Audio Engineering Society

Convention Paper

Presented at the 117th Convention
2004 October 28–31 San Francisco, CA, USA

An Internet Protocol (IP) Sound System

Tom Blank[1], Bob Atkinson [2], Michael Isard[3], James D Johnston[4], and Kirk Olynyk[5]

[1] Microsoft Corporation, Redmond, WA, 98052, USA
tomblank@microsoft.com

[2] Microsoft Corporation, Redmond, WA, 98052, USA
bobatk@microsoft.com

[3] Microsoft Corporation, Mountain View, CA, 94043, USA
misard@microsoft.com

[4] Microsoft Corporation, Redmond, WA, 98052, USA
jamesdj@microsoft.com

[5] Microsoft Corporation, Redmond, WA, 98052, USA
kirko@microsoft.com

ABSTRACT

We describe a system that applies Internet concepts and software techniques to deliver audio from source to speakers using common computing hardware. The techniques overcome clocking and jitter problems. Microphones built into each transducer to locate loudspeakers allow the system to identify speaker placemet, automatically compensate for off-center listening locations, adjust for inter-channel gain, delay, and do frequency response matching. A research prototype demonstrates the concepts and measures the resulting quality.

1. INTRODUCTION

Even though there has been a shift to digital audio components, no complete audio/video (A/V) playback systems have emerged that use standards based internet protocols (IP) where third party vendors can extend the capabilities of the consumer A/V systems and leverage the cost reductions in consumer computing. We describe a system approach targeted at consumer playback systems where all media and media control are distributed over IP and media/control routing is handled as internet packet routing using standard routers and switches. For example, a conventional audio system might include a tuner, CD player, device selector, amplifier, speakers, and remote control. Our new system would connect all of the devices to the home IP network then do all data routing (e.g. play the CD output to the speakers) and control routing (e.g. press volume up on the remote) through the IP network. Each individual speaker is IP connected. Using the IP approach, every device connects bidirectionally to every other device.

The motivation for this architecture is that of convenience, allowing the customer to “just connect” his or her equipment, and have the equipment “do the right thing” automatically. Since every device is connected to every other device, numerous scenarios that are very difficult in conventional A/V systems become easy using this new model, including whole house audio (e.g. all the speakers rendering the stream from the same source.), the upstairs jukebox playing on the downstairs speakers, or the family computer that has a music library playing on the speakers in the kid’s room. Even the wiring in the house gets simpler since all media devices only need to connect to an IP network which can be wireless (e.g. 802.11a,b,g), IP over power line, 100baseT, or other network type.

A/V’s use of IP poses a number of challenges including:

- IP networks have no timing guarantee,
- Device physical locations are unspecified by their connection point (e.g. which speaker is the front left is not specified by its use of an 802.11g wireless network),
- Relationships between devices are not assigned by dedicated wiring.

Our techniques overcome these limitations.

Using the simple example of a CD/DVD player sending its output to a 5.1 speaker configuration where each speaker is only connected through an Ethernet connection, the challenge is to get all the channels to be rendered at the proper time. IP networks do not provide isosynchronous communication (isosynchronous communications provide both data and a global clock so that a receiving device can phase lock to the incoming clock and be guaranteed to never have buffer overflow or underflow e.g. S/PDIF or IEC958 are isosynchronous links.) So, we adopted a scheme that simulates isosynchrony using synchronized software clocks, time-stamped audio packets, and an audio rendering system in the speaker that can play a packet at a precise time and at a precise rate. The basic technique follows:

- All clocks are synchronized via IP protocol methods to 50 microsecond granularity.
- Spatial relationships among the speakers and the listener are automatically inferred from initial system calibration.
- The source device then constructs data packets with future rendering times providing enough latency to allow for IP transmission.
- The synchronized clocks are used to control the rendering times in order to provide proper and timely audio rendering.

Using this technique, any number of speakers can be synchronized to a source. The approach allows surround sound systems (5.1, 7.1, to all speakers in a house, and mixdown to “stereo” systems, etc.) synchronized playback. This approach is not intended for live entertainment where the latency between the source and destination devices must be as low as possible.

This system allows the use of IP protocols to connect, validate, compensate for physical positioning and room acoustics, and render audio/video in a wide variety of environments. We’ll describe the approaches and a prototype system in more detail in the following sections.

2. SYSTEM CONCEPT

Figure 1 shows an example of a simple home entertainment system using the proposed approach. The key elements of the design are source devices like:

DVD players, tuners, and computer stored source material (e.g. a Media Center PC); rendering devices like an LCD display and speakers; and finally a clock synchronization master. They are connected using a conventional internet protocol (IP) network. In this approach, each source device is connected to each sink device and that there is no central A/V switcher, amplifier, nor control point. All media flow and control flow are through the common peer-to-peer IP network.

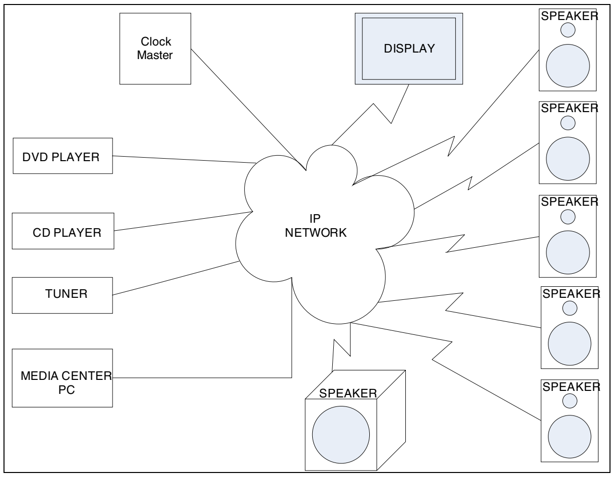


Figure 1 - Example Home Entertainment System

A simple example with a CD player as the source and two speakers as output will demonstrate basic system operation. The steps are as follows:

1. All devices synchronize their clocks to the clock master (the exact methods used will be described in more detail in section 3)

2. The speakers are configured so that each knows its position (e.g. front left, front right, etc.). Note, this is a one time operation completed at installation time (described in more detail in section 4).

3. Using a front panel control, or IP remote (voice or IR are also possible), the CD starts playing by sending out IP packets with a timestamp when the packet should be rendered.

4. The render time is based on the clock master and selected sufficiently far in the future so that network delivery variability is eliminated. Typical render times are selected 50-500ms in the future depending on the physical implementation.

5. The speakers receive and buffer the audio data. When the specified rendering time arrives, the speaker renders the packet, using its spatial position to modulate phase alignment. Further, the speaker rendering rate adjusts to the rate of the incoming packets providing frequency alignment.

Two core concepts allow the system (and previous example) to work properly:

1. Synchronized clocks and packet buffering can replicate the behavior of a synchronous network over an isochronous network.

2. The speakers (and all sink devices) can phase and frequency align the incoming media packets to their correct spatial position in the performance space.

These concepts are described in the following sections.

