CRITICAL I/O WHITE PAPER

10Gb Ethernet Data Recording

StoreBox, StoreEngine, StorePak

Abstract

10GbE is increasingly being used as a sensor and processor interconnect technology in high performance embedded systems. This paper provides a brief discussion of 10GbE recording options. This is followed by an overview of the features, configuration, and usage of Critical I/O's StoreBox, StoreEngine and StorePak based data recorders which provide ultra-high performance and easily configurable multi-GB/s 10GbE TCP/UDP data recording.

10Gb Ethernet Data Recording

The use of 10Gb Ethernet is proliferating as a sensor and processor interconnect technology in high performance embedded systems. These types of systems often feature multiple channels of high bandwidth sensors, FPGA processing, DSPs, and ultra-high-performance multicore SBCs. 10GbE can provide a universal interconnect for these components. With this 10GbE usage proliferation comes an increased need for data capture and recording using 10GbE. This paper explores some issues, alternatives, and implementation details with respect to 10GbE data recording in general, as well as providing some details of Critical I/O's integrated building block approach to 10GbE recording.

The first part of this is paper discusses some different approaches than can be taken to implement a 10GbE recording solution. The following section describes Critical I/O hardware and software components that can be used to provide all or part of a 10GbE recording solution, and includes a description of the UDP Direct stream protocol and it's applicability to 10GbE recording. And the final sections provide some additional information on some of the specific features and capabilities of the turnkey Critical I/O 10GbE recording approach.

1. 10GbE Recording Approaches

The focus of this paper is 10GbE TCP/UDP data stream recording, where streams of data are produced by network data sources which are addressed to specific IP addresses and UDP/TCP ports. This paper also focuses specifically on recording performance, and perhaps more importantly, on *consistency* in recording performance. High performance sensors and processing systems produce data at very high and constant rates, often multi-GB/s. The data produced by these systems typically cannot be throttled. The sensor or processor itself controls the data rate to the recorder, and the recorder must be able to reliably handle this rate with absolutely no pauses, time-outs, or throttling.

10GbE Low Level Protocols: TCP and UDP

Although they may be augmented with higher-level protocols, there are fundamentally two low level protocols used for 10GbE recording: TCP and UDP. (Raw capture of Ethernet frames is another possibility, but this approach is fundamentally different and quite limited). Both TCP and UDP can be used effectively, and both have advantages and disadvantages.

TCP is normally considered to be a reliable protocol due to its built-in flow control and missing data detection and retransmission. But the error detection and retransmission process takes time (often 10's to 100's of msec) which real-time recording applications generally cannot afford. Thus the "reliable" aspect of TCP can provide a false sense of security in real-time applications. And any need to invoke the built-in "flow control" in reality means that the recorder potentially can't keep up with data rate and must be able temporarily to slow down the data producer, which is generally not an option in real-time systems. A summary of TCP might be to say that is it highly deterministic with respect to guaranteed data delivery, but is highly non-deterministic with respect to guaranteed timing and performance.

UDP by its nature is deemed to be an "unreliable" protocol. But the unreliability most often results from the UDP receiver (the 10GbE recorder in this discussion) not being able to keep up with the sender data rate. However, if the receiving 10GbE hardware and software, and the recording/storage hardware and software implementation is *guaranteed* to be able to keep up with the incoming UDP datagram stream, UDP then becomes a viable option that provides two key advantages. First, it has highly deterministic timing. There is no varying segmentation, varying send windows, no Nagle or ACK delays, retransmission delays, etc. such as can be the case with TCP.

And second, it is very simple to implement from a sender point of view, with no complex send side state machines such as is the case with TCP. This means that a 10GbE UDP sender can be simply implemented in hardware such as an FPGA, without the need for extensive accompanying processors and software to manage the interface. Critical I/O 10GbE hardware supports an enhanced UDP Direct stream protocol (see sidebar) which improves UDP performance and reliability in recording applications.

