The Application of Network Based Synchronisation in HFT Infrastructure

Version V1.0

January 2011

Confidential

Contents

1 Introduction

As High Frequency Trading relies more and more on higher power processing at an ever decreasing time granulation for decision making, the importance of knowing precisely what time trading events have, and are, occurring has become critical to the effective performance of the trading algorithms involved.

A typical server makes use of an inexpensive crystal oscillator to maintain a sense of time. These crystals cost less than $1 in single quantities and offer only marginal timekeeping performance. They are sensitive to temperature and other factors and their frequency uncertainty is not likely to be better than 1×10^{-5} (about 1 second per day). In actual operation, most hardware clocks gain or lose about 5 to 15 seconds per day, with 10 seconds per day being typical. These crystal oscillators are the heartbeat for the server's clock.

The consequence of the typical server oscillator is an inaccurate timebase at the server level. This in turn results in a degradation of the statistical basis upon which the HFT algorithm trading decisions are made. The greater the accuracy of the knowledge of time, and the fact that this time is represented accurately over the entire decision making process, then the better, and crucially the more consistent, the performance of the trading algorithms will be.

For Post Trade analytics the replay of trading data can also be played back against the trading models in a more accurate and consistent way, directly corresponding to the events in real time allowing the user to develop performance improvements in the algorithms.

The algorithm performance requirements in HFT have resulted in greater numbers of servers and greater numbers of processing platforms within each server. This distributed processing model can only perform at its best if it can utilise the dual aspects of low latency network infrastructure *and* precisely accurate timing.

Low latency networks deliver timely trading information. Precise and accurate timing in the server receiving the trading information allows the HFT algorithms to measure latency, intervals and to log events for post algorithm replay and development.

The key aspect of accurate timing is that it provides for a **deterministic** set of latency measurements that exist directly in the trading platforms themselves in real time.

2 Distributed Synchronisation Solutions

Network time can be distributed in many ways but typically (due to scalability issues) Network based timing protocols are used. These timing protocols are used to 'steer' the server's oscillator to maintain true time.

The accuracy of recovered time at the server is affected by several factors.

- The performance of the time recovery algorithm at the back end of timing protocol
- The accuracy of timestamping
- The packet jitter in the network (Network Loading)
- The stability of the local oscillator

Any synchronisation solution must take account of all of these factors to maintain a high performance.

2.1 NTP

Traditionally NTP has been used to try to alleviate the effects of the poor oscillator performance at the server. NTP was designed for miilisecond level timing accuracy across Wide Area Networks. As such it cannot deliver nanosecond accurate timing.

Some of the key reasons why only millisecond accuracy can be obtained by NTP are :

- NTP only sends a timing packet once every 16 seconds (this is a maximum rate)
    - The large amount of time between packets means that the server time can 'wander' away from real time between packet arrivals.

- NTP uses software timestamping
    - Almost all NTP standard solutions use software timestamping of packet arrival and transmission events. Software timestamps are inherently inaccurate because they rely on the many variable factors within the computer that is running the NTP daemon. Issues such as IP stack delay, interrupts etc ensure that the timestamp cannot be accurate.

- NTP uses a standard time recovery algorithm
    - The time recovery algorithm has to take account of network jitter effects on the timing packets. It must effectively remove the network jitter to determine what the relationship of the time in the server to true time in the NTP master is. NTP defines this algorithm in the NTP standard. Whilst it is a perfectly acceptable algorithm for Wide Area time recovery to millisecond accuracy it is simply not intelligent enough to deliver time to a better accuracy than this.

NTP then, whilst prevalent in networks, is unfit for use for nanosecond accurate timing.

2.2 PTP

The greater accuracy of the Precision Time Protocol has started to gain traction over NTP in recent years. It should be noted however that PTP in itself is *not* an inherently more accurate protocol than NTP.

Simply put, PTP allows for the *ability* to compute more accurate time. It does this by addressing all the factors of that affect timing performance as already detailed in this White Paper.

