Suitable abstract model for representing the market data feed format of different exchanges

Milan Dvořák, idvorakmilan@fit.vutbr.cz

TID 2012

Most of the securities and financial derivatives are traded electronically today. There are many financial exchanges which offer electronic trading platform, for example NYSE, Globex or NASDAQ. These exchanges provide real time data feeds containing updates on the changing status of the market. Data rates of these feeds are in a gigabit range and they require very fast processing. Furthermore, very low processing latencies are crucial, especially in case of the high frequency trading (HFT).

Our goal is to create a hardware unit for decoding and processing of the market data feeds for HFT purposes. This is a complex task due to several factors. Firstly, the solution must be sustainable for high data rates with bursts, while maintaining very low latency. Another problem is that the protocol type and channel encoding differ between each exchange. Moreover, the data format changes quite often with only a short notice. This is not very suitable for hardware implementation, since the hardware design flow is inflexible and changes in specification and implementation are costly in terms of time and effort. In the recent years a few High Level Synthesis (HLS) products were created, but significant C code rewrite and designer assistance during the synthesis is still required when the specification changes.

The solution to this problem could be finding a suitable abstract model for representing the market data feed format. This model must be able to represent different feeds on different exchanges. Each feed includes few message types, each having different meaning and template. All messages consist of several data fields. Every field needs to be processed differently, as it represents a specific piece of information. Some fields or whole sequences of fields may be repeated. The number of repetitions is usually given by the preceding field.

Besides the reading and classifying of the fields from input data feed, the model needs to cooperate with the other parts of the design as well. Processing of some fields may take longer than the others for example, so wait state may be required. Some fields may also change the meaning of the whole message. For instance, we have to ignore message with previously seen sequence number.

The model needs to be very flexible and easy to modify to reflect the dynamic and unstable nature of the data feed format. The final requirement is possibility to map the abstract model into hardware. This means transformation to the register-transfer level in VHDL or other similar language. Overall the solution should simplify the support for various exchanges and minimize the reaction time to the data format change.

Different models suitable for use in hardware will be described in the presentation. Some changes of these models will be proposed, in order to improve their characteristics for use in high frequency trading hardware. Finally, converting of these models into hardware description language will be discussed.