Transmission of 1394 Protocol Over an RF Link

ECE 4040 Progress Report
April 4, 2000
Table of Contents

<table>
<thead>
<tr>
<th>Section</th>
<th>Page</th>
</tr>
</thead>
<tbody>
<tr>
<td>Introduction</td>
<td>p. 3</td>
</tr>
<tr>
<td>How IEE 1394 Works</td>
<td>p. 5</td>
</tr>
<tr>
<td>Explanation of RF Link</td>
<td>p. 9</td>
</tr>
<tr>
<td>PC Board Design</td>
<td>p.14</td>
</tr>
<tr>
<td>Micro-Processor Operation Section</td>
<td>p.17</td>
</tr>
<tr>
<td>Summary/Status</td>
<td>p.20</td>
</tr>
<tr>
<td>Appendix</td>
<td>p.22</td>
</tr>
<tr>
<td>Previous Attempts at Theoretical Solution</td>
<td>p.23</td>
</tr>
<tr>
<td>Web Site References</td>
<td>p.28</td>
</tr>
</tbody>
</table>
**Introduction**

People are sharing video, still images, and audio, and are constantly searching for faster, easier ways of transferring such information. The IEEE 1394 protocol appears to be a strong candidate for these communication channels. The IEEE 1394-1995 protocol had its genesis at Apple Computer, which still retains the Firewire trademark. The goal of the protocol is to provide easy-to-use, low-cost, high-speed communications. The protocol is also very scaleable, provides for both asynchronous and isochronous applications, allows for access to vast amounts of memory mapped address space, and possibly peer-to-peer communication.

The IEEE-1394 High Performance Serial Bus is a versatile, high-speed, and low-cost method of interconnecting a variety of personal computer peripherals and consumer electronics devices. The IEEE-1394 bus began life in 1986 as Apple Computer's alternative to the tangle of cables required to connect printers, modems, external fixed-disk drives, scanners, and other peripherals to PCs. The proposed standard (P1394) derived from Apple's original FireWire design, was accepted as an industry standard at the December 12, 1995 meeting of the Institute of Electrical and Electronics Engineers (IEEE) Standards Board. The official name is IEEE 1394-1995 Standard for a High Performance Serial Bus. The 1394 Trade Association was formed in 1994 to accelerate adoption of the Bus by personal computer and consumer electronic manufacturers. The 1394 Trade association has dubbed IEEE-1394 the MultiMedia Connection. Some of the advantages of 1394 over other buses are described in the following paragraphs.
Versatility: FireWire provides a direct digital link between up to 63 devices without the need for additional hardware, such as hubs. Digital Video (DV) camcorders, scanners, printers, videoconferencing cameras, and fixed-disk drives all share a common bus connection not only to an optional PC, but to each other as well. FireWire is a candidate for the "Home Network" standard initiated by VESA (Video Electronic Standards Association) and other industry associations.

High speed: The present implementation of IEEE-1394 delivers 100 Mbps (Megabits per second) or 200 Mbps of data (payload) and control signals (overhead). Future versions that support 400 Mbps are in the development stage, and a 1.2 Gbps (Gigabits per second) version of IEEE-1394 has been proposed. Isochronous data transmission lets even the lowest-speed implementation support two simultaneous channels of full-motion (30-frame-per-second), "broadcast quality" video and CD-grade stereo audio.

Low cost: The cost of the integrated circuits and connectors to implement FireWire is often less than the cost of the connectors and circuitry it replaces. FireWire uses a flexible, six-conductor cable and connectors derived from Nintendo's Gameboy to interconnect devices. (A four-conductor version of the standard cable is used to interconnect consumer audio/video components.) Use of FireWire for consumer electronics gear, such as camcorders and VCRs, will provide the high-volume market needed to achieve low-cost implementation of FireWire on PCI adapter cards and PC motherboards.

Ease of installation and use: FireWire extends Plug and Play features far beyond the confines of the personal computer. When you add a new device, FireWire
automatically recognizes the device; similarly, on disconnect FireWire automatically reconfigures itself. The standard FireWire cable provides up to 1.5 amps of DC power to keep remote devices "alive" even when they're powered down. You don't need a computer to take advantage of FireWire; as an example, a VCR can act as a FireWire controller for camcorders, TV sets, receiver/amplifiers, and other home theater components.

