

TRADECOPE

LOWER THE LATENCY OF YOUR TRADING SYSTEM

U.S. based trading firm deploys Tradecope to increase the hit rate of their trading strategy

For successful trading on electronic exchanges today, it is not enough to come up with the smartest strategy anymore. Without the proper technology to run your strategy, you might end up with the deals that no one else wants. This is especially true if you have written

an algorithm that is looking for profitable opportunities in the market as they emerge. Finding these opportunities is not enough; you also need to be the first one to grab them.

Speed limits of commodity hardware

Our customer has written a strategy which is able to detect profitable opportunities as they appear in the market data feed and send a relevant trade order back to the exchange. They had deployed this strategy on a collocated commodity box with proprietary software and a low-latency NIC connected to a major U.S. exchange. Even though the algorithm was

reporting a lot of opportunities in the market, the profits were limited – most of the deals were gone before the corresponding order was received by the exchange. Thus, it was necessary to look for ways to reduce the latency of the system.

Deployment of Tradecope

FPGA technology is the perfect technology for further removing latency from trading systems. Tradecope is an easy-to-use solution that allows everyone to benefit from pure hardware trade processing without the need of being an FPGA expert. After deploying Tradecope, the firm has seen a significant decrease in wire-to-wire latency. **The hit rate of the algorithm doubled** compared to the previous software solution. Using Netcope's FPGA technology, the firm has turned what was an interesting-but-nobtainable market insight into a profitable business strategy. In addition, all variability in the strategy's response time disappeared – the firm could be sure that they responded to every opportunity as fast as possible, without needing to be concerned about detailed kernel tuning or minimum-code-path spin-loops or idiosyncratic specialized NICs.

The transition from software strategy to the Tradecope framework was straightforward. Netcope Technologies provided detailed training to familiarize the developers with the framework and technology. After that, the customer modified their existing codebase to use the new platform with little assistance. The decision logic written in C/C++ was automatically transformed into hardware representation using tools provided

with Tradecope. Software for configuring and monitoring the algorithm in the FPGA was written using C++ API. This work was performed by software developers without any prior knowledge of FPGA technology.

Tradecope is based on FPGA cards equipped with **the latest Xilinx's Virtex chips**. These high-performance cards support packet timestamps with nanoseconds precision. Therefore, the trading firm was able to determine wire-to-wire latencies of both solutions – with Tradecope, the reaction time **decreased from 10 μ s to a deterministic latency close to 100 ns**.

“The process of moving our trading strategy to the Tradecope platform was very smooth even though we did not have any previous experience with FPGA technology. The solution is easy-to-use, and Netcope guys were very helpful. After their initial assistance and introduction to the framework, we were able to maintain and update trading strategies ourselves. We have doubled our hit rate in a month and significantly increased our profits. The product literally paid for itself in no time,” said the firm's director of technology.

Summary

- Deployment and use of Tradecope do not require FPGA expertise. The decision logic and monitoring application can be written in C/C++ by a software developer.
- Tradecope achieves the lowest latency possible because the whole trading pipeline from incoming market data packet to outbound order message is implemented in a high-performance FPGA card.
- The hit rate of the trading algorithm was doubled as a result of reduced latency.
- Precise packet timestamps are supported, and detailed latency statistics are provided by the framework without any additional hardware.

For more information do not hesitate to contact us.