2.1. Replicating Synchronous Behavior over an IP Network.

IP networks are intrinsically non-isosynchronous (or asynchronous) networks that do not provide a timing guarantee for packet delivery in contrast to other networks like ATM. Nor do the lowest level IP network layers provide guaranteed packet delivery. These two problems must be addressed since an A/V system must have both synchronized playback from all sink devices (e.g. all the speakers and displays) and highly reliable data delivery.

As described earlier, buffering at the sink end with playback at a specified time provides the timing behavior of an isosynchronous network with large latency. There are three distinct cases:

- Analog Input (e.g. no intrinsic clock) – the source sampling clock can be same the as global clock.

- Stored Digital Input (e.g. from a CD/DVD where you can read the bits at any rate you like. This is similar to the previous Analog Input case.

- Isosynchronous Digital Stream Input (e.g. from wide area network, HDTV or S/PDIF source). Jitter is a possible issue here, and the system must provide adaptation, if necessary, between source clock and destination clock, preferably by locking the destination clock to the source clock. Note that problems may still exist with combined A/V streams, since the source audio and video clocks may not be locked to each other.

For either an analog input or a stored digital input, the input data rate can be slaved to the Clock Master. For example, an analog input (e.g. from a baby room monitor) could use an A/D converter whose clock is phase locked to the Clock Master. This guarantees that all sinks phase locked to the Clock Master are functionally phase locked to the incoming data rate. Similarly, for a stored digital input device (e.g. from an audio CD in a computer), the samples could be input at 44,100 samples per second with respect to the Clock Master. This again guarantees that all sinks phase locked to the Clock Master are locked to the source.

The key observation is that the input and output rates are synchronized to the Time Master. Therefore, they are synchronized (phase and frequency aligned) to each other which guarantees a fixed buffer size at the sink will never over or underflow.

The more complicated case inputs data at a rate which is neither phase nor frequency aligned to the local Time Master. An example would be an HDTV or satellite digital stream where the input clock rate is established at the signal origin e.g. the TV station or satellite head end. In this case, the input data stream is packetized and the starting data arrival time is time-stamped into the local packet stream using the Time Master (plus a small future offset described earlier). At the sink device, the incoming packet is buffered until the packet specified time then released to the rendering unit. At this point, the rendering unit receives the input data at exactly the same rate as broadcast from the source but with a fixed latency. The rendering unit, then phase locks to the incoming data rate and plays the data. Phase locking again guarantees that a fixed buffer cannot over or underflow.

Using the previously described methods, any source can be distributed over an asynchronous IP network and be rendered synchronously throughout a home.

2.2. Device Phase and Frequency Alignment

The previous section described device time related requirements:

1. All devices must synchronize their clock with the Time Master.

2. Source devices must timestamp all outgoing A/V packets with a future rendering time.

3. Sink devices must provide packet phase alignment by buffering packets until a specified time.

4. Sink devices must provide frequency alignment by locking the rendering rate to the source data input rate.

The previous four requirements allow an asynchronous IP system to replicate the behavior of an isosynchronous hardwired system. Clock synchronization is described in the section 3.1. Source time stamping is straight forward and simply needs to be done with as little timing variation as possible using either software or hardware mechanisms. The third requirement (buffering until a specified time) is again straight forward using either software or hardware.

The fourth requirement, while not conceptually complicated, must be carefully implemented. This goal requires that the output rendering rate be matched to the incoming data rate and that the physical output occur at a fixed delay after input. The fixed delay is necessary to maintain the phase alignment provided by step three and to respect the physical position of the speaker with respect to the listener. Again, either hardware or software methods can be used.

3. CLOCK SYNCHRONIZATION

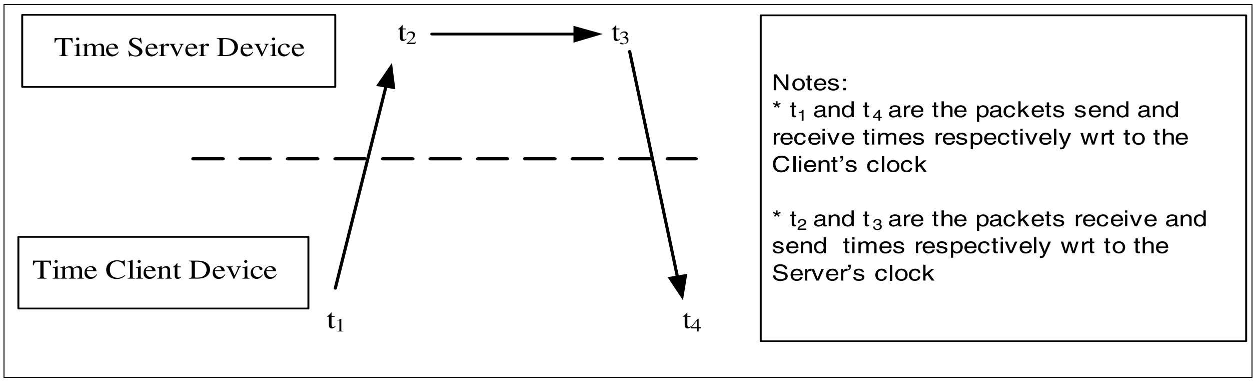
One of the core components of the IP media system is tightly synchronized time between all of the sources and sinks. The system provides the tightest time synchronization possible over variations in source device components, network variations, and environmental variables. A variety of techniques are possible ranging from a centralized time source wired to each device (like in a studio), a GPS connection for each device, a locally broadcast radio signal, phase locking to the AC line frequency, or an IP network based time synchronization technique. The IP based synchronization approach has the lowest software cost (a little more code in the already existing CPU), no additional hardware cost, and no additional wiring. The largest disadvantage of the software only approach is reduced accuracy. The following sections describe our use of the software time synchronization method and its accuracy.

3.1. Software Based Time Synchronization

A software based time synchronization system typically sends packets between a client and a server comparing their respective time stamps to generate an offset value which is then minimized by adjusting the client's clock. Using typical A/V device construction techniques (e.g. a micro-controller, crystal, dedicated function units, and a network interface), the largest time synchronization obstacles are IP network delivery variability, OS timing variability, and tracking crystal timing differences caused by temperature variations.

Our software time synchronization system uses the industry standard version of the Network Time Protocol (NTP)[1] wire protocol using a highly modified implementation. The basic NTP concept allows time clients to query a server for the proper time (See [2] for a History of NTP developments). Based on the premise that on average, the forward and reverse network times and OS times are equal, a time difference can be calculated as shown in Figure 2. If any asymmetry exists in either the network or OS processing path, the difference will be included as part of the clock offset. Equation 1 provides the time offset between two machines:

$$Offset \equiv \frac{(t_2 - t_1) - (t_4 - t_3)}{2} \qquad \textbf{(1)}$$



Notes:
* t_1 and t_4 are the packets send and receive times respectively wrt to the Client's clock

* t_2 and t_3 are the packets receive and send times respectively wrt to the Server's clock

Figure 2 - NTP Packet Timing Diagram

The offset computation is completely based on the four timestamp values. Therefore, the better the precision and accuracy of the timestamps, the better the offset value.

