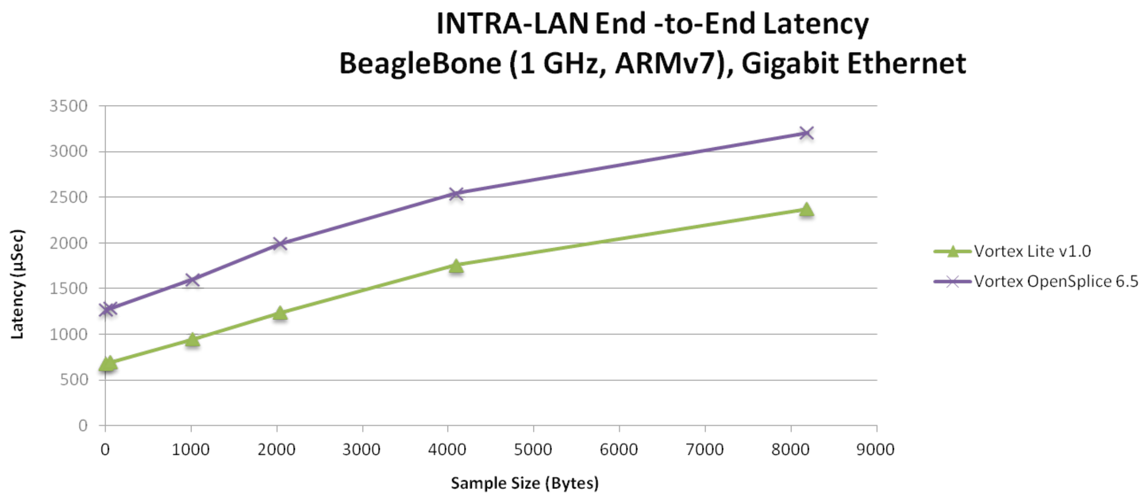

Figure 2 – Vortex Lite Latency Measurements, Embedded

4.2 Throughput

The following curves demonstrate the sustainable throughput application level measured in Mbits / second that can be achieved between a Data Writer (Publisher) on one node and a DDS Data Reader (Subscriber) on another node. The approximate available application bandwidth of the network is 950 Mbits / second taking into account the overhead of Ethernet and UDP. Using the standard DDSI protocol, the curves show that even for reasonably small packet sizes (< 2 Kbytes) throughput is limited by the network and not by Vortex Lite. A 1 Kbyte packet size corresponds to a message rate of 110,000 Messages /second.

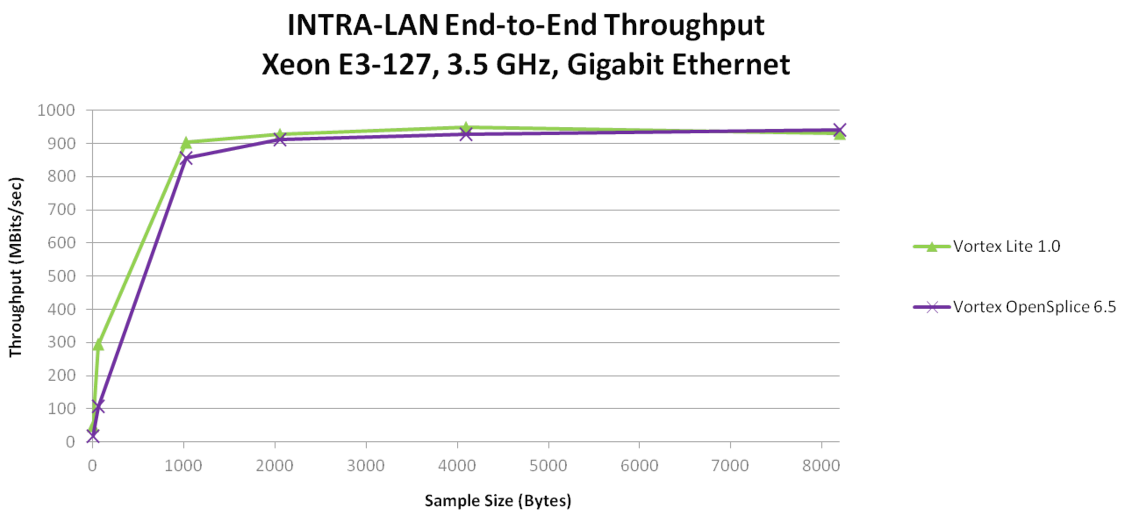

Figure 3 – Vortex Lite Throughput Measurements, Multicore Hardware

Figure 2 shows the throughput measurements that were recorded when the same test was repeated on the embedded BeagleBone hardware (1 GHz ARM). On this hardware a 1 Kbyte packet size corresponds to a Vortex Lite message rate of 3,800 Messages /second.