How IEEE 1394 Works

The IEEE-1394 High Performance Serial Bus is a remarkable engineering feat that has occupied many highly-creative digital circuit designers and software programmers for the past 10 years. FireWire is a very complex serial bus protocol, as evidenced by the hundreds of pages that comprise its standard specification.

Standard cables and connectors replace the I/O connectors employed by consumer electronics equipment and PCs. 1394 wire multiplexes a variety of different types of digital signals, such as compressed video, digitized audio, MIDI, and device control commands, on two twisted-pair conductors. Multiplexing is used in virtually all analog and digital networking systems, but usually only a single type of signal is involved. As an example, Ethernet multiplexes digital data streams from workstations and servers over one (10Base2, "Thin" Ethernet) or two (10BaseT, 100BaseT) pairs of conductors. (1394 cabling is quite similar to that of 10BaseT Ethernet.) Sending real-time, high-quality audio and video data over Ethernet, however, requires special protocols presently implemented only by proprietary multimedia networking systems. 1394 is much more flexible in its accommodation of different data types and topologies than alternative
networking systems. FireWire uses a "fairness" arbitration approach to assure that all nodes having information to transmit get a chance to use the bus; standard Ethernet does not provide this type of arbitration.

![Diagram of FireWire layers]

Figure 2. Layout of 1394 Layers

Special integrated circuit chips implement the 1394 protocol. Like Ethernet and other high-speed digital data transmission systems, FireWire is a layered transport system. The IEEE-1394 standard defines three layers: Physical, Link, and Transaction.

The physical layer provides the initialization and arbitration services necessary to assure that only one node at a time is sending data and to translate the serial bus data
stream and signal levels to those required by the link layer. Galvanic isolation may be implemented between the physical layer and the link layer using optical isolators; with isolation, the chip implementing the physical layer is powered by the bus conductors. Isolation should be provided where three-wire power cords are used to prevent ground loops through the green-wire ground; consumer devices, which use two-wire power cords or wall-wart power supplies, ordinarily don't require galvanic isolation.

The transaction layer is used for asynchronous transactions. The 1394 protocol uses a request-response mechanism, with confirmations typically generated within each phase. There are about five different types of transactions that are allowed. They are simple quadlet read, simple quadlet write, variable-length read, variable-length write, and lock transactions. Asynchronous packets have a standard header with an optional data block. The packets are then assembled and disassembled by the link layer controller. The link layer is described in the following paragraph.

The link layer is the interface between the physical and transaction layer. This layer is responsible for checking received CRCs and calculating this to transmitted packets. Since, the transactions layer does not provide use when isochronous transfers occur, the link layer is directly responsible for the sending and receiving of this type of data. The link layer also does the examination of the header information and will determine the type of transaction that is occurring.

Standard bus interconnections are made with a 6-conductor cable containing two separately-shielded twisted pair transmission lines for signaling, two power conductors, and an overall shield. The two twisted pairs are crossed in each cable assembly to create
a transmit-receive connection. The power conductors (8 to 40 V, 1.5 A max.) supply power to the physical layer in isolated devices. Transformer or capacitative coupling is used to provide galvanic isolation; transformer coupling provides 500 volts and lower-cost capacitative coupling offers 60 volts of ground potential difference isolation. Connectors are derived from the GameBoy design and use either a friction detent (standard) or the special side-locking tab restraints shown in figure 3.

![Figure 3. Layout of 1394 cable design](image)

1394 provides a flexible bus management system that provides connectivity between a wide range of devices, which need not include a PC or other bus controller. Bus management involves the following three services:

- A cycle master that broadcasts cycle start packets (required for isochronous operation)

- An isochronous resource manager, if any nodes support isochronous communication (required for DV and DA applications)

- An optional bus master (usually a PC adapter, but an editing DVCR might act as a bus master)
On bus reset, the structure of the bus is determined, node IDs (physical addresses) are assigned to each node, and arbitration for cycle master, isochronous resource manager, and bus master nodes occurs.

Consumer audio/video applications use logical "plugs and sockets," which are analogous to the physical RCA phono jacks and mini-DIN S-video connectors used by TV sets, VCRs, camcorders, receivers, amplifiers, and other audio/visual components. A "plug" corresponds to an audio or video output and a "socket" represents an input connector. The implementation of logical plugs and sockets is defined by the pending Digital Interface for Consumer Electronic Audio/Video Equipment specification, an extension to the IEEE-1394 standard proposed by members of the Japanese Digital Video Consortium (DVC), which is responsible for establishing the consumer DV standard. The Digital Interface specification has been prepared by the DVC for submission to ISO/IEC (International Standards Organization/International Electrotechnical Committee), rather than the IEEE.

