Proposal: Monitoring and Processing Stock Market Data In Real-Time Using the Cyclone V FPGA

Alexander Gazman (ag3529), Hang Guan (hg2388), Nathan Abrams (nca2123)

We are intrigued by the idea of high frequency trading after the brief informational session during class. While high frequency trading is certainly a profitable sector of the financial industry, it has not been completely embraced by the entire industry. Opponents of high frequency trading claim that it has the potential to disrupt markets through malicious trading patterns that can only be implemented due the high-speed nature of the equipment used.

Examples of these malicious trading strategies include latency arbitrage, spoofing, and market ignition. Latency arbitrage capitalizes on the fact that the amount of time it takes for a stock quote to go from a stock exchange's server to a trader's computer varies. By combining location (being near a stock exchange to reduce latency) with speed (fast hardware for processing quotes and acting on those quotes), latency arbitrage allows a trader to intelligently buy or sell on slightly out of date quotes. Spoofing involves placing orders with the intent that the order will cause other market participants to react, and then the orders are cancelled and the spoofer enters on the opposite side of the market (this was prohibited in 2010 by the Dodd-Frank act). Momentum ignition involves making trades with the intent of attracting algorithm traders to artificially inflate a stock, and selling rapidly before the stock settles back to normal. Momentum ignition was prohibited by numerous exchanges at the end of last year.

We would like to demonstrate the effectiveness of using an FPGA to implement malicious high frequency trading algorithms. Such a demonstration could be used as evidence for the need of better regulation of high frequency trading. However, we are not very knowledgeable about high frequency trading and how to implement it with an FPGA, and have a number of questions regarding this project. We noticed several past projects fit in the scope of processing 10 Gb/s stock data using daughter cards separate from the Sockit board, such as the SolarFlare AoE FPGA card. Are these additional cards available to use, or were those unique situations? Past high frequency trading projects seemed to follow the general format of using parsing stock information, updating a logbook with the recent data, and then displaying something. Is it feasible to hope to also act on the updated logbook, following some algorithm? Is it possible to get input/a more refined project idea from David Lariviere? We have a general idea of wanting to work with high frequency trading and perhaps implementing some contentious algorithm, but we are not well versed in high frequency trading and not sure what is feasible.

Our preliminary goal is to design a real-time FPGA-based stock market monitoring system that can parse the itch-protocol, translate the message byte array to a human readable format, and display it on the monitor. Information like the price, volume, etc. will be shown on the monitor in this first stage. For the testing, we plan to first use some non-real time data to verify the correctness of our system. However, we do hope that we can finally reach to the point that some real-time testing can be done by connecting our system to a real-world stock market. In the second stage, we are thinking about implement some data mining algorithms such that our users can somehow access the value of a stock. For example, we can provide the price/earnings ratio, support level, and many other useful characters to our users. Our final goal is to demonstrate some malicious trading strategies as we mentioned above. But we are not quite sure how this can be done so far.