Finally, the client slowly adjusts its clock to minimize the offset so that slaved audio rendering systems generate no audible artifacts (as described in the next section).

The primary error sources are: offset measurement variability due to network variations and operating system (OS) scheduling uncertainty; and crystal variations causing clock frequency differences. Network variations are highly dependent on the physical network type and load. For example, wireless or powerline networks are far more susceptible to radio frequency noise sources like bad brushes on a vacuum cleaner motor or a cordless 2.4 GHz phone that uses the same frequency band as an 802.11b wireless network. Further, widely changing network loads cause significant timing variations due to network delays.

OS variations cause errors due to the variability of the actual time the device received (or sent) the packet and the recorded time. Ideally, all hardware network interfaces would be modified so that packet time stamping would occur as a hardware function without any OS intervention (or variability). However, this is not available today in commercial devices. The best practical implementation simply puts a time stamping driver as low as possible in the IP stack common to all network interfaces.

The final primary error source is crystal variation. The motivating concern includes the scenario in a consumer home where an A/V device may sit close to an air-conditioning vent where substantial temperature swings (e.g. 1 degree centigrade per minute) are possible as the heating/cooling system cycles during the day. Typical consumer devices use inexpensive quartz crystals that may have a +/- 100ppm variation over an 80 degree centigrade temperature range. The simplest problem is that the base frequency will start with an offset between two devices (typically less than 50ppm difference). However, the more challenging problem is the temperature dependent frequency variations causing the clocks to drift apart. The exact function is manufacturer and operation dependent with a typical number less than .5 ppm per degree centigrade around room temperature. Even though the temperature change has a relatively small impact, over time, even a small difference will quickly become noticeable. For example, a 1 ppm clock difference adds an additional 1 ms error every 17 minutes.

Even though we used the standard NTP protocol on the network (e.g. the network packet bit definition is unchanged), we've made substantial implementation changes in the client and server code currently supplied from http://www.ntp.org. In contrast to the standard NTP implementation where the objective is to provide good time synchronization with the least possible interaction with the time server, our time synchronization goal was to provide the best time synchronization within an acceptable network and server load in a home environment. The following table shows the critical differences between our target home environment and the typical wide area network (WAN) internet environment:

	Typical Home	**WAN**
Number of Clients	<1000	>10e7
Network Topology	No or Few Routers; simple switched, non-redundant paths	Complex routing fabric with a myriad of network devices, and numerous redundant paths

Table 1 - Home versus WAN Differences

The critical observation from the previous table is that the home environment is small with a simple network environment allowing substantial simplifications and accuracy improvements in our implementation.

The basic architectural differences required to deliver higher precision in our NTP implementation include:

1) All time critical software components run in kernel mode (primarily the packet time-stamping). This eliminated variability due to user mode process execution variability which can be multiple milliseconds variation when the system is busy.

2) Time-stamped the packets at the lowest possible network layer that is not device specific. This provides the highest quality timing information that is least effected by system load.

3) Used the highest precision reliable system clock (or counter) available from the hardware. This is hardware dependent but ranged from 1 ms to better than 1 ns where each clock/timer has pros and cons.

4) Additional prefiltering components to eliminate irrational timing packets.

5) Group polling where a number of packets are sent at each polling interval[3]. The current NTP specification states that only one NTP packet is sent at each polling interval. Even in a home network, large network and system variations demonstrate that a single packet is statistically unworkable. By sending out a polling group (e.g. 5 packets and selecting the median value) at each polling interval, a much more accurate time measurement is possible.

6) Simplified and tuned software phase lock loop (PLL) for core client clock control. This replaces the phase/frequency loop locking technique specified in the NTP RFC. This change provided smoother offset transitions with better tracking of temperature variations in home consumer devices.

7) Reduced the polling interval. By changing the polling interval to four seconds, we are able to track one degree centigrade per minute changes while maintaining tight time synchronization.

The results described in section 6.1 indicate an average error of +/- 50 µs with a change time constant of roughly one minute.

3.2. A/V Quality versus Time Synchronization Accuracy

Lip-sync between audio and video is maintained if the following criterion is met: -20 ms < delay < + 40ms[4]. The AES11-1997 standard for digital studio operations specifies clock accuracy but does not address absolute time stamp accuracy for independent multi-channel audio playback (Caine provides a good description of studio timing issues in [5].

Unfortunately, no studies have been discovered that provide a detailed analysis of timing requirements for multi-channel audio playback. A close relevant observation is that humans are able to detect stereo click arrival times to about 10 µs; however, this does not seem to be directly applicable to the delay requirement between stereo speakers, since any fixed delay seems to only represent a repositioning of the preferred listening position. In a surround sound system, a fixed delay should be equivalent to speaker misplacement by roughly 1 inch per 100 µs delay, which is likely very hard to hear, especially since the speaker would not actually be moved, and the reflections from each individual speaker will not, therefore change. All that will change is the relationship between speakers.

A more detectable problem is variable delays. However, if the rate of delay change is slow, the effect should again be undetectable. In our case, the effect is that of moving speakers by .5 inch or less, without changing the surrounding acoustics. While this kind of delay shift, when done quickly, may create audible image changes, it is very hard, if at all detectable, to hear when the system changes this much with a 1 minute time constant.

In our case, the usual considerations of jitter do not apply, as the bandwidth of this jitter is on the order of centiHertz. This kind of jitter can possibly, if large enough, create audible pitch shifts, or **wow**, on a much slower scale than that of old LP technology, however in our case the change and magnitude of the shift is slow enough that such pitch change is unlikely.

The key to this system is that all changes are slow, having a bandwidth of something around 1/60Hz at most.

4. SYSTEM CONFIGURATION AND CALIBRATION

The scenario targets a customer installing an IP based home theater in a box. This section deals with the installation and configuration issues specifically focusing on a surround sound audio installation (there are few unique IP video issues beyond discovery which are simpler than the audio problem.) Two specific audio problems are addressed:

- identifying the speaker positions

- calibrating the preferred listening position.

To simplify the discussion, an example IP home theater in a box contents would likely contain the following:

1) wireless IP amplified speaker (quantity 5)

2) wireless IP amplified subwoofer (quantity 1)

3) Media Center PC and display with wireless IP network interface (quantity 1)

The goal is to enable the customer to get a 5.1 surround sound system operational. The following sections describe our calibration solution.

4.1. Speaker Position

In contrast to a conventional home theater system where the speakers are directly wired to an output that identifies its position (e.g. Front Left, Surround Right, etc), an IP based system does not have this information. Whether using IP wireless or wired technology no physical position information is provided (Note: It might be possible to use wireless signal strength as a mechanism to roughly determine speaker position but errors would be large, and no directional information would be available.)

Our approach solves the configuration problem by incorporating two inexpensive microphones into each speaker. The configuration sequence overview follows with more detailed descriptions latter:

1) All the nodes synchronize their clocks.

2) Using conventional computer networking techniques (e.g. uPnP), the PC is capable of automatically discovering the IP addresses of each of the 6 speakers and their capabilities.