**Explanation of RF Link**

The main task for this semester’s 4040 group is to create a working 1394 RF link. The challenging part of this task is the data conversion between the 1394 protocol and the RF transmission signal. The RF link requires an input of eight bit serialized digital data. While, the 1394 protocol provides a serialized digital stream of data that is unrecognizable to the link. The two data types cannot be directly linked without some type of conversion.
After several talks with Texas Instruments we have determined a topology that will allow a solution to the RF link. A topological layout of the design located in the figure below:

![Diagram of RF link topology]

Figure 4 Block Diagram for RF link

The 1394 camera array outputs data via 1394 cable. This data is then feed into our 1394 to digital converter. The converter uses a Texas Instruments TSB41LV03A physical layer chip to capture events off the 1394 cable. The physical layer chips are designed to work in conjunction with a link layer chip. We chose to use the Texas Instruments TSBL12LV42 link layer chip. The main reasons for choosing this particular combination of chip sets are due to the compatibility designed into these chips, as well as the multimedia nature inherent to the TSBL12LV42 link layer chip. A schematic of the
connections between the link layer and physical chip is located in the figure 5. This figure also shows the necessary biasing to configure the chips to work properly.

Figure 5 Schematic of Pin Configuration between Link Layer and Physical Layer Chips

As can be seen in the diagram pins 42 though 46 will be the input to the converter from the cable. The physical layer will subsequently convert the signals into another format to be transferred to the link layer chip. This event corresponds to pins 1,2,3-8,10,11,22 on the physical layer chip and pins 11, 35-42, 44 on the link layer chip. Once the signal is transferred to the link layer chip we are able to access the digital data needed by the RF link. Pins 50-59,80, 78, 76, 74, 71, 69, 64, 62, 79, 77, 75, 73, 70, 68, 68, 61 are used by the micro-controller to handle the events of the chip which will be explained under the “Micro-Controller Operation” section.
Before we implant the complete 1394 RF transmission system we intend to see if we can implement a working link between the 1394-to-Digital link layer chip and the Digital-to-1394 link layer chip. Figure 7 shows the basic configuration for this intermediate evaluation of the design.

![Figure 6 Block Diagram for Link Layer to Link Layer Chip Configuration](image)

The diagram below shows the pin out configuration between the two chips.

![Figure 7 Detailed Pin Out Configuration between the two Link Layer Chips](image)
The digital data is handled in two different portions of the link layer chip. The first area is designated as the “Bulky Data Output” (BDO), and the second area is designated as the “Bulky Data Input/Output”. Each of these portions contains clock rates, frame rates, formatting bits, availability bits, enable bits, and finally data lines. The list below summarizes the connections between the chips:

- The data lines from the 1394-Digital link layer’s BDO data lines, will be tied to the Digital-1394 link layer’s BDI data lines.
- The frame rate will be outputted from the 1394-Digital link layer’s BDO_FR pin into the Digital-1394 link layer’s BDI_FR.
- The available data pin on the 1394-Digital link layer’s BDOAVAIL pin will be tied to the BDIEN on the Digital-1394 link layer chip’s pin. This will cause the Digital-1394 BDIO pins to become enabled anytime the 1394-Digital BDO pins have data to send.
- Both clocks (BDOCLK, BDICLK) will be tied to some external clock source.
- The formatting pins have yet to be determined.

Once the performance of this non-RF link is verified, the data lines will be tied into an RF transmission and receiver system. The data lines from the RF link will need to be tied into a multiplexer, since the number of lines exceeds the number of lines the RF transmitter can handle. In addition the bits of data will need to be serialized for RF transmission, hence, a serialzer will be used. A diagram of the RF transmission Block is located below in Figure 8.
Figure 8 Detailed view of RF Transmission Block

Similar to the “RF Transmitter Block”, the “RF Receiver Block” utilizes a deserializer and a demultiplexer in order to change the data back into a suitable format for the link layer chip. The layout for the RF Receiver Block follows below:

