

Teabox: A Sensor Data Interface System

Jesse T. Allison
UMKC & Electrotap
402 Kimberly Dr.
Belton, MO 64012
011.816.331.2453
jta@electrotap.edu

Timothy A. Place
UMKC & Electrotap
606-A East St.
Parkville, MO 64152
011.816.309.8658
tim@electrotap.com

ABSTRACT

Artists have long sought after alternative controllers, sensors, and other means for controlling computer-based musical performance in real-time. Traditional techniques for transmitting the data generated by such devices typically employ the use of MIDI as the transport protocol. Recently, several devices have been developed using alternatives to MIDI, including Ethernet-based and USB-based sensor interfaces.

We have designed and produced a system that uses S/PDIF as the transport mechanism for a sensor interface. This provides robust performance, together with extremely low latency and high resolution. In our system, data from all sensors is multiplexed onto the digital audio line and demultiplexed in software on the computer using standard techniques. We have written demultiplexer objects and plugins for Max/MSP and Jade, as well as a MIDI Conversion program for interapplication uses, while others are in the works for PD, SuperCollider, and AudioUnits.

Keywords

Teabox, Electrotap, Sensor Interface, High Speed, High Resolution, Sensors, S/PDIF

1. HISTORY / DEVELOPMENT

Practitioners of interactive music frequently require sensing devices to obtain gestural control of various signal-processing parameters. While much attention is given to new and novel sensing devices, and to the interactive system itself, the interface that connects the two is often overlooked. This “middle man” which negotiates between the analog sensors and the digital computer is frequently a source of bottlenecks, latency variations, and system instability, not to mention considerable expense. These attributes combine to make such systems inadequate for a wide range of expression and impractical for many artists [1].

The Teabox is an interface developed in an attempt to resolve these difficulties. Our initial experiences with the Teabox have shown it to perform favorably when compared to most commercially available systems.

The current implementation of the Teabox is the fifth generation in a series of solutions utilizing audio hardware to transmit sensor data. The initial solution, an interface housed in a real tea box, converted continuous voltages from two sensors into square waves, using 555 timers, and transmitted them to the computer over two audio lines. The data was represented as the frequency of the square waves. The computer decoded the sensor information by counting the zero-crossings of each signal. This solution worked well. However it inefficiently used bandwidth and quickly filled all available analog audio inputs on a computer.

A second revision improved on this design by replacing the square wave oscillators with sine wave oscillators. The new box collected data from three sensor sources and mixed the sine wave signals they generated onto a single audio line to send to the computer. The computer then separated the signals using band-pass filters and analyzed each signal individually. The data was represented by the amplitude envelope of the sine waves, which was converted into a control signal in the software.

The third approach we took manifested itself in a project we called the SensorBox. The SensorBox accepted 6 sensor inputs and two audio inputs. The data from each sensor was carried as the amplitude of a sine wave which was located in the 18Khz-20Khz frequency range, and piggy-backed on the two audio inputs. This allowed for the sensor data to be transmitted to the computer on the same line as the audio input.

The fourth experiment was to attempt the multiplexing of a number of sensor voltages and sending the data into an analog audio input. However, the drawbacks of this system were ringing and overshoot in the anti-aliasing filters on most audio interfaces. This ultimately led us to a fifth revision that has turned out to be incredibly successful.

Permission to make digital or hard copies of all or part of this work for personal or classroom use is granted without fee provided that copies are not made or distributed for profit or commercial advantage and that copies bear this notice and the full citation on the first page. To copy otherwise, or republish, to post on servers or to redistribute to lists, requires prior specific permission and/or a fee.

Nime '05, May 26-28, 2005, Vancouver, BC, Canada.

Copyright remains with the author(s).