3) From the capabilities, the subwoofer is uniquely identified but the other 5 speakers are identical (note, the center channel could optionally be uniquely identified but this does not change the problem of identifying the position of the remaining speakers).

4) The host controller sends an audio probe signal to each speaker in turn and that speaker emits the probe signal.

5) All loudspeakers hear the signal (or not) via their two microphones (note possible to use only one mic per speaker but is more error prone). They send these time-stamped signals to the host controller.

6) Using the timing data from each speaker, distances and angles to the source speakers can be calculated by solving a set of simultaneous equations.

7) From the distance/angle data, the identity of each speaker can be determined in a 5.1 system by using typical geometrical constraints (e.g. the front center speaker is close to two other front speakers and roughly centered.)

In the following sections, we describe the design and characteristics of the test signal, and methods used to calculate both the delay and angle from the microphone data in more detail.

4.1.1. Test Signal Design

The primary design objective for the audio test signal is a strong correlation function peak, while having a bandwidth that excludes ambient noise. A strong autocorrelation peak helps, as well, in the presence of ambient noise. We experimented with a number of different signals including single tones, white noise, and bandwidth limited noise, and found that a continuous phase bandpass signal that has pseudo-pulse characteristics exhibited the best behavior. Figures 2, 3, 4 show the time domain, frequency domain, and output correlation of our test signal. Note, we built our signal from 500Hz to 10KHz, since the autocorrelation process implements a matched filter and can exclude the most likely frequencies for ambient noise, (i.e. air-conditioning and high frequencies where the signal may not be well-rendered.

The test signal is convolved with the time-reverse of the sequence in the detector in order to create a linear-phase system. This property allows us to recognize other characteristics of the transducers and room above and beyond gain and delay.

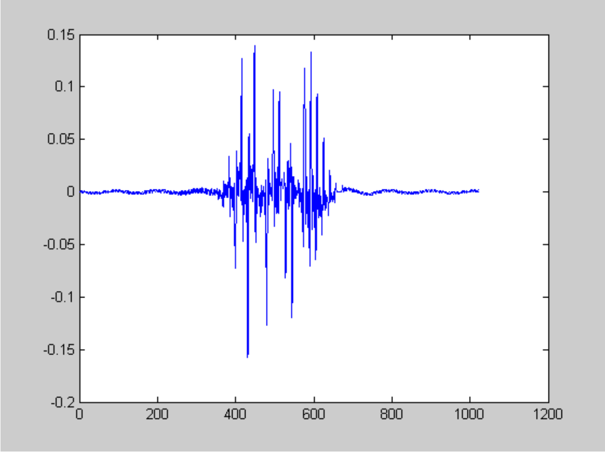


Figure 3 - Test signal in Time Domain

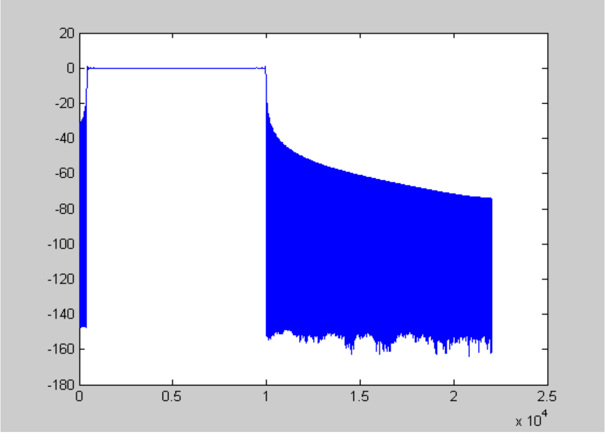


Figure 4 - Test signal in Frequency Domain

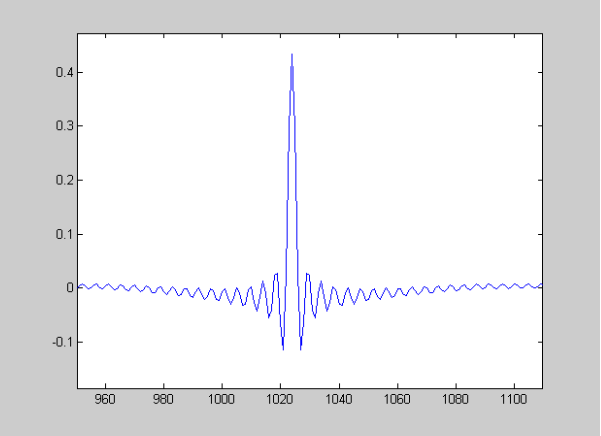


Figure 5 - Test signal Correlation Output

4.1.2. Distance/Angle Computation

In section 4.1, the final two steps calculate the x/y coordinates and finally the positional name (e.g. front left) name of each of the main speakers. This section describes in more detail the steps used to perform this action.

Synchronized time is critical to the distance/angle computation and we use the previously described modified NTP approach. Specifically, the previously described test signal is emitted precisely at a known time. Further, the time-stamps on the received signals are using the globally accurate time. Therefore, distances between the speakers can be determined by simply calculating the time difference between transmission and receipt and multiplying the by speed of sound. The first challenge is accurately determining the test signal arrival time.

The simplest approach worked best for the distance measurement where we convolved the received signal with the transmitted test signal. Similarly, the angle to the source speaker is calculated by convolving the signals from the two microphones. In both the previous cases, the actual distances/angles were not necessarily the largest peak since the calibration is performed in an environment delivering significant echoes. We've seen reasonable examples where the first or second echo was stronger than the incident wave.

Our current implementation finds the convolution maximum within a specified range (or angle). If an earlier local maximum is greater than 50% of the absolute maximum, the earlier value is used. We discovered that this approach worked well in our test cases.

4.2. Listening position calibration

At this point in the configuration sequence, the computer now knows the relative x and y coordinates of each of the speakers except for the subwoofer which is nearly position independent. The primary remaining calibration step is to configure the system for the preferred listening position. In an ideal physical configuration e.g. that meets ITU-R recommendation BS.775[6], no system changes would be required. However, in a typical consumer home installation, the speakers are installed on (or close to) walls in a rectangular room without attention to listener distance, yielding a substandard listening arrangement without at least delay compensation. Further, the consumer listening position may not be geometrically centered due to decorating/physical constraints. The following calibration sequence allows delay and optionally amplitude calibration:

- The user is instructed to move to the preferred listening position and clap their hands after a beep is played.
- Each of the microphones records and timestamps the clap (the technique identifies the handclap as the loudest sound in a 2 second window after the beep).
- Even though the exact time and signature of the clap are unknown, the x/y coordinates of the location can be calculated by solving the simple simultaneous equations based on the known speaker coordinates and the recorded clap times from each speaker.
- From that information, gain and delay parameters can be constructed for each speaker. While we do not presently do it, it is also possible to create an appropriate test signal for the subwoofer, and align the subwoofer similarly in time and frequency. Such a subwoofer test sequence would be longer than 1024 samples, however.