Figure 9 Detailed view of RF Receiver Block

PC Board Design

Table 1 describes the parts list for all of the capacitors and inductors needed to connect the link layer chip to the physical chip for one board. Figure 10 shows a detailed view of the board layout. Note that this current schematic does not include the portion dealing with the micro-controller.
<table>
<thead>
<tr>
<th>DESCRIPTION</th>
<th>SUPPLIER</th>
<th>PART NUMBER</th>
<th>PACKAGE</th>
<th>QTY</th>
<th>REFERENCE</th>
</tr>
</thead>
<tbody>
<tr>
<td>22 pF 5% Capacitor</td>
<td>KEMET</td>
<td>C0805C120K5GAC</td>
<td>805</td>
<td>2</td>
<td>C46,C47</td>
</tr>
<tr>
<td>220 pF Capacitor</td>
<td>KEMET</td>
<td>C0805C221J5GA</td>
<td>805</td>
<td>2</td>
<td>C41,C7</td>
</tr>
<tr>
<td>0.001mF Capacitor</td>
<td>KEMET</td>
<td>C0805C102K5RAC</td>
<td>805</td>
<td>17</td>
<td>C8,C9,C10,C11,C12,C13,C14,C15,C16,C17,C18,C19,C20,C22,C23,C42</td>
</tr>
<tr>
<td>0.01mF 10% Capacitor</td>
<td>KEMET</td>
<td>C0805C103K5RAC</td>
<td>805</td>
<td>6</td>
<td>C5,C6,C24,C39,C40,C43</td>
</tr>
<tr>
<td>0.1m10% Capacitor</td>
<td>KEMET</td>
<td>C0805C104K5RAC</td>
<td>805</td>
<td>11</td>
<td>C27,C28,C29,C30,C31,C32,C33,C34,C35,C49,C50</td>
</tr>
<tr>
<td>1 mF50 V capacitor</td>
<td>KEMET</td>
<td>T491A105M016A</td>
<td>805</td>
<td>3</td>
<td>C4,C36,C48</td>
</tr>
<tr>
<td>3.3 mF50 V capacitor</td>
<td>KEMET</td>
<td>T491D335K050AS</td>
<td>2816</td>
<td>1</td>
<td>C1</td>
</tr>
<tr>
<td>10 mF20%25 V Capacitor</td>
<td>KEMET</td>
<td>T491D106K025AS</td>
<td>2816</td>
<td>2</td>
<td>C37,C38</td>
</tr>
<tr>
<td>100 mF20%10 V capacitor</td>
<td>KEMET</td>
<td>T495X107M010A</td>
<td>2816</td>
<td>2</td>
<td>C2,C3</td>
</tr>
<tr>
<td>DIODE</td>
<td>MOTOROLA</td>
<td>MSRS1100T3</td>
<td>1815</td>
<td>1</td>
<td>D1</td>
</tr>
<tr>
<td>DIODE</td>
<td>MOTOROLA</td>
<td>MBRS340T3</td>
<td>2824_dio1</td>
<td>1</td>
<td>D2</td>
</tr>
<tr>
<td>Fuse 0.75 A</td>
<td>RayChem</td>
<td>SMD075-2</td>
<td>2920_Fuse</td>
<td>1</td>
<td>F1</td>
</tr>
<tr>
<td>1394 R/A flat header</td>
<td>MOLEX</td>
<td>53462-0611</td>
<td>Socket_6</td>
<td>2</td>
<td>J1,J2</td>
</tr>
<tr>
<td>680 mH 20% Inductor</td>
<td>TDK</td>
<td>SLF7032T-681MR16</td>
<td>SLF7032</td>
<td>1</td>
<td>L1</td>
</tr>
<tr>
<td>0 W Resistor</td>
<td>KOA</td>
<td>RM7322AT</td>
<td>805</td>
<td>1</td>
<td>R18</td>
</tr>
<tr>
<td>56.2 1% Resistor</td>
<td>KOA</td>
<td>RK73H2AT56R2F</td>
<td>805</td>
<td>8</td>
<td>R1,R2,R3,R4,R7,R8,R9,R10</td>
</tr>
<tr>
<td>5K 5% Resistor</td>
<td>KOA</td>
<td>RM73B2AT102J</td>
<td>805</td>
<td>5</td>
<td>R17,R23,R24,R25,R26</td>
</tr>
<tr>
<td>5.11 K Resistor</td>
<td>KOA</td>
<td>RK73H2AT5111F</td>
<td>805</td>
<td>2</td>
<td>R5,R11</td>
</tr>
<tr>
<td>6.34K 0.5% Resistor</td>
<td>KOA</td>
<td>RM73H2AT6341D</td>
<td>805</td>
<td>1</td>
<td>R15</td>
</tr>
<tr>
<td>10K Resistor</td>
<td>KOA</td>
<td>RM73B2AT103J</td>
<td>805</td>
<td>5</td>
<td>R13,R19,R20,R21,R22</td>
</tr>
<tr>
<td>110K Resistor</td>
<td>KOA</td>
<td>RM73B2AT114J</td>
<td>805</td>
<td>1</td>
<td>R27</td>
</tr>
<tr>
<td>390K 5% Resistor</td>
<td>KOA</td>
<td>RM73B2AT394J</td>
<td>805</td>
<td>1</td>
<td>R14</td>
</tr>
<tr>
<td>1M 1% Resistor</td>
<td>KOA</td>
<td>RK73H2AT105F</td>
<td>805</td>
<td>3</td>
<td>R6,R12,R16</td>
</tr>
<tr>
<td>Voltage regulator</td>
<td>National</td>
<td>LM2574HVM-3.3</td>
<td>So14_464</td>
<td>1</td>
<td>U1</td>
</tr>
<tr>
<td>LT1117CST-3.3</td>
<td>Linear Tech</td>
<td>LT1117-CST-3.3</td>
<td>sot223</td>
<td>1</td>
<td>U2</td>
</tr>
<tr>
<td>1394 3-port 400 Mbps PHY</td>
<td>TI</td>
<td>TSB41LV03</td>
<td>80_Pin</td>
<td>1</td>
<td>U3</td>
</tr>
<tr>
<td>DVLynx</td>
<td>TI</td>
<td>TSB12LV42 100Pin</td>
<td>TQFP</td>
<td>1</td>
<td>U4</td>
</tr>
<tr>
<td>XTAL 24.576 MHz</td>
<td>FOX</td>
<td>FE 24.576 20 pF</td>
<td>XTAL_FE</td>
<td>1</td>
<td>X1</td>
</tr>
</tbody>
</table>
Figure 10 Schematic of Board Layout Top (a) and Bottom (b)
Please refer to Figure 5 for the “1394 RF-Link” board pin out details. As a side note in the bottom board layout; the lower line is the ground, the upper left line indicates the Physical Layer power supply and, the upper right line denotes the Link Layer power supply. In order for a preliminary design to be tested, some alterations to the above board will need to be made. These alterations will be clearly listed in the “Summary Status” of this report.