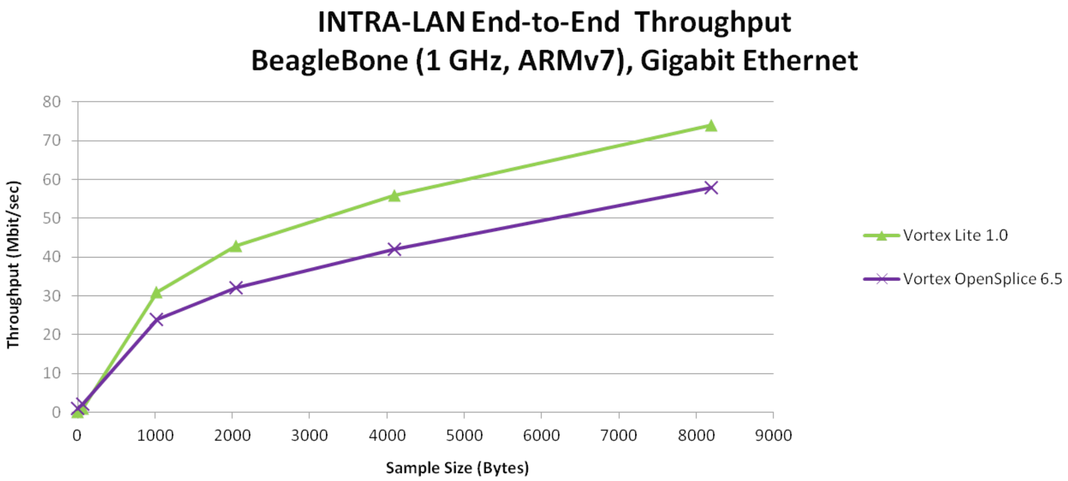

Figure 4 – Vortex Lite Throughput Measurements, Embedded Hardware

4.3 Footprint

Table 1 shows the static memory footprint for both the Publisher and Subscriber application images built with static libraries. It also shows the contribution to each image of the libdds (required by Vortex Lite applications) static library. Depending on the underlying platform the Vortex Lite static memory footprint ranges from approximately 360 Kbytes to 500 Kbytes.

	Publisher (bytes)	Subscriber (bytes)	libdds (bytes)
BeagleBone	395648	394618	368557
Intel Zeon	531320	530783	495320

Table 1 – Vortex Lite Static Footprint (Static Library)

Table 2 shows the static memory footprint for both Publisher and Subscriber application images built with shared libraries. Depending on the underlying platform, the Vortex Lite shared libdds library footprint ranges from approximately 470 Kbytes to 620 Kbytes.

	Publisher (bytes)	Subscriber (bytes)	libdds (bytes)
BeagleBone	2252	2344	470984
Intel Zeon	3038	3214	622565

Table 2 – Vortex Lite Static Footprint (Shared Library)

By reading `/proc/(pid)/status` on Linux we can determine information about the runtime memory consumption for the sample Publisher and Subscriber applications. As shown in Table 3. VmRSS (resident set size) shows the total RAM allocation for the Subscriber application. By subtracting the static image size from VmRSS value we can determine approximately the dynamic memory allocation for this simple application. For both platforms under consideration the dynamic memory consumed for a basic Publisher and Subscriber application is about 1 MB.

	VmRSS (Kbytes)
BeagleBone	1440
Intel Zeon	1520

Table 3 – Vortex Lite Runtime Memory Consumption (Subscriber) – Static Library

Table 4 shows the resident RAM memory allocation when running a shared memory build of the Subscriber application. In addition to these values there is between 470 and 620 Kbytes (depending on the platform) of memory required to hold the libdds shared libray.

	VmRSS (Kbytes)
BeagleBone	1340

Intel Zeon	2624

Table 4 – Vortex Lite Runtime Memory Consumption (Subscriber) – Shared Library

By comparison, Vortex OpenSplice shared libraries require more than 2MB of memory and the equivalent Publisher or Subscriber application process will consume around 8MB of RAM.

Vortex Lite was designed and optimised for use with embedded IoT devices and is very thread efficient. For the Publisher and Subscriber applications used for the tests described previously, Vortex Lite uses only 7 threads. By contrast the same test applications running on Vortex OpenSplice use 26 threads each. This is due to the fact that Vortex OpenSplice supports additional services not available with Vortex Lite and was designed for enterprise systems requiring massive scalability. Its more sophisticated federated architecture can support many applications running on the same node.

5. Conclusions

The results presented in this paper illustrate the exceptional low latency, high throughput and deterministic performance that can be achieved with Vortex Lite. On a range of different platforms and operating system and across a range of packet sizes, we believe that Vortex Lite has the lowest latency of any DDS implementation available on the market.

To address the memory footprint constraint requirements of embedded IoT devices, Vortex Lite's static memory overhead is measured in a few hundred kilo bytes.

Web: www.adlinktech.com

Email: ist_info@adlinktech.com

Notices