Additionally, if a calibrated test signal source e.g. a speaker reproducing the test signal, is used at the preferred listening position, complete room acoustic equilibration is possible again using the microphones in the speakers. Note, this is the mathematical dual to the method used by companies like Bose where a microphone is placed at the preferred listening position, except that with two microphones per speaker, more even optimization over space is possible.

4.3. Other Loudspeaker Positioning Issues.

Presently, we have only two microphones, located on a horizontal axis on the loudspeaker. Therefore, we can not measure vertical loudspeaker misalignments or placements, however we can still measure the actual distances between speakers well enough to determine that there must be a vertical component to loudspeaker positions. In the future, we may increase the number of microphone channels, and calculate vertical displacement, should this become an issue with actual applications.

In most cases, the time delay from the speaker to the listener is the most important parameter, and we can properly measure that irrespective of vertical positioning, and if need be, compensate for the different delays from different channels to the listening position.

Presently, we can measure the angle of each loudspeaker position in regard to the other loudspeakers. In a practical system, if a particular loudspeaker is turned in a fashion that may affect system performance, we could report that back to the user, and suggest that they re-aim the loudspeaker in question more toward the listening space.

5. PROTOTYPE SYSTEMS

Two different prototype systems were constructed. The first system focused on concept testing. The second prototype was a complete 5.1 audio surround system designed to verify multi-channel audio system issues.

The concept prototype validated two key ideas: first, IP based time synchronization and second, slaving an audio rending clock to the IP based clock allowing precise-time audio rendering. Figure 6 shows the concept test system block diagram.

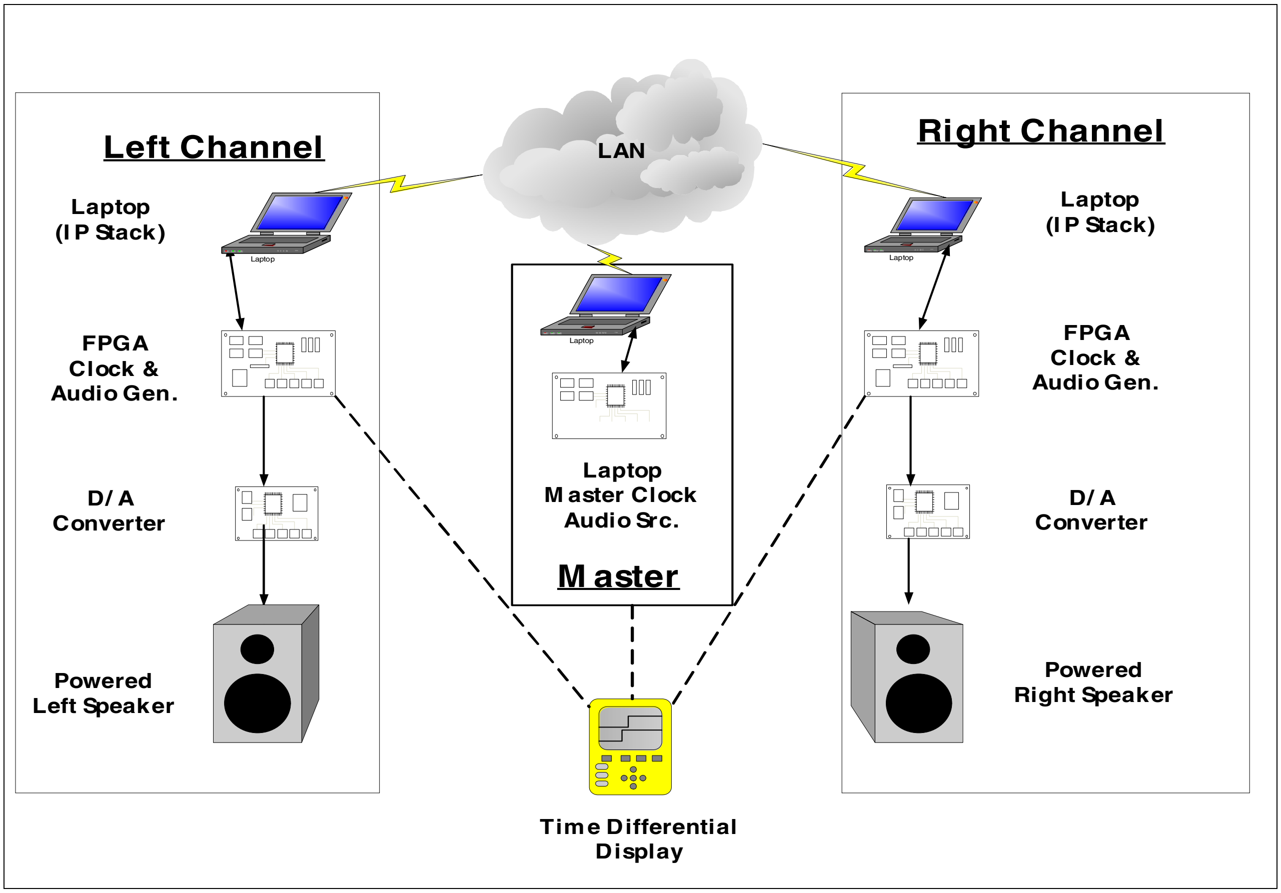


Figure 6 - Concept Prototype Block Diagram

The component functions are:

- **Master** – acted as the master time source and as the audio stream source. Constructed from the following components:
    - **Laptop** – provided IP connectivity for both audio streaming and time sync. The software system was built on Microsoft's Windows XP.
    - **FPGA Board** – prototype FPGA board contained a hardware clock used for system time.
- **Left/Right Channel** – rendered the audio stream both phase and frequency locked to the Master. Constructed from the following components:
    - **Laptop** – provided IP connectivity for both audio streaming and time sync. The software system was built on Microsoft's Windows XP.
    - **FPGA Board** – contained a hardware NTP clock used for system time, S/PDIF generator phase locked to the NTP hardware clock with audio data queue from Laptop. Used a PCMCIA interface for laptop interaction. Also contained dedicated interface allowing clock difference measurement.

- **D/A Converter** – Used an evaluation D/A board to convert S/PDIF to analog line out.
- **Powered Speaker** – provided the final signal output.

The motivation for using the external clock and audio system versus using the internal PC system was driven by the need for accurate inter-system clock measurements and the difficulty in a PC of rendering an audio sample at both a precise time and rate.

The first prototype system implemented a stereo music system. The Master acted as both the clock master and as the music streaming source. The left and right channels received, buffered, and played the streams with both phase and frequency alignment.

The second prototype reduced the ideas into a more practical form. Again using the basic ideas of system wide synchronized time, IP networking, and time-stamped A/V packets, Figure 7 shows the system block diagram of the 5.1 surround sound system.