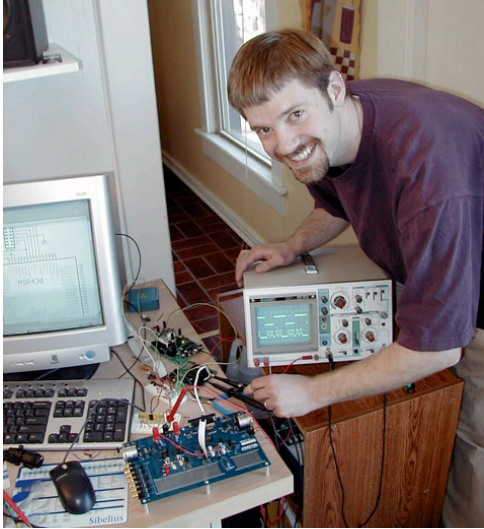


Figure 1. Early development of the Teabox

2. THE TEABOX

The Teabox uses the S/PDIF digital audio format to transmit 24 channels of sensor data via an audio interface to a computer. The Teabox conditions the sensor inputs, samples the voltages, and transmits the data via optical TOSLINK or coaxial RCA cables to an awaiting audio interface. The transmission speed is set to 44.1kHz, but can be modified to 48kHz to accommodate standard video sampling rates. At the computer, the data is normalized and can be accessed and used in Max/MSP through the `teabox~` object, in Jade through the `analysis.teabox` module, and in various other programs via inter-application MIDI or Open Sound Control through the Teabox Bridge program.



Figure 2. Teabox bridge application icon (the original Teabox)

2.1 Connections

The Teabox uses 4 standard Neutrik combo connectors for 4 of the sensor inputs allowing both XLR or 1/4" cables to be used. This allows users to easily make long cable runs to sensors with regular, accessible audio cables. It also provides access to all 8 sensor inputs through an 8 x 3 pin socket allowing you to use sensors built for other platforms. Finally, all 8 continuous inputs are available through a third set of connections via RJ11 (telephone) jacks on the back, allowing you to bundle 4 sensors on one cable.

Aside from the 8 continuous 12 bit sensor channels, there are 16 digital toggles that are accessible through 4 more RJ11 jacks.

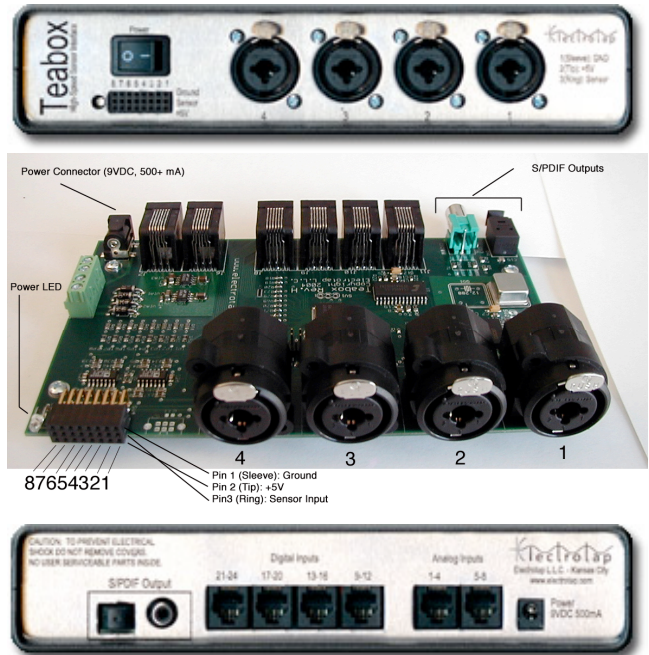


Figure 3. The Teabox Connections

2.2 Resolution

The data resolution sampled by the Teabox is 12 bits, although the S/PDIF transport will allow for higher resolution in the future. In comparison, most MIDI messages are transmitted using only 7 bits. (Up to 14 bits can be used, but it slows down the transmission rate. The resolution of the Teabox is independent of the transmission rate.) A 12 bit resolution gives the device 4096 individual steps compared to 128 for 7 bits.

This high resolution allows for the data to be mapped successfully onto almost any parameter. A good example is mapping a gestural sensor to the frequency of an oscillator. With 7 bits of resolution, the audio spectrum would be divided into 128 quantization steps, or 156 Hz per step. Even mapping this with a logarithmic correction provides a minimum quantization of more than a semitone. Using 12 bits however, the resolution is down to a modest 4.88 Hz per quantization step without using a logarithmic mapping.