- PTP can send up to 128 timing packets per second
  - The PTP protocol allows for many more timing packets to be sent per second than NTP. This greater statistical base is utilized to deliver much more precise time recovery.

- PTP uses hardware timestamping
  - PTP does not define the use of hardware timestamping, however most implementations utilize hardware timestamping. A PTP enabled NIC card is required to allow for hardware timestamping of the PTP timing packets. Any PTP implementation in the server (eg ptpd) will suffer from the same performance issues as NTP when software timestamping is used.

- PTP uses a proprietary time recovery algorithm.
  - Unlike NTP, PTP does not define a timing recovery algorithm as part of the protocol suite. Instead individual vendors implement their own timing recovery algorithm leading to differentiation of performance.
  - The Korusys PCIe card's precision PTP algorithm has been designed to pass stringent Telecom standards (G.8261) and provides best in class performance.

PTP allows for much greater timing recovery accuracy down to nanosecond level. However the detail of the PTP solution needs to be examined to ensure this level of performance is achieved.

2.3 The Korusync PTP Solution

Warning : Not all PTP solutions are the same !

As we have shown, simply adding 'PTP' to your system does not necessarily grant the user better performance than NTP.

The Korusync time recovery system utilises a PCI or PCIe card to connect to the synchronisation infrastructure. The PCIe card provides the hardware timestamping, high quality oscillator and a telecoms class timing recovery algorithm in one.

This architecture removes the processor load of the timing recovery algorithm, PTP stack and timestamping from the host processor.

The Korusync solution also provides an extremely lightweight server daemon that extracts the time from the card and trains the local server system time to within a few nanoseconds of the card time.

This has the benefit that there is no software re-tasking required or need to use proprietary API interface calls (although these are provided also).

The addition of the Korusync solution is transparent and seamless. Simply the server software runs as before except that the time available to the software processes is now accurate to a few tens of nanoseconds.

Summary Benefits of The Korusync Solution

- **System Overhead**
  - No proprietary API calls needed or re-engineering of synchronization in your system. Your existing software can remain the same and any timing calls you currently make remain the same. There is no extra processor load on the server.
- **Hardware Timestamping**
  - Nanosecond level hardware timestamping is provided to give the best possible timestamp resolution and accuracy.
- **Best In Class Time Recovery Algorithm**
  - The telecoms grade Time Recovery Algorithm provided in the Korusync solution provides the best possible synchronization irrespective of network events, network loading or packet jitter.
  - No need for expensive 'PTP transparent' switches.
- **The Last Mile**
  - Korusync provides the last mile in timing recovery by providing a supremely accurate time to the server system itself. Transferring the time from the timing card and into the server is the 'last mile' which can result in timing inaccuracy and is often ignored by competitors.
- **Stable Oscillator**
  - The Korusync system is provided with a Stratum 3 quality oscillator, frequency accurate to 50 parts per billion.

By coupling a great oscillator with a best in class timing algorithm, hardware timestamping and the ability to accurately train the server clock to PTP time Korusys have provided the most accurate and stable server timing solution to the market.