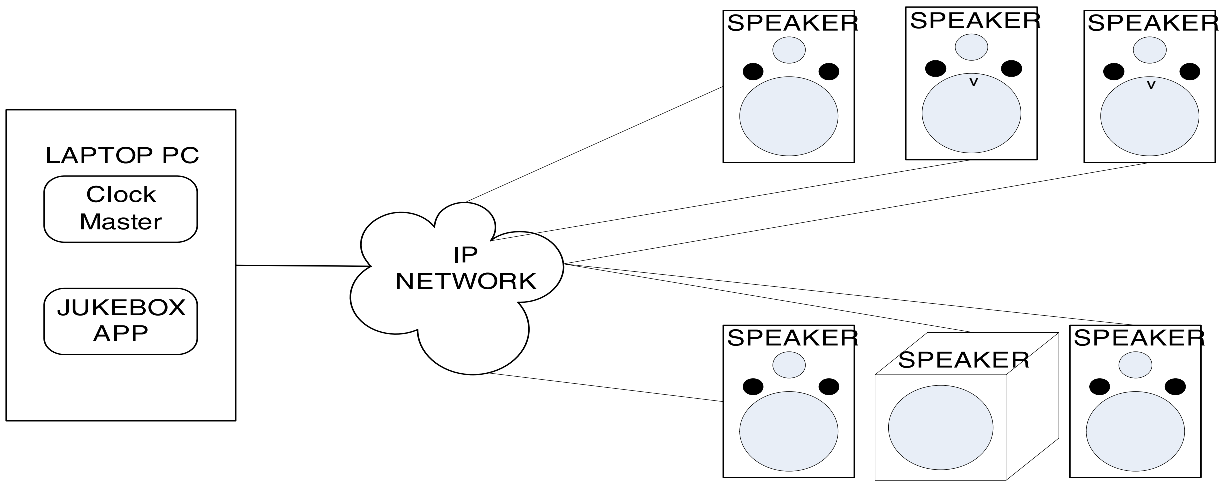


Figure 7 - Prototype System Diagram

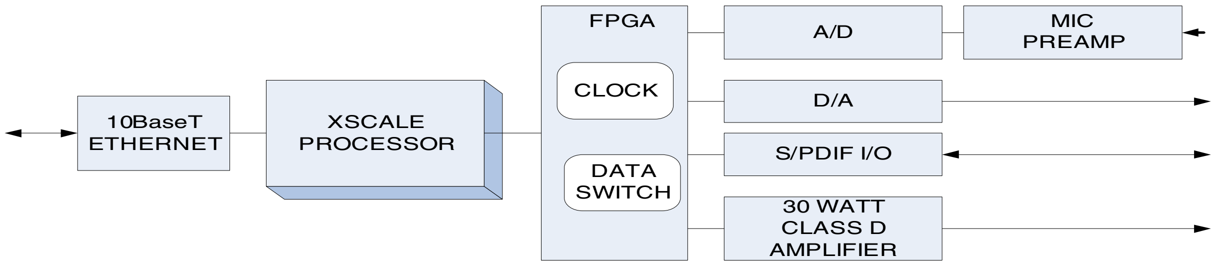


Figure 8 - IP Speaker Block Diagram

The system includes an unmodified laptop PC running Windows XP, network interconnect, and six IP speakers running WinCE. Figure 8 shows the block diagram of each of the speakers. The key components are functionality similar to the previously described concept prototype providing a hardware clock, and synchronized audio path. However, the speakers have been modified to include two microphones.

For both prototypes, custom application software streams the uncompressed 44.1Khz 16 bit samples to the speakers. Our system uses TCP/IP network connections providing reliable data and command delivery. The master jukebox application schedules audio packets to play 500 ms in the future. Commands like volume changes, pause, and, stop are processed immediately on the speakers enabling a responsive user experience.

6. RESULTS

The following sections describe the results obtained using the previously described prototype systems. We'll specifically address time synchronization, speaker position calibration, preferred listening calibration, and whole system listening.

6.1. Time Synchronization

The time synchronization results primarily depend on the network topology, network load, client device, client load, server device, server load, polling interval, system crystal, and temperature variations. Numerous papers have published results addressing network based time synchronization including [7,8,9]. Our measurements focus only on cases that would typically be found in a home environment with simplified networks and typical consumer electronic devices. We've included specific examples for networks using wired, AC power line, and wireless networking which we believe will encompass the majority of future home networks.

Figure 9 and Figure 10 show a representative NTP run using the hardware configuration shown in Figure 6 (prototype 1). We used a simple network consisting of 100BaseT interconnects to a switch (note, no impact is seen using either a 10BaseT or 1000BaseT configuration). The first plot shows the measured clock error between the Master and Left channel clocks using a hardware technique that eliminates software sampling errors. The error standard deviation was 5.6 µs around a mean of 39.2us.

The mean error is not a problem for the audio system since any fixed error will be removed during system calibration. The possible components of the fixed error include: network delay asymmetries, driver asymmetries, or OS IP stack processing asymmetries. For our prototype system, we identified 19 µs of error due to the IPSEC (IP Security Protocol) processing used on our corporate network.

The second graph shows the smoothed PPM adjustments applied to the Left channel clock. The basic graph shape shows the building temperature cycle where the day/night changes at 7:00AM and 7:00PM typically changing the office temperature by 2-3 degrees centigrade. The variations during the day are due to significant temperature changes as the office door is opened/closed significantly changing the airflow typically causing about a one degree centigrade change. Additional daytime variations are caused by the sun coming into the office (yes, the sun actually shines sometimes in Seattle). The key observation is that the errors remained constant and small even though significant environmental changes were encountered.

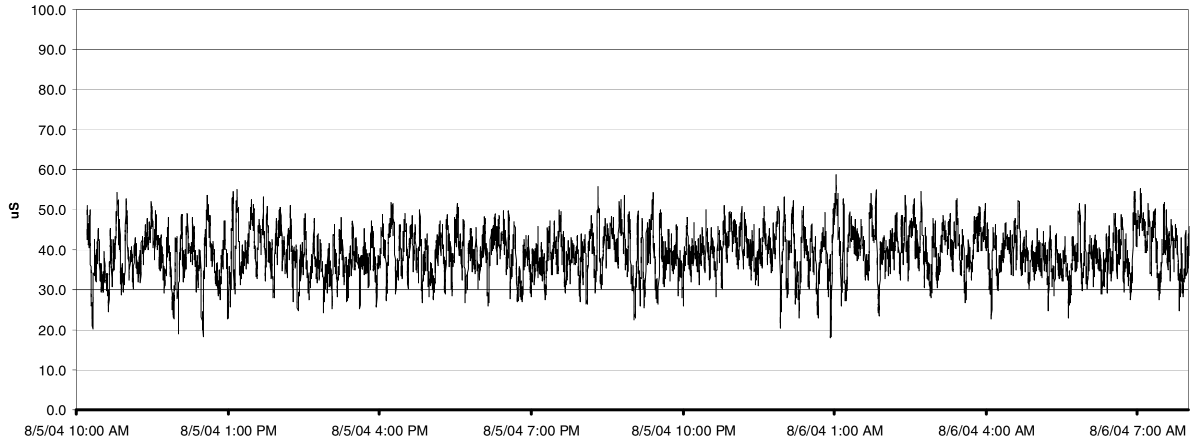


Figure 9 - NTP Timing Errors (Wired)

