Protocol Enhancements In P1394a And Why They Are Important For 1394 Devices In A PC Environment

Rev 1.0

A White Paper

By

Bill Pearson

Intel Corporation

January 15, 1998

Introduction

P1394a is an update to the IEEE standard 1394-1995 specification. It provides some fixes to old problems as well as some improvements and optimizations. P1394a provides a solid starting point for 1394 development. Intel encourages the use of P1394a as the standard to be used for 1394 development.

P1394a and IEEE standard 1394-1995 both specify S100, S200 and S400 speeds. The performance advantages of P1394a derive from protocol enhancements such as fly-by arbitration , ack-accelerated arbitration, and fairness optimization. The sum of these improvements is actually greater than the contribution of each alone. This results in substantial performance improvements for P1394a devices, particularly for storage.

In addition to the specific protocol enhancements detailed below, P1394a also improved the standard PHY/Link interface and added PHY register read and write capability. The standard PHY/Link interface improves interoperability among PHY and Link chips. The register read and write capabilities are used in conjunction with PHY pinging to get delay information for setting timings and are also used to initiate suspend, resume, and disable operations.

This white paper begins by providing a brief summary of the protocol enhancements in P1394a. It then goes on to explain some of the reasons why P1394a is important for devices in a PC environment.

Connection Hysteresis (debouncing)

When a connector is inserted or removed from a socket, a clean connection is not immediately made. What this means is that electrical contact is made and broken many times within a very short period. IEEE standard 1394-1995 does not take this into account. Whenever a device is connected or disconnected from the 1394 bus, a bus reset occurs. The connection or disconnection of a device actually creates storms of bus resets which disrupt isochronous traffic on the bus. P1394a solves this problem by specifying a connection time-out when devices are connected and disconnected, so only one bus reset will occur each time a device is connected or disconnected from the 1394 bus.

Eliminating the storms of bus resets and offering a more predictable timetable for bus resets allows PC devices such as isochronous printers and scanners to have smaller buffers. Since they will no longer have to try and make up for numerous bus resets, these devices will be able to more accurately predict the bus reset time. They will no longer be expecting storms of bus resets.

Arbitrated (Short) Bus Reset

The IEEE standard 1394-1995 definition of bus reset assumes that the state of the bus is not known when a bus reset is initiated. IEEE standard 1394-1995 resets disrupt packet transmission, and a node that is transmitting does not detect resets. The minimum reset assertion time must be long enough to complete any packet transmission that may have been in progress to ensure that resets are detected and propagated. In other words, the reset must last longer than the maximum

packet transmission time. However, if reset is asserted after first arbitrating for the bus the minimum reset time can be significantly reduced. This is because if one node has arbitrated and won the bus, no other node can be transmitting at that time.

By using this technique, as well as the connection debouncing technique, bus reset times can be reduced by almost 75%. This will allow for a smaller buffer size which translates into less expensive devices. Another big advantage of these techniques is that they prevent the disruption of isochronous traffic, providing more reliable transmission of audio, video, and other isochronous data streams.

Transmission Delay Calculation (PHY Pinging)

In principle, PHY to PHY pinging is the same as the classic ping. One node sends a ping to a target PHY for the purpose of determining the signal delay time. The target PHY receives and responds to the ping packet by returning self-ID packets. Knowing the maximum delay time across the bus allows the bus manager to configure the bus timing correctly. In IEEE standard 1394-1995 the bus manager was expected to assume 144ns PHY delay and 4.5 meter cables. Longer delays might have violated bus timing. By using PHY pinging, P1394a can correctly determine the maximum bus delay time and the appropriate adjustments to bus timing. Also, knowing the delay time allows the use of cables longer than 4.5 meters, or PHYs with a delay time of more than 144 ns on the 1394 bus.

Multi-Speed Packet Concatenation

This corrects a defect in IEEE standard 1394-1995 which requires a PHY to send a speed signal with only the first transmit packet of a multi-packet sequence, and at the same time requires a separate speed signal for each receive packet in a multi-packet sequence.

Many vendors have "corrected" this flaw using their own proprietary method. This is confusing and has cause incompatibilities among various manufacturers. P1394a provides a universal correction that provides a standard way of dealing with the issue. P1394a requires a separate speed signal for each packet of a multi-packet sequence. When the PHY repeats a packet that has no speed signal, it does not add one. Instead it assumes that the speed signal will be the same as the previous packet. Note that P1394a does not allow S100 packets to be concatenated to faster packets since S100 packets effectively have no speed signal.