Recorder Architecture Options

10GbE recording architecture alternatives generally fall into one of two categories. The first category might be termed *file access recording*. These use file access methods like NFS, FTP, or CIFS/SMB running as a higher level protocol on top of TCP to perform file writes to a standard network file server. The network file server can be a standalone box or PC, or it may be one or more VPX/VXS/VME blades (such as an SBC). File access recording requires that the sender (the file access client in this case) open one or more files on the server, the perform writes (or reads) to the file to accomplish the recording function. File access recording has the advantage of being extremely simple to implement on the recorder side, as it consists of completely off the shelf standard hardware and software. It has the disadvantage of being very limited in performance, and suffering from poor recording rate determinism and consistency. A file access recorder may be as simple as a Linux SBC with one or more on-board SSDs, or a rack mount embedded PC with multiple RAID's.

The other general recorder category might be termed *stream recording*. Stream recording means that there is no in-band higher level protocol (aka NFS), and that the recorder simply records all data that is received on a TCP/UDP socket. Stream recording may leverage a "out of band" control interface which allows the recording controls to be provided by a network node that is distinct from the recording data source. Two possible implementation approaches for stream recording include: 1) leveraging commodity SBCs and SSDs supplemented with in-house development of recording software, and 2) using optimized purpose-built recording hardware and software. These approaches provide varying levels of performance and performance consistency.

Several possible 10GbE recording architectures are illustrated below. Approach 1 uses a *file level* recording model, leveraging a

10GbE UDP Direct Transfer Protocol

Critical I/O's UDP Direct transfer protocol and XGE NIC hardware provide a highly efficient method of moving block UDP data over standard 10GbE networks, using completely standard UDP as the on-the-wire protocol. With a typical i7 CPU hosting the CIO UDP Direct driver, full 10GbE line rate sends and receives can be achieved using less a 5% loading of one CPU core. The UDP Direct stream mode of operation can be used concurrently with general purpose 10GbE network traffic using the normal network stack. The XGE NIC hardware, firmware, and driver software support simultaneous usage for UDP direct stream transfers and standard networking.

As implied by the name, at UDP Direct mode applies to UDP traffic streams only. A "UDP stream" is defined as a flow of data (a series of UDP datagrams) transferred between two UDP "endpoints", where each endpoint is defined by IP address and UDP port. For example, a connection between [192.168.5.1: 1005] and [192.168.5:1025] would be a stream. (IP addresses and UDP ports can also be wildcards).

For UDP Direct stream sends, the user defines a buffer anywhere in PCIe address space of arbitrary size. A single call to the CIO driver will result in the driver and NIC firmware initiating a send of the full buffer, with the NIC automatically breaking the buffer up into as many identically sized UDP datagrams as are needed to send the full user buffer. A benefit of UDP stream sends is that they can easily be implemented in FPGA based sensor nodes.

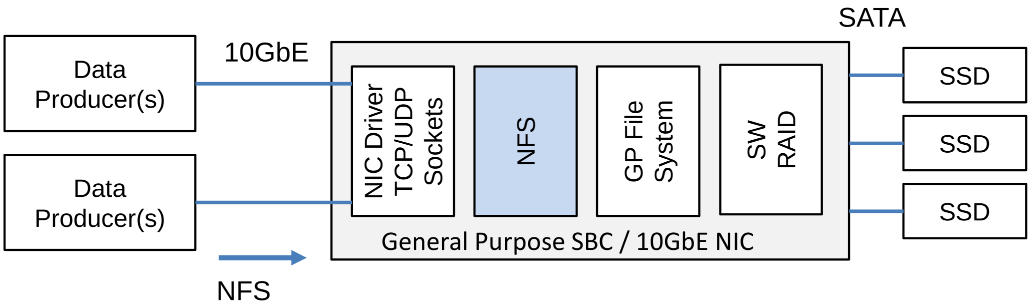
Note that while the other side of the interface can also be using the stream mode, it does not have to. It can also just send (or receive) data via standard socket calls, provided the datagrams are the correct size.

network file protocol such as NFS as the "on the wire" method of moving data. While conceptually straightforward, NFS is a fairly "heavy" protocol that places a significant burden on both the data producer as well as the recording subsystems.