**DSP Microprocessor Operation Section**

The original design called for the use of a TMS12LV42 DSP. As stated earlier, this chip was unavailable for purchase. Instead, a TMS320C206 was chosen, because it seemed to do the same function. After purchasing the chip and looking through its specifications, it was discovered that this chip may not work. An e-mail was sent to the people we had dealt with at Texas Instruments, and they stated that the TMS320C206 would probably not work, and the one they specified was a different one from the earlier two. They stated that on the evaluation board being sent to Georgia Institute of Technology, a TMS320C50 was to be used. This chip was also unavailable for purchase from any vendors, so the next step was to wait for the evaluation board TI was sending, and try to figure out how it works.

The manual called **DVLynx/MPEG Lynx Evaluation Board Hardware Design Manual** was studied, in order to gain an understanding of how the evaluation board worked. Additionally, the manual was studied to learn how the microcontroller/DSP communicated with the Link Layer chip. It turns out that the process of the microcontroller/DSP communicating with the Link Layer chip was not through a direct connection. There are many other chips and parts that help with the communication, as
shown in Figure 11. There is a 128K x 16 SRAM (for data/program space), a 64K x 16 FLASH (29F100) (Part I/O space, part Program Space), a 16-bit DSP data bus, a 256 x 16 Bidirectional Decoupling FIFO (2 x 74ACT7813), a MACH446 (2 DMA, glue, FIFO control etc), a u-Processor translation interface, a 16550 UART, and a Thru-Hole Prototyping area.

![Diagram of Evaluation Board Block Diagram](image)

**Figure 11. Evaluation Board block diagram**

The DSP in the design is considered the "intelligence" of the board. 64K words (x16b) provides the program storage for the DSP code. A short description of what each part does can be found in the [DVLynx/MPEGLynx Evaluation Board Hardware Design Manual](#), listed in the references section.

After sifting through the manual, and examining the block diagram in Figure 11, it is apparent that this board is not a simple design to emulate.
There are many parts that most students have never dealt with before, so there was great trouble in figuring out how the board worked and what each part did. The manual shows schematics of connections between each of the parts, but figuring out what the part actually is, and how to obtain the part, was a very difficult task. Figure 12 shows the board layout.

