

Sample-Rate Conversion for Non-Clocked Audio Sources

Bill Whikehart
Visteon Electronic Systems Div.

ABSTRACT

In a connected vehicle, digital audio content may arrive at the audio system from sources that provide packets of audio samples with no associated sample clock. A common example is streamed internet audio. To process and play these sources, the audio system must first convert the audio stream so that it is compatible with the system's sample clock. Traditional sample-rate conversion methods are typically not capable of doing this. This paper presents and compares approaches for performing the conversion.

CITATION: Whikehart, B., "Sample-Rate Conversion for Non-Clocked Audio Sources," *SAE Int. J. Passeng. Cars – Electron. Electr. Syst.* 7(2):2014, doi:10.4271/2014-01-0261.

INTRODUCTION

When digital audio signals are processed in an automotive audio system, the sample-rate of the audio signal usually must be converted to the sample-rate of the system. The particular problem of converting the sample-rate of non-clocked digital audio signals is described. A non-clocked digital audio signal is one that has no associated physical clock that marks the specific point in time for which each sample is defined. Existing solutions and their shortcomings are discussed. To address these shortcomings, a two-stage conversion approach is then presented.

AUDIO SYSTEMS, AUDIO SIGNALS AND SAMPLE-RATES

Figure 1 shows a typical automotive audio system. Digital audio signals enter the system or are formed inside the system, the system processes these signals and provides output audio signals to speakers. Equalization is an example of processing performed by the system.

The digital audio signals entering the system will sometimes have sample-rates different from that of the system sample-rate, and therefore the sample-rates of these signals must be converted to the system's sample-rate. The meaning of two sample-rates that are "different" is that they are not derived from the same clock source. Thus, an audio signal entering the system with a sample-rate of 44.1 kHz has a *different* sample-rate than that of the system sample-rate of 44.11 kHz *if* the incoming audio signal's sample-rate is *not* locked to the system's sample-rate. The sample-rates of the incoming audio signal and the system are both nominally 44.1 kHz, but they are not *exactly* the same. They cannot be because they are not derived from the same clock source.

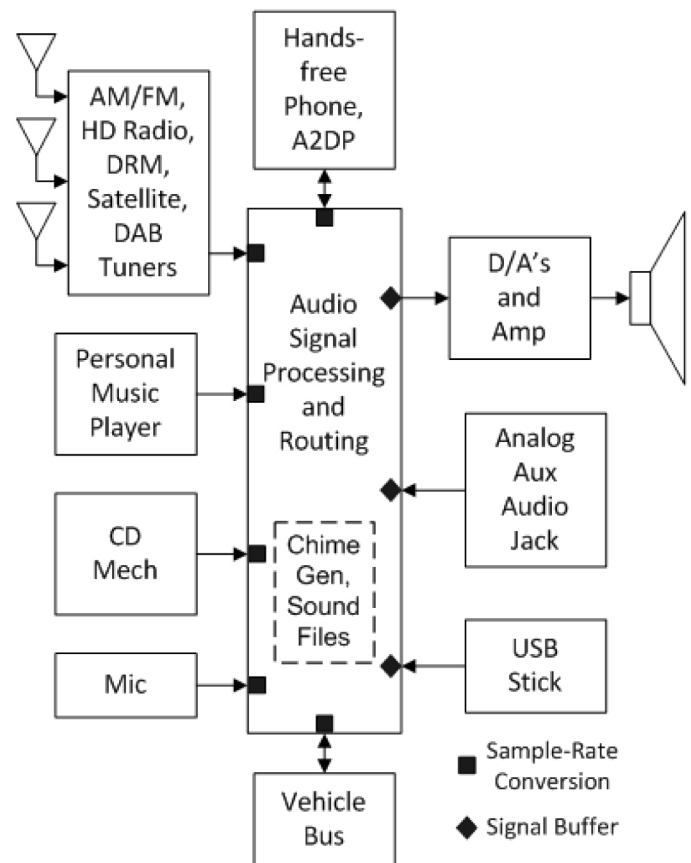


Figure 1. Typical Automotive Audio-Entertainment System

A digital audio signal consists of a stream of samples with an associated sample-rate. When the sample-rate of the incoming digital audio signal is not the same as the system sample-rate, the audio signal must be converted to the system sample-rate. In general, this means that the values of the incoming audio

samples are used to create new values of audio samples that are correct for the instances in time associated with the system sample-rate. This process is called sample-rate conversion. And when the sample-rates of the incoming signal and the system are not locked, this conversion is called asynchronous sample-rate conversion. An asynchronous sample-rate converter (ASRC) converts an input signal at a certain sample-rate to an output signal at a different sample-rate.