Approaches 2 and 3 assume the use of a simple socket level stream protocol, where the recorder subsystem simply records all data received on one or more TCP/UDP stream sockets. These approaches are much lighter weight, and can provide higher performance and better determinism. And implementation of a "socket" recording model can also be much more straightforward for hardware based (e.g. FPGA) data producers.

Approach 1: General Purpose Linux SBC and NFS – Low and Inconsistent Performance

This approach is completely "off the shelf" as it leverages NFS/NAS functionality built into any Linux SBC. This approach is viable for lower performance recording applications, where the data "producer" is also an SBC that can host a NFS client. Hosting a NFS client can be problematic when the producer is a hardware device such as a sensor or FPGA.



Advantages

- Completely "off the shelf" - minimal development

Disadvantages

- Limited recording rates
- Poor recording determinism
- Storage may be limited by number of SATA connections
- Data producer must implement a full NFS client.

Approach 2: General Purpose Linux SBC and SSDs with Custom Stream Recording Software

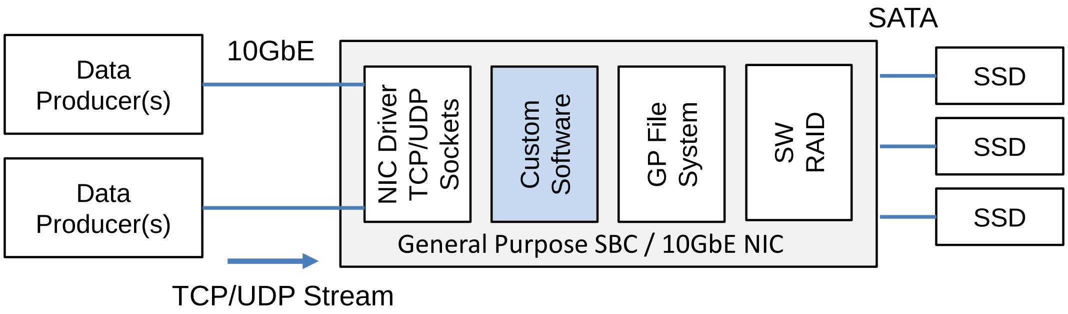
This approach can also leverage off-the-shelf Linux SBCs connected to commodity SSDs, but leverages a stream recording protocol rather than a file access protocol (e.g. NFS).   In this example, user create customer software is used to manage the stream recording in conjunction with a standard Linux file system.

Advantages

- Simple data producer TCP/UDP socket send implementation

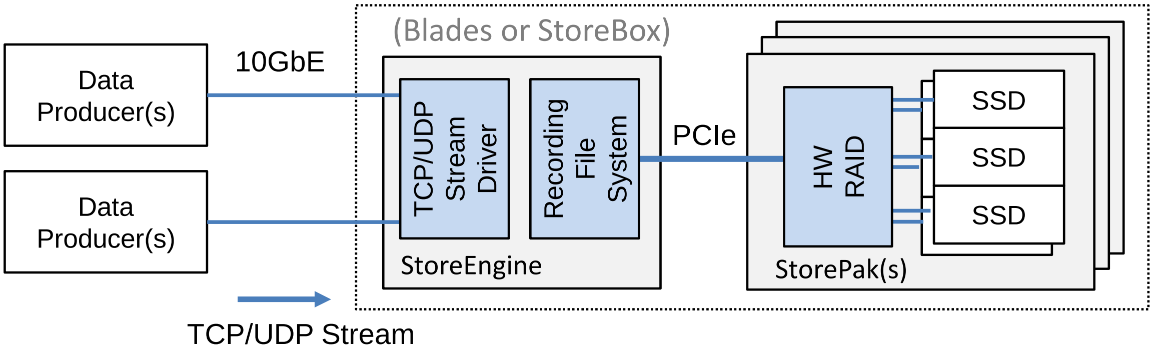
Disadvantages

- Recording rate typically limited by SBC SATA connections
- Storage may be limited by number of SBC SATA connections
- GP File system and SW RAID limit performance and determinism
- Must develop custom stream recording software.