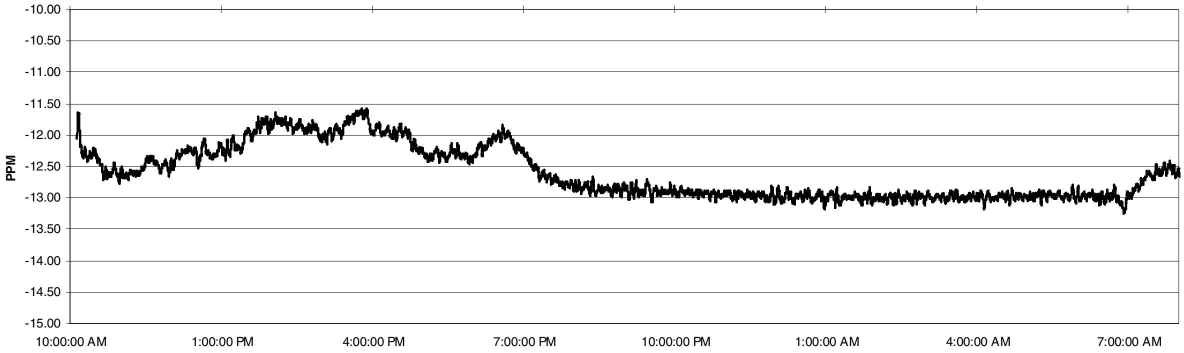


Figure 10 - NTP PPM Adjustments (Wired)

Figure 11 shows another representative NTP run using the exact same configuration and approach except that the network used HomePlug[10] network adapters between the left channel and the network switch. The HomePlug technology uses the AC power line to carry an IP data stream without any additional wires. The average error value was 41 µs with a standard deviation of 9 us.

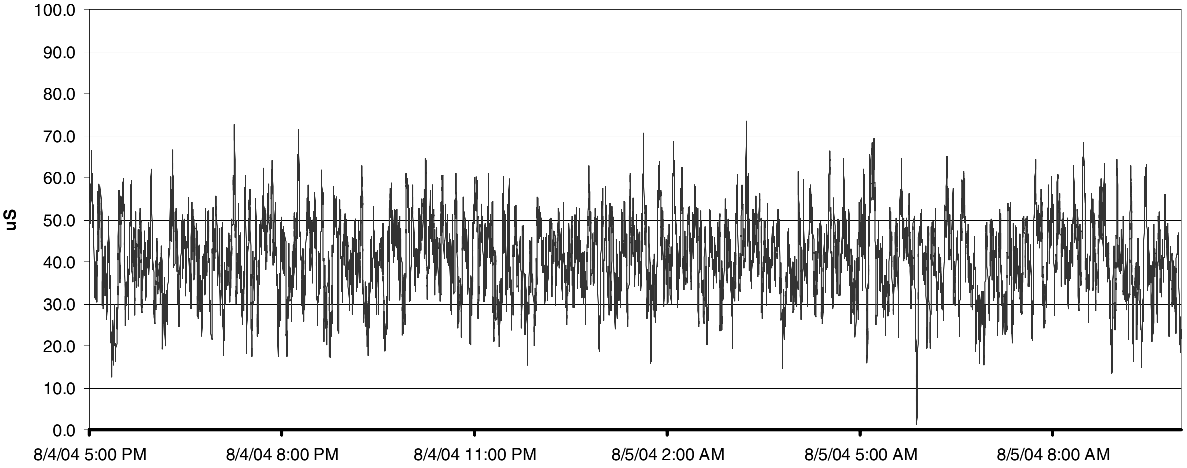


Figure 11 - NTP Timing Errors (Power Line)

Figure 12, again uses the prototype 1 configuration with a wireless connection between the left channel and the master. The network path included three switches connected to a corporate network router, one wireless access point and an internal 802.11 built-in interface on the laptop. During this time period, temperature variations caused the client to change from -11.25 ppm to -13.25 ppm. The large error timing spikes were during times of rapid temperature change, network traffic, or OS load induced error. The mean error value was -7 µs with a standard deviation of 7 µs .

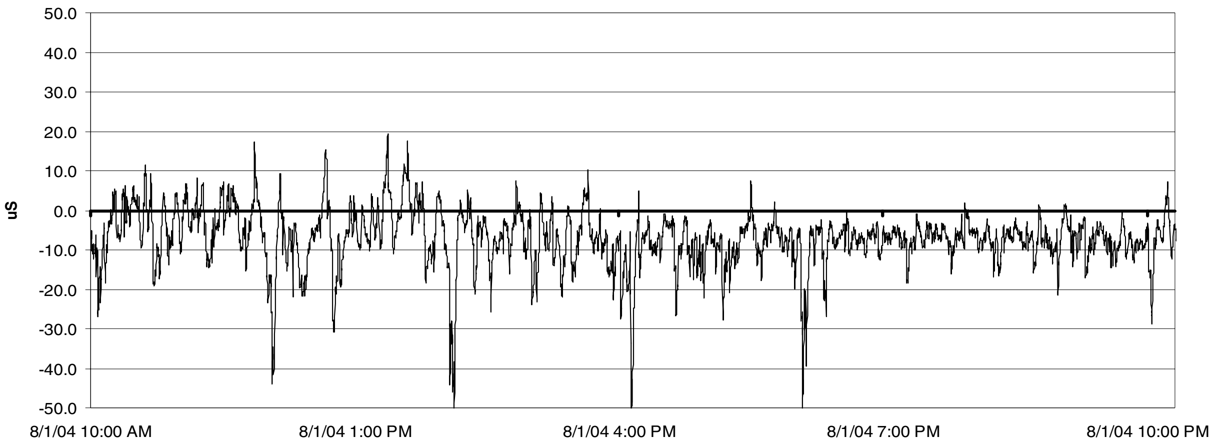


Figure 12 - NTP Timing Errors (Wireless)

The previous timing results show that accurate time keeping is possible using a variety of networks, configurations and over a range of environmental conditions. Even though the fixed time differences may be much larger than one sample period (44.1 KHz sample rate yields 22.68 µs), this error is not critical since it will be removed during system calibration. Finally, the standard deviation error is small and moves slowly which should generate no audible artifacts.

6.2. Speaker and Preferred Listening Position Calibration

As described in section 4, system calibration occurs in two steps: 1) identifying speaker positions and, 2) identifying the preferred listening position. Data was collected for two different room configurations where the large boxes represent the actual device coordinates and the diamonds represent the calculated data. All coordinates are plotted in feet. The subwoofer position is not included and the center most reference point is the preferred listener position. The system arbitrarily set the front center speaker position at coordinates (0,0), and the front left speaker at (-x,0). All other coordinates are calculated. The heavy bounding lines show the room walls.

Figure 13 shows an example calibration data set output from a business conference room. The left and right walls are covered by white boards creating strong problematic echoes. All the speakers were located at the same height. Seven runs were made.