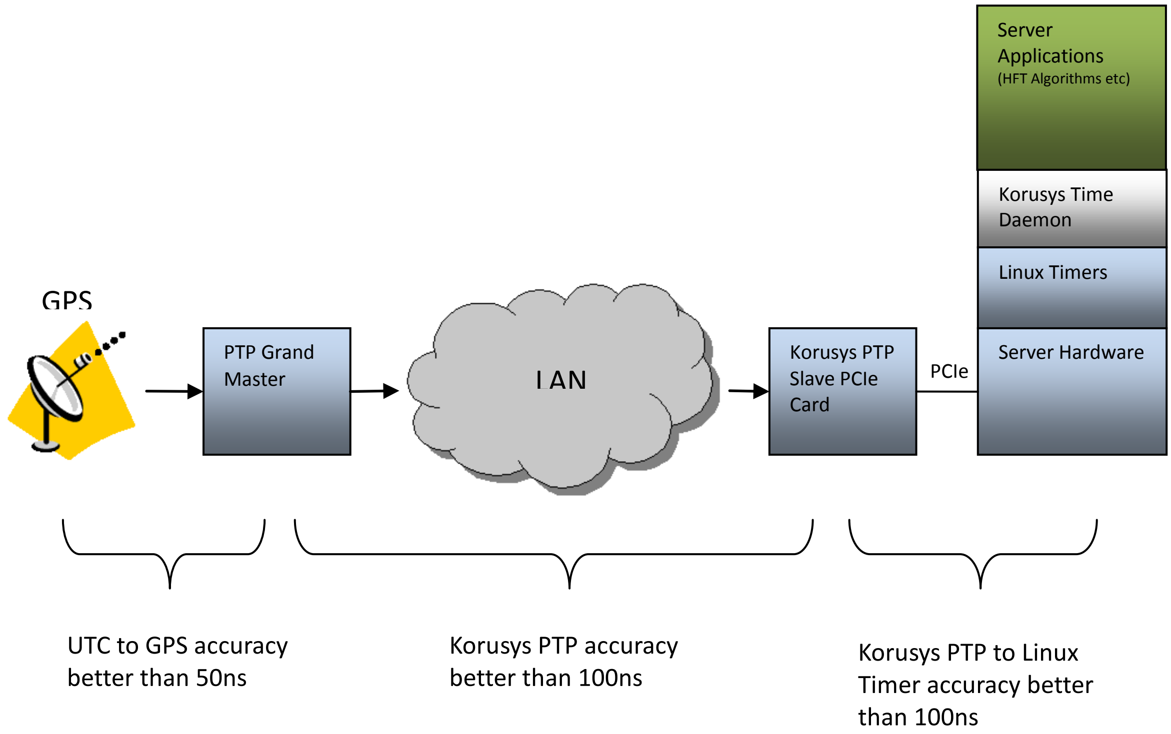

Figure 1 : Sever Timing Relationship to GPS Source

2.4 The Value of Synchronisation to Trading Systems

Precise time synchronisation between entities in a trading environment is becoming a more critical issue as latencies reduce and the numbers of trades increase. Precise, accurate time consistently applied across entities within financial system allows correlation of events and trades between disparate entities in the system.

Exchanges are now moving towards more accurate timestamping of their data feeds down to nanosecond accuracies. Because the timestamping is based on common UTC time then this allows the use of distributed accurate time within all elements of the trading infrastructure to determine true latency of information and events within the system actually at the application layer.

Current systems typically allow measurement of latency from one point of infrastructure to the next (typically by sniffing communications on the network) but with no overall common timebase to relate the various latencies between these parts.

By distributing accurate time between all entities, the use of a true latency metric which includes the real time difference from the original source of the data to the use of the data can be gained.

Whether it's to satisfy regulatory compliance, to allow a more forensic approach to back testing or to develop more effective trading algorithms utilising real time data, the necessity of truly accurate distributed time for financial IT infrastructure is clear.

2.4.1 High Frequency Trading

High Frequency Trading relies on timely execution of trading algorithms based on current market data.

Since the processing platforms and networks upon which HFT algorithms operate have become more and more powerful, the importance of the knowledge of the exact time of both the trade information and the algorithm execution has become more critical to the performance of the HFT system as a whole.

Algorithms are designed utilizing statistical information which has its basis, in part, on the relative frequency, inter arrival times, latency from the exchange and true time of receipt of various trading events or market information. By delivering a truly accurate timebase to the application level of the algorithm, 'true time' and true real time latency can be used as an input to the trading decision itself.

2.4.2 Precision Logging of Financial Trading Data

Logging of transactions is critical to the trading infrastructure, whether it's to satisfy regulatory compliance or to allow for accurate replay of data for back testing of algorithms to identify new and improved trading strategies.

Logging systems do exist at the network level but these log packets in transit and not at the server or processor interface.

Traditional implementations of logging for application events and transactions in high frequency trading have suffered from mutliple problems of

- Inaccurate time information
- Significant performance penalties whilst logging
- Lack of visibility in core-core or Infiniband coms

Beyond Latency

Currently low latency networks are implemented to allow for the most efficient HFT implementation

The actual latency of these networks is typically measured by external systems by sending out of band signalling packets between network nodes placed on the network

This gives a sense of the typical latency on the network and allows operators to know if there are network problems.