Approach 3: A Turnkey Optimized 10GbE Recorder

This approach leverages optimized recording hardware and software to provide the best performance and performance consistency. This example leverages Critical I/O's StoreEngine and StorePak storage blades (possibly in a fully integrated StoreBox chassis), along with highly optimized recording software. The recording "software stack" is fully optimized all the way from the 10GbE interface hardware to the individual SSD read/write operations.



Advantages

- Simple data producer TCP/UDP socket send implementation
- Highly scalable, no limit to number of SSDs
- Higher performance and much better determinism
- Recorder management can be from different node from data producers
- Minimal in-house development effort and minimal risk

2. Critical I/O Recording Hardware and Software Components.

Critical I/O's recorder building blocks can effectively be used to support all of the approaches described above. However, Critical I/O's integrated 10GbE recording architecture is based on the third approach described, and is designed to provide very high, scalable, and deterministic recording performance.

Critical I/O provides fully pre-configured chassis level 10GbE recording systems including the StoreBox and StoreRack products. Critical I/O also provides blade level recorder building blocks, including the StoreEngine and StorePak blade level products.

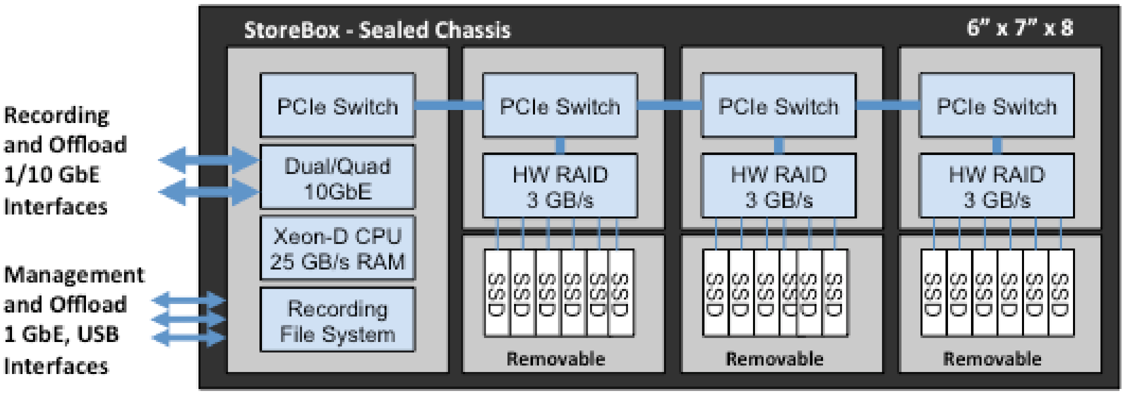
The StoreBox and StoreRack integrated 10GbE recorder products internally use the StoreEngine and StorePak blades and StoreEngine recording software. Thus a StoreBox 10GbE recorder is 100% functionally equivalent to a recorder implemented using StoreEngine and StorePak blade level building blocks installed in a customer supplied chassis.

StoreBox – Compact, Rugged 10GbE Recorder

The StoreBox Recorder consists of an ultra-compact conduction cooled chassis with dual or quad 10Gb Ethernet optical or copper recording interfaces, as well as dual 1GbE control/management interfaces. Critical I/O's StoreEngine and StorePak boards and flexible recording software functionality work together to provide a dual channel full line rate 10GbE data recording capability. The recorder provides a capability to record multiple streams of data directly from multiple 10GbE connected data sources at aggregate rates of up to 2.5 GB/s.

Multiple configurations are available which provide up to 36TB of SSD storage, with dual/quad port, copper/optical 10GbE interface options. Recorded data may be played back to any 10GbE connected "data destinations", or alternatively, recorded data may be accessed via 1 or 10 Gb Ethernet connections to the recorder using standard file access protocols such as NFS.

StoreBox-CC Provides up to 36 TB of Hot-Swappable SSD Storage - 6"x7"x8"



StoreBox-CC Internal Architecture

StoreEngine and StorePak – Recorder Hardware Building Blocks

StoreEngine and StorePak are flexible storage building blocks that can be used to implement a wide range of data storage systems. StoreEngine is an ultra-high performance VPX storage controller blade that hosts up to 12 TB of non-removable on-board SSD storage. StoreEngine simultaneously provides high performance recording functionality; serves block data (like a disk drive or RAID system); and provides NAS file sharing (like a NFS/CIFS file server).