Figure 12. Evaluation Board Layout
The future of the 1394 microcontroller/DSP is that the next set of students will have to figure out how this board works, and gain an understanding of the DSP used. After obtaining this knowledge, it may be possible to emulate something similar to the Texas Instruments design, and implement a working 1394 RF link.

**Summary/Status**

Currently the PC board design is being designed and is almost completed. Once completed the design will be outputted to a gerber file and subsequently fabricated. Almost all parts needed to build the board have been ordered or have already been received. The following steps need to be done in order to complete the board:

1. Have PC board fabricated
2. Solder chips, and biasing components to board
3. Connect micro-controller to board
4. Jumper Link Layer chips together form the transmit board to the receive board
5. Verify performance
6. Add multiplexer and serializer chips to board (or separate board with jumpers)
7. Add RF transmission and receive portion
8. Verify performance

In addition several items on how the configuration works still need to be solved.

1. What signals do the format pins need on the link layer chips
2. Clock rate for the link layer
3. How the micro-controller will control link layer chip
4. Add power decoupling networks to all power supply points

5. Add 1394 connector decoupling networks
Appendix
Previous Attempts at Theoretical Solution

What follows below are excerpts and ideas from the original conception phase of this project. In addition, explanations of why these particular ideas were not used are also described. This is meant purely to guide any person wishing to further understand the nuances of creating an 1394 RF link.

Chip Level RF Implementation Solution

The solution, "Chip Level RF implementation", will implement the RF link by separating the Link Layer and Physical layer chips, in order to access the raw data. A layout of the first solution can be seen in Figure 4.

![Diagram of Chip Level RF Implementation Solution]

Figure 4. Chip Level RF Implementation Figure

For this design we plan to connect an analog camera to a NTSC to digital-D1 converter. This signal will then be interfaced to a link layer chip. Typically, the link layer chip is connected to the physical layer chip and subsequently into a 1394 cable. However, for this solution we are instead splitting the link layer and the physical chip and inserting a RF link.
Figure 5. Physical-Link Interface

We will attempt to transmit the eight data bits, D[0:7], the Control line, Clk[0:1], the Link request line, LReq, and the clock signal, SClk, over the RF link. We will ignore all link support lines, since they are inconsequential. Thus, the RF link will be the physical connection between the link layer and physical layer. The twelve signals may need to be multiplexed, in order to be transmitted over the RF link.

As an intermediate step, we will attempt to jumper lines directly from the link layer chip to the physical layer chip. This will allow us to have access to the 12 lines, which will later be connected to the RF link. In lieu of the RF link, we may attempt to incorporate an optical link as a final intermediate step.

This solution could not be realized because of the necessity to keep the physical and link layer devices as a pair. This is attributed to the uncertainties exist as to whether the 1394 line will recognize the physical layer and link layer individually.

Extended Cable Solution

The second solution departs slightly from the first solution. In this scenario we will attempt to implement the RF link in place of a 1394 cable. We shall call this solution the "Extended Cable Ideology". Figure 6 shows the basic configuration of this solution.
One of two options will be used for the camera portion of the circuit. Either a 1394 camera, or a standard NTSC camera in conjunction with link layer and physical layer chips will be used. Both options will ultimately output 1394 protocol data. The 1394 data must be converted into an acceptable format before RF transmission. Once transmitted the RF data must be converted back into 1394. We theorize that the physical layer chip can be used to convert 1394 data to serialized 8 bit digital data. However, we may have issues with the control lines and other link lines. Hence these lines may need to be multiplexed before being transmitted over the RF link.

Similar intermediate steps may be taken as in the first solution. We may jumper the output of the 1394-RF converter to the input of the RF-1394 converter. Also, we may try to transmit the data over an optical link before trying the RF link.

This solution could not be realized due to the inability of using physical layer chips by alone. The physical layer must be used in conjunction with the link layer, mainly for recognition on the 1394 bus line.

The following couple of paragraphs describe the previous microcontroller solution that was derived. This solution was rejected because of the incompatibility of the chip with the necessary tools that were sent to us from Texas Instruments.

Microprocessor Explanation

The microprocessor selected for this design was Texas Instruments' TMS320C206 chip. The main reason for selecting this chip is because it most closely resembled the recommended
microprocessor according to the link layer manual. Table 2 shows some of the parameters associated with this chip.