However it does not give a sense of the true latency experienced on a packet by packet basis actually at the server level.

Even with a well operating, low latency network, any individual packet could be delayed by a significant amount over the typical transit delay.

By providing a coherent sense of time at all nodes on the network to within tens of nanoseconds of UTC, individual transactions and packet arrivals can be accurately timed.

Knowledge of the precise time a trade event could be executed at the output of an HFT algorithm can be used to determine the risk of the decision on the basis of how 'stale' the data has become

Accessing accurate time at the application level in a traditional implementation involves a hefty price to be paid in terms of CPU cycles. Typically a call to get system time, further cpu cycles to wrap the system time in an application log message and, critically, calls to the IP stack to send the logging information to an external server.

In order to reduce network latencies and propagation times multi core servers have become increasingly prevalent in high frequency trading applications. Inter-process communication between the cores effectively replacing the higher latency network communication allows customers to improve their business performance and competitiveness in an application space where speed is critical.

Whilst multi-core processing and ‘networks’ provide tangible benefits in the HFT space it also comes with its share of difficulties. Prime amongst these is the lack of transparency available of application events and metrics that would otherwise have been available on the external network transactions. This has become even more of an issue with use of Infiniband and DMA based transactions. Correlating the events and transactions on hosts with nanosecond accurate time has become crucial to implementing and improving the strategic HFT algorithms implemented on the server.

Korusync solves these issues by providing a lightweight API through which simple text strings can be passed. The Korusysnc API provides for the accurate timestamping of these text strings as they hit the PCI card and generates the TCP packets to forward the strings to a remote logging server. Thus the processor cores on the server are able to offload all timestamping, OS and IP stack calls and routines to the Korusync card whilst maintaining a nanosecond accurate log of all application events within the system.

2.4.3 The solution

In assessing any solution operating in this space the customer has to address three critical requirements :

- How accurately synchronised is the server ?
- How accurately can application events be synchronised and measured ?
- How can accurate event logging be achieved with minimum CPU cycles ?

Korusnyc fulfils all these objectives by providing a solution which has zero or minimal impact on the server system and provides the industry leading synchronisation performance upon which all other applications are built.

3 Summary

Many solutions claim 'nanosecond performance'. What that means in reality can vary greatly. Nanosecond resolution timestamping, for example, is pointless unless you have timing accuracy down to the nanosecond level too.

The Korusync system delivers on all aspects of the requirements for accurate time distribution and usage in a trading environment.

- Best synchronisation performance is the bedrock upon which applications can leverage accurate knowledge of time
  - Korusys provides this with its combination of oscillator, hardware timestamping and, critically, the best time recovery algorithm available on the market.
    - Time recovery algorithm is G.8261 approves and engineered to telecom standards to overcome any extremes of network jitter or loading.
- Clean implementation
  - No need to reengineer your software , it all works out of the box due to Korusync's API providing the last mile sync between the card and the HPET timer
  - The Korusync daemon to do this has undetectable overhead on the system performance.
  - Korusync daemon accurately synchronises Linux system time to within a few nanoseconds of the card time. The card time is synchronised to within a few tens of nanoseconds of real time (over a typical financial network).
- Lightweight logging function
  - Minimal overhead ability to provide visibility of 'transactions' between cores with nanosecond accuracy
  - Output to a customer defined external server for replay, analysis or for regulatory issues.

4 About Korusys Ltd

Korusys Ltd are leading experts in packet based synchronisation techniques providing both consultancy services and synchronisation products to various market segments.

Korusys Ltd is also a trusted provider of Electronics Design Services. Focused primarily on FPGA, ASIC, and Embedded Software design and development, Korusys Ltd has earned a reputation for high quality, right first time developments for a wide variety of clients.

Please visit us at http://www.korusys.com for contact and product information or visit our reseller partners at http://www.chronos.co.uk