StorePak is a PCIe connected VPX storage expansion blade that can host up to 24 TB (6u) or 12 TB (3u) of easily hot swappable SSD storage per blade. Together, StoreEngine and StorePak provide unmatched storage capability, ultra high performance and high capacity within a small size, weight, and power (SWaP) footprint.

StoreEngine and StorePak blades are available in 3u and 6u VPX, both air-cooled and conduction-cooled.

Both StoreEngine and StorePak provide rich PCIe connectivity, with up to eight x4 PCIe backplane ports per VPX board. These ports are used for connections to the user's data sources as well as for interconnections between StoreEngines and StorePaks. Both boards feature PCIe switches which are fully partitionable, and support NT bridging, providing flexible system architecture options.

Critical I/O 10GbE StoreEngine 10GbE Recording Software

The StoreEngine recording platform provides a simplified architecture through seamless aggregation of multiple StoreEngine and StorePak storage blades that maintains a single entity operational and management view.

Core to the recorder is the Critical I/O recording software, which runs directly on the StoreEngine blades and provides turnkey data recording operation. The software coordinates the recording of data across multiple StorePak blades, and also coordinates unified playback of recorded data. Full functionality to control both the recording of data and the playback of data from multiple blades is built into the manager software.

Recorder Streams

Central to the configuration and operation of the Critical I/O 10Gb recorder is the notion of data *streams*, which are bidirectional "socket connections" (a UDP or TCP flow of data) between an external 10GbE data producer and a StorePak/StoreEngine LUN (a LUN is a logical recording storage container on a StorePak or StoreEngine blade, analogous to a RAID of multiple SSDs). For UDP/TCP recording, a stream is fully defined by the IP addresses and UDP/TCP port numbers, and the storage LUN.

A single LUN can service multiple streams. For example, two or more different 10GbE (or PCIe or Fibre Channel) devices may read or write data for the same LUN using two or more different streams. A *channel* is defined as a logical flow of data within a stream; a single stream can support multiple channels of data.

Recording File System

All data is recorded to SSD storage through a recording file system. The file system is designed to ensure high performance and highly deterministic recording performance. After LUNs are created and streams are defined, data is recorded by simply issuing "start" and "stop" commands to the recorder for the desired streams/channels. As data blocks are received on a stream/channel, the recording file system defines where in SSD storage the data blocks are placed. All recorded data blocks are time-stamped as they are received by the recorder.

Recurring Mode

The recording mode for a stream may be set to *recurring* mode. Recurring mode will cause files to be automatically and continuously created that correspond to the size and/or time limits as defined by the recording controls. For example, if a 1GB size limit is defined, then recorded data will be automatically divided into a series of 1GB files.

Continuous Mode

The recording mode for a LUN may also be set to *continuous* mode. Continuous mode allows data to be continuously recorded in a wrap-around fashion where newer data continuously overwrites older data.

Hot Swap Modes

StorePak modules are easily removed, and can be hot-swapped while the recorder is actively recording, allowing continuous recording. There are several unique recorder configuration features that support hot-swap and continuous recording functionality.

Round Robin Continuous Recording -- The round robin spare LUN feature is a mechanism which allows continued recording in the event that a LUN becomes full or is taken offline due to a StorePak hot-swap event being triggered. When either of these two events occurs during an active recording, the recorder software will transition a stream target LUN to the next available spare LUN, and continue recording. The StorePak which was taken offline may then be removed and replaced with a fresh StorePak. Multiple spare LUNs may be assigned, which are used in a round-robin manner, with the order matching the order that the spare LUNs were assigned to the primary LUN.

Hot Swap Auto Configuration --The StorePak hot swap auto configuration is a feature which will guarantee that a StorePak being added to the system during the StorePak hot swap procedure will match the configuration of the StorePak being removed from the system. As a result this will wipe any data on the newly installed StorePak, leaving it in freshly reset state after the hot swap event.

