Demonstration of COTS Hardware for Capture, Playback and Processing of SMPTE ST 2022-6 Media Streams

Thomas Edwards
FOX Network Engineering & Operations
Los Angeles, CA
thomas.edwards@fox.com

***Abstract*—The broadcast industry is investigating moving from a point-to-point serial digital interface for uncompressed live media streams in the broadcast plant to packetized connectivity. Doing so could provide enhanced business flexibility and agility, as well as potentially providing cost advantages through the use of COTS hardware instead of broadcast-specific interfaces such as SMPTE ST 292-1. Unfortunately, most COTS operating systems and NIC drivers are not optimal for moving multiple uncompressed isochronous media streams such as SMPTE ST 2022-6 at bit rates of 1.5 Gbps per stream. This poster regards a proof of concept where a COTS hardware server is teamed up with special software to more efficiently transmit and receive packets. This includes bypassing the typical TCP/IP stack, using DMA-enabled NIC drivers, and using packet playback software that accurately replicates the cadence of captured packets. The potential for capture, clean editing, graphics insertion, and playback of uncompressed video files using SMPTE ST 2022-6 will be demonstrated.**

Keywords—SMPTE 2022-6, uncompressed, professional media networking

I. INTRODUCTION

The broadcast industry is considering a move to converge professional quality audio, video, data, and production communications to a common networked infrastructure. The business drivers behind this revolution include: enhancing the agility and flexibility of the broadcast plant; reducing the amount of cabling through aggregating multiple signals onto Ethernet connections; expanding the use of common off-the-shelf (COTS) PC and networking hardware that have economies of scale beyond just the broadcast industry; making the broadcast plant more easily monitorable; distributing video processing in an efficient fashion over private or possibly public cloud infrastructures; and providing an efficient multi-resolution unified fabric for SD, HD, and UHDTV video signals in both compressed and uncompressed forms.

The live media signals used in professional broadcast plants are unlike most types of signals transported over typical packetized networks. They typically cannot afford any packet loss. Data rates for video can be very high, for example typically around 1.5 Gbps for uncompressed HD video. It is expected that the media signals will be multicast so that one source can be sent to multiple destinations simultaneously. Packet delay variation must be small – if the data for a video frame is not available by the next frame period, it is useless.

One of the major questions is whether common off-the-shelf (COTS) hardware such as standard PC servers and Ethernet switches can meet the requirements of packetized professional media. This work examines the capture, clean editing, graphics insertion, and playback of uncompressed HD video using the SMPTE ST 2022-6 packetized video standard.

II. SMPTE ST 2022-6

The SMPTE ST 2022-6 standard [1] provides for encapsulation of the payloads of a variety of SMPTE serial digital video standards. The standard carries 1376 octets of video payload into each RTP datagram. In particular, this work looks at the carriage of 720p/59.94 video in SMPTE ST 292-1 ("HD-SDI"). Conversion between HD-SDI uncompressed video streams and SMPTE ST 2022-6 RTP data streams was performed by a Nevion VS902 card.

III. PF_RING DNA

A challenge of using COTS hardware with SMPTE ST 2022-6 streams is the difficulty of capture and playback of such high speed streams. One solution is the use of a TCP/IP "stack bypass" system such PF_RING from ntop.org [2]. This provides a new socket service in the form of a Linux kernel module that more directly communicates with the NIC with fewer of the "bells and whistles" of the standard Linux TCP/IP stack (such as the crafting of packet headers, routing, or error detection). By reducing the amount of processing done by the kernel, PF_RING dramatically speeds up packet processing.

For further speed enhancement, the "DNA" DMA-enabled NIC driver for Intel 10 GbE NICs was used as well. This allows PF_RING to communicate with the NIC via DMA transfers performed by the NIC NPU (network processing unit), rather than involving the CPU in mmap.

Two PF_RING DNA enabled applications were used for packet capture and playback. “n2disk” provides high speed packet capture to standard PCAP formatted files. “disk2n” provides playback from PCAP format files at the proper packet cadence as specified in the PCAP packet timestamps. It does so by using a "pulse" thread calling clock_gettime() on CLOCK_REALTIME (the system-wide real-time clock).