Arbitration Improvements (ack-accelerated and fly-by arbitration)

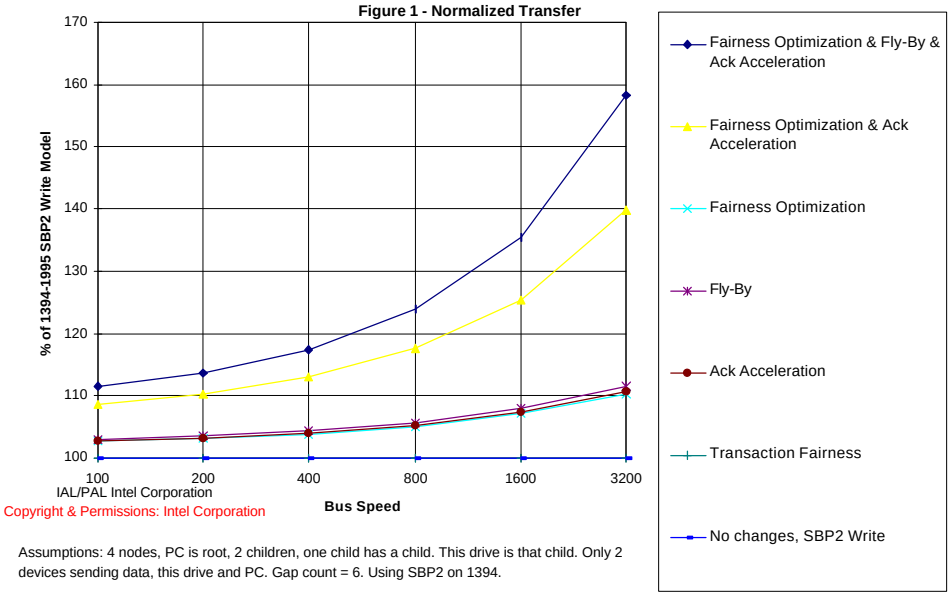
Fly-by allows a packet to be concatenated without arbitration and ack-accelerated arbitration allows a node to arbitrate for the bus without first observing a subaction gap. Fly-by arbitration occurs when a primary packet is observed and then propagated toward the root. That is, when a packet is received on a child port, concatenation can occur for two cases. In the first case, if the packet is an acknowledgement packet, an unrelated asynchronous packet could be concatenated

on the fly. In the second case, if the packet is an isochronous packet, then an isochronous packet could be concatenated on the fly. Ack-accelerated arbitration occurs following the observation of an acknowledgement packet. This is based on the fact that IEEE standard 1394-1995 made no distinction between an acknowledgement packet and a regular packet. Since there will be no acknowledgement packet following an acknowledgement packet, the subaction_gap_detect_time can be eliminated. This allows arbitration to begin immediately following the acknowledgement packet. P1394a also allows a node to concatenate unrelated asynchronous packets to its own ack packet.

Arbitration Improvements (Fairness Optimization)

Fairness Optimization is a mechanism for a node to be granted permission to use priority arbitration for requests during the asynchronous period. This basically lets a given PHY transmit more than once per period. For the entire bus, the maximum number of priority requests to send request packets is 63 minus the node count. This means that the maximum number of request packets per fairness interval remains 63. It allows the bus to be utilized more effectively and allows for greater bandwidth. Letting nodes arbitrate more than once per interval increases efficiency by eliminating arbitration reset gaps, the biggest gaps used in 1394. Fairness Optimization also makes it possible for fly-by concatenation and ack-accelerated arbitration to happen more frequently.

Another P1394a enhancement, usually called “transaction fairness” allows a node to use priority arbitration for an unlimited number of response packets.



While each of the three arbitration improvements discussed above are significant in their own right, it is not until they are combined that the real benefit is realized. Figure 1 shows compelling evidence of the importance of the combination of ack-accelerated arbitration, fly-by arbitration, and fairness optimization. The figure shows that the combination of these three enhancements provides some improvement in disk transfer speed over the IEEE standard 1394-1995 model. What is interesting about the combination of these three is that the performance gains from the combination of the three is actually greater than the sum of the individual gains! This performance gain is only true when Fairness Optimization is included in the combination of enhancements. This improvement translates into greater throughput for P1394a. This is especially good for disk drives, but also benefits other devices as well.

Port Disable (Per Port Software Disconnect)

This is a simple mechanism that allows a 1394 port to be shut down by software. A disable bit is sent to the desired PHY. The PHY is then disabled and does not respond to any traffic. This feature provides several benefits in the PC environment. By allowing the system to disable individual nodes, uncooperative nodes can be shut down without disrupting the rest of the system. The Port Disable feature can also be used as a troubleshooting tool for determining problem nodes on the 1394 bus.

This feature is also a part of the power management features proposed for P1394a. Power Management (Suspend/Resume) is currently not discussed in this paper. It will be added once the details are finalized.

Summary

P1394a fixes some of the compatibility issues in IEEE standard 1394-1995, allowing 1394 devices from various manufacturers to communicate seamlessly. By eliminating the incompatibilities between 1394 devices, users will be able to connect their 1394 camcorders, disk drives, printers and scanners with a wide range of compatible peripherals, including their PC!

P1394a offers significantly enhanced robustness over IEEE standard 1394-1995. It also offers enhanced performance (~20%) over IEEE standard 1394-1995. The robustness and performance

gains offered by P1394a lay the groundwork for 1394 storage devices.

These enhancements make P1394a the right solution for use in current 1394 designs.

Disclaimer