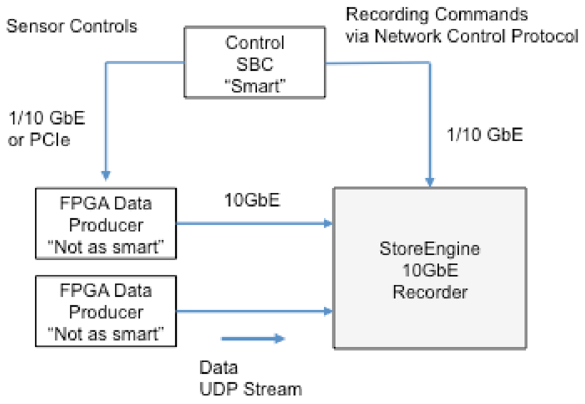
Hot Swap Configuration Cloning -- The StorePak cloning feature allows the user to replicate a StorePak's configuration onto another StorePak. The boot time auto configuration feature allows the user to save a StorePak's configuration which the system will use at boot time to verify and if need configure the currently installed StorePak. The hot swap auto configuration feature will automatically configure the newly installed StorePak during the StorePak hot swap process.

Recording Controls

Recording operations may be configured, managed, and controlled via:

- Web based control using the built-in recorder web based management pages
- Network connection, using the Recorder Network Control Protocol
- PCIe connection, using the Recorder Driver hosted on a user's system controller SBC

The control capabilities that are available through the three control interfaces are similar. As shown in the figure below, the source of recorder controls can be separate from the source(s) of the data to be recorded.



Stream Recording Allows for Separation of Recording Control from Data Sources

Access to Recorded Data

Recorder Data Forwarding

Data Forwarding allows data that is being recorded to also be forwarded to another 10Gb Ethernet network simultaneously with being recorded. Forwarding builds on the recorder concept of streams. Any recorder stream can be configured to be forwarded to another stream. Forwarded data can be sent on the 10GbE network as a sequence of normal TCP or UDP datagrams. Thus the data can be received by any 10GbE receiver using normal sockets that are bound to the appropriate IP addresses and TCP/UDP ports.

Recorder Data Playback

Playback capability is provided through a playback manager function provided within the recording software. Data playback may be via a PCIe connection, Fibre Channel or 1/10 Gb Ethernet. For playback of recorded files, the recorder responds to requests from a playback consumer, which is the device that requires access to the recorded data. The playback consumer may be a processor board, an external PC, or any other computer that can connect to the data recorder via PCIe, Ethernet, or Fibre Channel.

Network File Access (NFS, FTP, CIFS/SMB)

Recorded data may be network accessed via a 1/10 Gb Ethernet connection using NFS (or FTP). Each Recorder LUN has an associated instance of a network exportable file system. These file system instances can be individually NFS exported (or CIFS/SMB or FTP) through StoreEngine. Each LUN (NFS export) then appears to the user (NFS client) as a separate NFS mount.

StorePak Offload Station

The StorePak offload station allows StorePak removable SSD modules to be connected directly to a desktop PC or server for offload of data from the StorePak. The offload station includes a PCIe controller board, a StorePak offload cable, and optional Linux data offload software. The Offload Controller is a PCIe plug-in card. The card hosts a SATA/RAID controller, and plugs into a standard PCIe x8 expansion slot. The offload cable plugs into the controller card, and connects directly to the StorePak removable SSD module

The Critical I/O Offload Software, installed on the PC/Server, is used to view and offload the recorded files that stored on the StorePak. After offload of data, a secure erase of the StorePak storage may be initiated to free SSD space and restore optimal performance, and prepare the StorePak for reuse.

Remove
StorePak
StorePak Offload Cable
6x SATA + Power

Optional Data Offload Method Using StorePak Offload Station

Summary

10 GbE provides a very high bandwidth data transport between sensors, processors, and storage systems. Critical I/O's integrated 10GbE recording systems leverage StoreEngine and StorePak storage blades to provide a highly flexible and capable 10GbE data recording system. The same StoreEngine and StorePak blades can also be leveraged as building blocks to provide equivalent 10GbE recording functionality hosted in a customer provided chassis.