The server used for packet capture and playback was a standard Supermicro 1RU chassis with X9SPU-F motherboard, one Intel 3.3 Ghz Xeon CPU E3-1230 V2, 8GB RAM, a standard Intel 82599EB-based dual 10 GbE SFP+ NIC, with four 10K RPM SATA hard drives. The OS was Ubuntu 12.04.2 LTS (GNU/Linux 3.2.0-44-generic x86_64), along with the PF_RING kernel module & DNA NIC driver. This system along with the Nevion VS902 card and a video display would be made available for demonstrating playback of captured and processed video files.

IV. Packet Capture & Playback

To test the PF_RING DNA enabled system, a standard FOX Lab video test clip was played back from a Video Clarity system over HD-SDI to the Nevion VS902 card. The output 10 GbE of the VS902 encapsulated the HD-SDI in SMPTE 2022-6 streams. A distribution amplifier provided a copy of the HD-SDI to the VS902 card as well so that two identical HD-SDI streams were output by the VS902 card, a total bandwidth of just over 3 Gbps.

12584 frames of 720p/59.94 video (~226 seconds) of each of the two streams was recorded onto the server using n2disk. No packets were dropped in the capture. Disk2n was then used to play back the streams over 10 GbE to the VS902 card, which converted the streams to HD-SDI for recording and analysis on the Video Clarity system. This system compared every active video pixel of the original to the playback, and every pixel was perfectly played back, bit for bit.

V. Clean Editing of 2022-6 Files

SMPTE RP 168 [3] defines a switching point in the video raster such that the effects of any signal discontinuity in the processing chain is minimized. For 720p/59.94, the RP 168 switching line is line 7, and the switching area is 455-780 pixels after the end of SAV (824-1149 pixels from the beginning of EAV on that line). As mentioned in [4], one can simply count a number of RTP datagrams into the frame to find a packet boundary that lies in the switch point area, as shown in Figure 1.

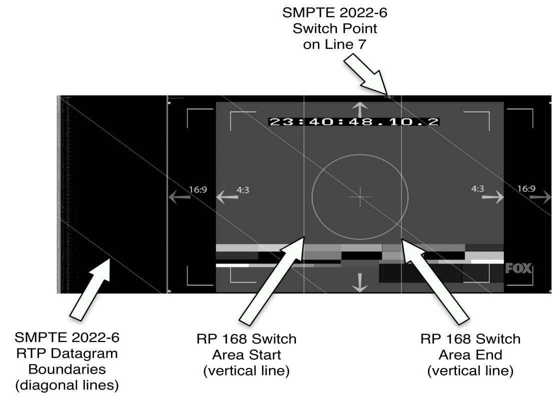

Fig. 1. Relationship of SMPTE RP 168 to SMPTE ST 2022-6 Packet Boundaries

In this work, two different file captures were performed with n2disk, and the resulting PCAP files were edited together by switching at the SMPTE RP 168 appropriate RTP packet boundaries. Disk2n is used to play back the edited file sequence.

VI. Logo insertion of 2022-6 files

PCAP files containing SMPTE 2022-6 captures also had logos inserted into them, and playback is to be demonstated. Special care must be taken to calculate the HD-SDI Cyclic Redundancy Check (CRC) on lines where the active video is changed. This is not a problem for video switching as per RP 168, as the active video during the vertical blanking interval should always be black.

References

[1] SMPTE ST 2022-6:2012, “Transport of High Bit Rate Media Signals over IP Networks (HBRMT)".

[2] “PF_RING: High-speed packet capture, filtering and analysis.” http://www.ntop.org/products/pf_ring/.

[3] SMPTE RP 168:2009, “Definition of Vertical Interval Switching Point for Synchronous Video Switching". K. Elissa, “Title of paper if known,” unpublished.

[4] M. Laabs, “SDI Over IP – Seamless Signal Switching in SMPTE 2022-6 and a Novel Multicast Routing Concept,” EBU Technical Review, 2012 Q4.