Traditional asynchronous sample-rate conversion requires that the ratio of the input sample-rate to the output sample-rate be estimated. The estimate is used to create the output samples from the input samples. The estimate generally must be made continuously, because the ratio's value will drift over time. If it does not drift, it is because the input and output sample-rates are locked together (derived from the same master clock), and in this case *asynchronous* conversion is not required.

DIGITAL AUDIO SIGNAL SOURCES WITHOUT CLOCKS

In many cases, the stream of samples of a digital audio signal has a physical sample clock, which is a clock signal with its frequency equal to the sample rate of the audio signal. If the sample clock is thought of as a square-wave, then each rising edge of the clock marks the precise instant in time for which a sample is defined. Given this physical sample-clock, it is straight-forward to estimate the ratio of the input sample-rate to the system sample-rate, and so a normal ASRC can be used to convert the sample-rate.

However, sometimes a digital audio signal will not have an associated physical sample clock, i.e. it is non-clocked digital audio signal. Examples in the automotive environment are streamed internet audio, audio played from a Bluetooth phone, and broadcast digital audio sources such as DAB in Europe. For each of these sources, the audio content is being pushed by the source to the audio system. In other words, the source is controlling the flow of the audio samples into the system, rather than the system controlling the flow by requesting the samples from the source. It is important to note that the nominal sample-rate of the source may be known by the system, for example from meta-data accompanying the audio signal. However, this nominal sample-rate is just a number, it is not a physical clock.

In most cases, including for the sources given above, a non-clocked audio source will not have a sample-rate that is locked to the system's sample-clock, and therefore asynchronous sample-rate conversion is required. For example, the sample-rate of a streamed Bluetooth audio signal from a phone will be locked to the time-base of the phone. But as discussed already, operation of an ASRC usually requires the availability of a physical clock signal for the source.

Furthermore, the points in time when the samples arrive at the system from a non-clocked source will usually not be precisely periodic. Thus, it is not possible to easily estimate the real-time sample-rate of the source. Information accompanying the source, or information known at system design-time may indicate the nominal source sample-rate, but not the actual real-time rate when the system is in operation. For example, a Bluetooth phone sends audio signals to the audio system in compressed form, and the audio system must decompress (decode) the audio to form an actual stream of audio samples. The compressed audio signal going into the decoder has no sample clock. The decoder just outputs groups of decoded samples as encoded data enters the decoder. And it is likely that the data entering the decoder will not flow at a uniform rate related to the sample-rate of the encoded audio signal, and so there will be variation when each sample exits the decoder. Internet audio and DAB audio decoders operate similarly.

Despite the source digital audio signal having no physical clock, and the likely variations in the arrival times of the audio samples in the audio system, the signal must still be converted to the sample-rate of the system, so that it can be processed and eventually outputted at a uniform rate to a digital-to-analog converter to produce an analog signal to drive speakers.

EXISTING SOLUTIONS FOR CONVERTING THE SAMPLE-RATE OF A NON-CLOCKED AUDIO SOURCE

Estimation of the Source Sample-Rate

For an incoming audio stream with no sample clock, if the system has a means of time-stamping the arrival time, then the time intervals between incoming samples can be calculated, averaged and inverted to provide an estimate of the incoming sample-rate so the sample-rate ratio required to operate the ASRC can be obtained. Practically, the time intervals are measured using a clock that is derived from the same system time-base as the system sample-rate, so the ratio can be directly calculated by appropriate scaling of the inverse of the averaged time intervals. A somewhat more complex method involves using a phase-locked loop (PLL) algorithm in place of the averaging step. This allows the estimated ratio to be constantly updated even if there are large time gaps in the arrival times of the incoming samples. Regardless of whether simple averaging or a PLL is used, a design trade-off must be considered: the faster the averager or PLL reacts to changes in the time intervals, the higher will be the variation of the ratio estimate. And the slower the averager or PLL reacts to interval changes, the less variation the estimate will have, but changes in the intervals may not be tracked as accurately. This trade-off is well known in PLL applications. In addition, when the averager or PLL first starts processing the interval values, it will have a certain settling time before reaching steady-state, when it is tracking the average sample-rate or ratio. The settling-time issue can be somewhat alleviated by using a two-rate PLL as is well known in the art - at start-up, the PLL is operated to react