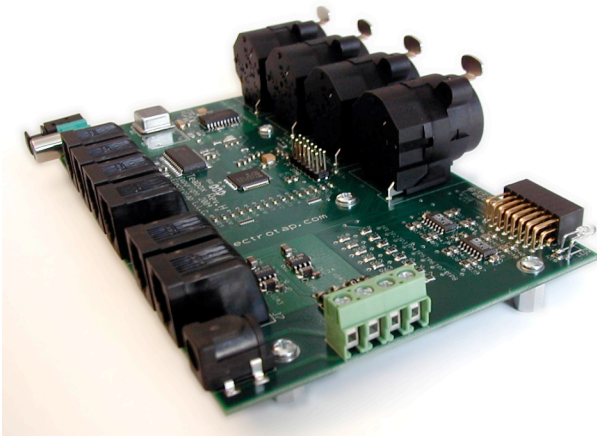


Figure 4. Teabox circuit board

2.3 Audio Rate Transmission

The Teabox uses each sample of the audio transmission to send the data from a single sensor. In practice it takes 11 samples to cycle through all 24 sensors (8 sensors + 2 system messages + 1 16-bit toggle sample). At 44.1k this means all 24 sensors are re-acquired 4009 times per second, e.g. every 0.249 ms. Compare this to MIDI which at best can send 1000 total values per second (1000/8 sensors = 125cycles/s), and the results are astounding. Up to eight continuous controllers and 16 toggles can be used simultaneously without any change in latency.

2.4 Demultiplexing

The sensor data is multiplexed in the Teabox and sent sequentially over the digital audio connection. From there it is straight-forward to demultiplex the audio stream coming in and map the sensor data. Figure 5 is an excerpt from the perform routine of the teabox~ object for Max/MSP.

```
// INPUT SECTION
value = *in++ * 8.0; // Normalize the range for 12-bit
if(value < 0.0 || x->counter > 9){
    x->counter = 0;
}
else if(x->counter == 0){
    x->hw_version = value;
    x->counter++;
}
else{
    x->data[x->counter - 1] = value;
    x->counter++;
}
```

```
// OUTPUT SECTION
*out1++ = x->data[0];
*out2++ = x->data[1];
*out3++ = x->data[2];
*out4++ = x->data[3];
*out5++ = x->data[4];
*out6++ = x->data[5];
*out7++ = x->data[6];
*out8++ = x->data[7];
*out9++ = x->data[8];
```

Figure 5. Code for demultiplexing sensor data from a Teabox

3. SYNCHRONIZATION OF DATA TO AUDIO

In most applications of alternative controllers, data synchronization is paramount. If the controlling gesture is not captured and sent to the computer within milliseconds, it cannot be used to produce an effect that appears to coincide with the event.

The high data rate of the Teabox, combined with the low latency of most audio interfaces, allows for incredible synchronization between gestural stimulus and computer response. Because the data from every sensor is input every 0.25 ms, the audio interface is the only factor in latency. (With an I/O vector of 64 samples it can be reduced to around 1.3 ms with a quality audio interface.)

Another benefit is that any other data that is being input, such as video and especially audio, will have audio and gestural data synchronized to within 10 samples, without any jitter!



Figure 6. Teabox Chassis View

4. CONCLUSION

Sensor data can be sampled by dedicated ADCs, then multiplexed and transmitted over a S/PDIF digital audio connection. This virtually eliminates problems present in other methods of transporting sensor data to a computer, namely problems of resolution, latency, and latency jitter. Additional benefits are gained by connecting the interface with an optical TOSLINK cable. The cable isolates the box from the other circuitry and is

impervious to electromagnetic interference. By eliminating a MIDI-interface dependency, the Teabox also eliminates USB-based reliability problems that the authors have experienced in long-running installations. These considerations make the Teabox not only a viable alternative to existing sensor interfaces, but a much stronger solution for the critical demands of the interactive artist.

5. ACKNOWLEDGMENTS

Our special thanks to Paul Rudy and Jim Mobberley who have encouraged us in this work. We also wish to thank Rich Place for his consultations, problem solving, and direction on the Teabox project.

6. REFERENCES

- [1] Allison, J., and Place, T. "Practical Interfaces for Gestural Sensing Devices." Presented at the 4th annual Electronic Music Midwest Festival, Romeoville, Ill. December 7, 2002.
- [2] Allison, J., and Place T. "SensorBox: Practical Audio Interface for Gestural Performance." NIME-03 Proceedings, 208–209.
- [3] Loy, G. "Musicians Make A Standard": The MIDI Phenomenon. Computer Music Journal Vol. 9 no. 4 (Fall 1985). Cambridge, MA: The MIT Press.
- [4] Wessel, D. and M. Wright. "Problems and Prospects for Intimate Musical Control of Computers." Computer Music Journal Vol. 26 no.3.