Table 2.

<table>
<thead>
<tr>
<th>Parameter Name</th>
<th>TMS320LC206</th>
</tr>
</thead>
<tbody>
<tr>
<td>Frequency (MHz)</td>
<td>80</td>
</tr>
<tr>
<td>MIPS</td>
<td>40</td>
</tr>
<tr>
<td>Cycle Time (ns)</td>
<td>25</td>
</tr>
<tr>
<td>Data/Program Memory (words)</td>
<td>64K/64K</td>
</tr>
<tr>
<td>RAM (words)</td>
<td>4.5K</td>
</tr>
<tr>
<td>ROM (words)</td>
<td>32K</td>
</tr>
<tr>
<td>OTP (words)</td>
<td>32K</td>
</tr>
<tr>
<td>DMA</td>
<td>Ext</td>
</tr>
<tr>
<td>Timers</td>
<td>1</td>
</tr>
<tr>
<td>Total Serial Port</td>
<td>2</td>
</tr>
<tr>
<td>Serial Ports</td>
<td>2</td>
</tr>
<tr>
<td>Parallel Ports</td>
<td>64Kx16</td>
</tr>
<tr>
<td>Boot Loader Available</td>
<td>YES</td>
</tr>
</tbody>
</table>

According to the Texas Instruments' manual, the TSBLV42 chip, the Motorola 68000, or the Intel 8051 are the three types of microcontrollers that the link layer chip supports. The Texas Instrument chip was the easiest of the three to connect.

The C2000 DSP core has a static design and is manufactured with a CMOS process for high performance and a low power operation. A single cycle instruction allows for complex math functions to be solved in real time. The memory bus of the C2000 uses an advanced Harvard-type architecture to maximize the processing power by separating the memory bus structures into a program and data section. This allows for the reading of both the data and the instructions at the same time. This architecture allows you to store coefficients in memory to be read in RAM, eliminating the need for a separate coefficient ROM.

The microprocessor interface has multiple programmable functions such as the handshake signal and the type to be used for byte swapping. The microprocessor uses its I/O control register to select which optional function on the interface is going to be operational. The link layer chip supports both 8 bit and 16 bit data busses. The TMSC320C206 central processing unit contains a 16 bit scaling shifter, a 16x16 bit parallel multiplier, a 32-bit central arithmetic logic unit, an accumulator, and additional shifters at the outputs of both the accumulator and the multiplier.
This microcontroller has some features that are worth noting. These are some of the features involved with this microprocessor:

- 32K words of on-chip memory lowers the system cost
- Instruction Cycle Times of 25ns for increased performance
- Static design, power down modes, and PLL options for low-power dissipation
- Source code is compatible with the other DSP chips

The chip also has an enhanced synchronous serial port. This port allows for a throughput of up to 20 Mbps. The bi-directional synchronous serial port provides direct communication with serial devices with serial devices.

Both receive and transmit sides of the synchronous serial port have four level FIFO which allow the CPU to accept an interrupt. This function means less intervention from the CPU, as well as increased flexibility and efficiency with respect to data transfers.

Since the need for a microcontroller with considerable speed and flexibility was necessary for integration with the link layer, the C2000 series DSP was the one best fitted for this operation. The specific details concerning the pin connections and the layout of the chip is discussed in the next section.

The manual for the TSB12LV Link Layer chip specified a TMS12LV42, but this chip is not available, as stated earlier. A TMS320C2000 was purchased instead, which closely resembled the LV42. Most of the pins are the same, except for three pins which are believed to serve the same functions, but have different names. The configuration shown in Figure 11, shows the configuration that will be attempted, except for the CSZ to PS_DS connection, which is still under evaluation as to which connection to use, the PS (program select) and the DS (data select).
Web Site References:

I. Organizations
   A. http://www.1394ta.org
   B. http://www.ieee.org

II. 1394 Chip Manufacturers
   A. http://www.chips.ibm.com/products/1394/
   B. http://www.apple.com/powermac/
   C. http://www.semiconductors.philips.com/home.stm
   D. http://www.ti.com/sc/docs/msp/1394/1394.htm

III. Technical Papers
   A. http://www.skipstone.com/ss21st.html
   B. http://www.skipstone.com/compcon.html

Manuals Referenced:


Received from: David K. Johnson and Burke Henehan. Texas Instruments.