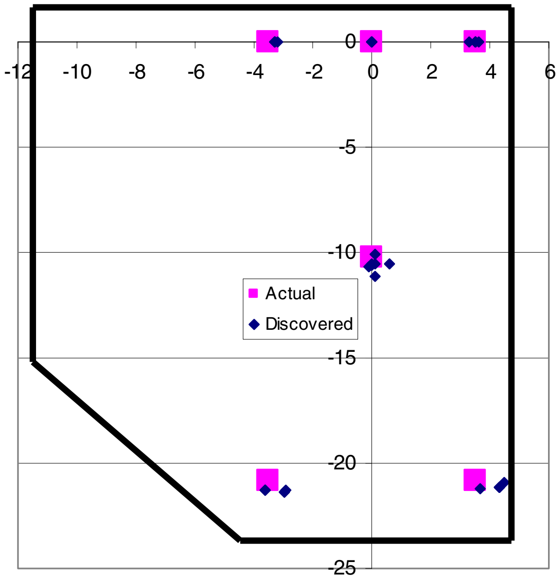


Figure 13 - Calibration (Room 1)

The measured data shows a strong correlation to the actual data where the mean error was 4.2 inches with a standard deviation of 4.0 inches across all the data points.

Figure 14 shows the data from a more typical home configuration in the “bonus” entertainment room above a garage. The heavy lines again show the room outline where the dashed line represents a picket style railing overlooking a downward stairway. All the speakers are again at the same 4 foot height. The left and right walls had large windows but were covered with light curtains. The positions were chosen to match the existing home theater speaker installation.

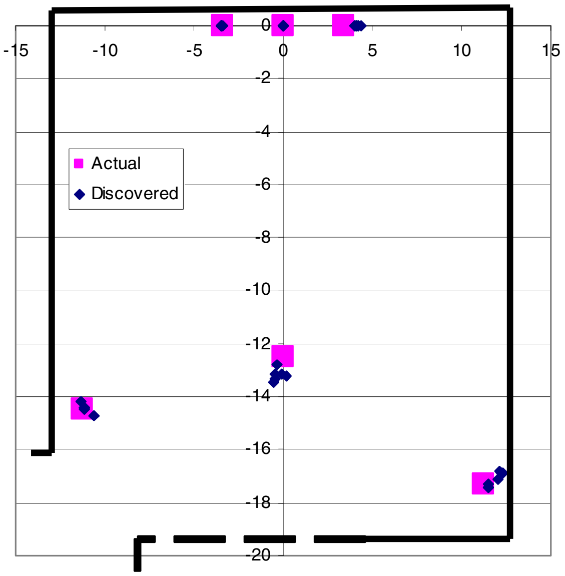


Figure 14 - Calibration (Room 2)

The mean measurement error was 5.0 inches with a standard deviation of 4.5 inches.

In the previous examples, the correlation between the measured and actual data was good with average errors less than 5 inches. However, the data did show a slight right hand bias which we believe can be corrected by doing an initial calibration (done once per speaker).

6.3. Whole System Listening

Informal listening tests have been performed in the previously described room configurations and in many others. We used clarity and consistency of the audio image as the key criteria. We were especially looking for any audible artifacts that seem to change position or move. We trained on a conventional 5.1 surround sound audio system so that we were both familiar and comfortable with the listening material, then listened to the prototype two system

Even though we were unable to do a switched A/B comparison due to complex system issues (e.g. integrated speaker electronics), we did listen carefully for unnatural artifacts caused by using the IP based system. In our tests, we were unable to detect any noticeable changes.

7. CONCLUSIONS

We’ve described a system concept that is completely based on internet protocol interconnections between all A/V devices allowing great flexibility and configurability. Home A/V scenarios like whole house audio, accessing media located in another room, or allowing simple/inexpensive upgrades become simple. The idea leverages the huge growth in both home computers and home networking. Your stereo or A/V system becomes just as flexible and customizable as your home computer. You’ll be able to install software upgrades to fix problems or enable new features. Further, the idea allows the same accelerated cost savings as personal computing where you can purchase a two times better computer every two years for the same price.

The basic idea establishes a common time across all home A/V devices. Each source device time-stamps all A/V packets with the time they should be rendered. Each sink devices then holds the received packets until the specified time when they are rendered at the incoming rate. This approach overcomes the IP network non-guaranteed latency problem as long as the rendering time is selected sufficiently in the future.

The approach is no panacea. For example, the system will not work if the home network is too slow or unreliable to support the A/V needs. However, home networking reliability and performance have achieved similar improvement gains as described for personal computers. Further, the devices are not perfectly time synchronized and have built in latency so are not appropriate for a studios or live performances. However, our tests have shown excellent results for a typical home experience.

Our prototype results have demonstrated timing errors under 50us with only very low frequency changes. Further, we’ve shown that automatic calibration is both simple and accurate to less than .5 foot. This kind of accuracy is better than that reached by most home installations, and is close to the limits of hearing as far as interchannel position and delay are involved, except in cases where a direct comparison to inter-channel delays is available. Our approach has the potential to radically improve the customer experience, allow scenarios like whole house audio, which were previously difficult or impossible to achieve, because delay and gain can be adjusted per-loudspeaker, and provide a better quality experience with nearly automatic installation.

Finally, we're continuing our investigation focusing on a number of system extensions including user interface issues for whole house A/V, more precise timing approaches, network protocols, content protection, and techniques for reliably reducing system latency.

8. REFERENCES

[1] Mills, D.L. "Network Time Protocol (Version 3) specification, implementation and analysis." Network Working Group Report RFC-1305, University of Delaware, March 1992.

[2] Mills, D.L., "A Brief History of NTP Time: Memoirs of an Internet Timekeeper," ACM SIGCOMM Computer Communications Review, vol. 33, issue 2, pp 9-21, April 2003.

[3] Levine, J. "Time Synchronization Using the Internet," IEEE Trans. Ultrason., Ferroelect., Freq. Contr., vol. 45, no: 2, pp 450-460, 1998.

[4] Wilkinson, James H., "Communications in the Digital Audio Studio," Paper 7-038; The AES 7th International Conference: Audio in Digital Times; pp 263-267, April 1989.

[5] Caine, Robin, "Timing Issues," Paper MA-07; AES UK Conference: Moving Audio, Pro-Audio Networking and Transfer; April 2000.

[6] ITU-R Recommendation BS.775-1, "Multichannel Stereophonic Sound System with and without Accompanying Picture," International Telecommumcation Union, Geneva, Switzerland, 1992-1994.

[7] Mills, D.L, "Adaptive hybrid clock discipline algorithm for the network time protocol," Networking, IEEE/ACM Transactions on, Vol. 6, Issue: 5, Pages: 505 – 514, Oct. 1998.

[8] Jeremy Elson, Lewis Girod and Deborah Estrin, "Fine-grained network time synchronization using reference broadcasts," ACM SIGOPS Operating Systems Review, Volume 36 , pp. 147 – 163, 2002.

[9] Liao, C., M. Martonosi, and D. Clark. "Experience with an adaptive globally-synchronizing clock algorithm." Proc. 11th Annual ACM Symposium on Parallel Algorithms and Architecture, pp. 106-114, June 1999.

[10] See http://www.homeplug.org.