International Journal of Research in Signal Processing, Computing & Communication System Design Volume 3, Issue 1, 2017
ISSN: 2395-3187

Significance of Sampling Rate Converters in WSNS

P. Swetha[1*], S. Srinivasa Rao[2]
[1]Assistant Professor, Department of ECE, MRCET, Secunderabad, Andhra Pradesh, India.
Email: Swetha1102@gmail.com
[2]Professor & Head, Department of ECE, MRCET, Secunderabad, Andhra Pradesh, India.
Email: ssrao.atri@gmail.com
*Corresponding Author

Abstract: A Wireless Sensor Network (WSN) is a group of specialized transducers with a communications infrastructure for monitoring and recording conditions at diverse locations. A sensor network consists of multiple detection stations called sensor nodes and sampling rate offsets exist between these nodes. As nodes utilize individual clock sources, sampling rate offsets are inevitable and may cause severe performance degradation. Hence Sampling Rate Converter (SRC) becomes very important and integral part of the multi-rate wireless sensor network.

Keywords: Sampling rate offsets, Sensor nodes, SRC, Wireless sensor networks.

I. Introduction

Sample rate conversion, or interpolation and decimation as they are known, are a clever digital signal processing techniques that broadband and wireless design engineers can employ during the system design process. Using these techniques, design engineers can gain an added degree of freedom that could improve the overall performance of system architecture.

Re-sampling is usually done to interface two systems which have different sampling rates. If the ratio of two system's rates happens to be an integer, decimation or interpolation can be used to change the sampling rate (depending on whether the rate is being decreased or increased); otherwise, interpolation and decimation must be used together to change the rate.

Crochiere and Rabiner [1] present a good background for the sampling-rate conversion problem, and describe the "classical" rational-ratio design of implementing the ratio L/M as an up-sampling by a factor of L followed by appropriate filtering, followed by a down-sampling by a factor of M. This method tends to be useful for small values of L and M; otherwise the intermediate sampling-rate tends to get unwieldy.

A practical and well-known example results from the fact that professional audio equipment uses a sampling rate of 48 kHz, but consumer audio equipment uses a rate of 44.1 kHz. Therefore, to transfer music from a professional recording to a CD [2], the sampling rate must be changed by a factor of: (44100 / 48000) = (441 / 480) = (147 / 160).

There are no common factors in 147 and 160, so we must stop factoring at that point. Therefore, in this example, we would interpolate by a factor of 147 then decimate by a factor of 160.

II. Methods of Sampling Rate Conversion

A. Interpolation

"Up-sampling" is the process of inserting zero-valued samples between original samples to increase the sampling rate. (This is called "zero-stuffing".) Up-sampling adds to the original signal undesired spectral images which are centred on multiples of the original sampling rate.

$$y(n) = ([\uparrow L]x(n)) \times h(n) = \sum_k x(k)h(n - Lk)$$

"Interpolation", is the process of up-sampling followed by filtering. In order to remove the undesired spectral images, the following system is used for interpolation.

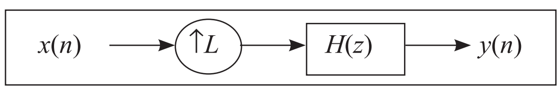

Fig. 1: Interpolation

The combined up-sampling and filtering can be written as
The filter fills zero's that are introduced by the up-sampler. Equivalently, it is designed to remove the spectral images. It should be a low-pass filter with a cut-off frequency $\omega_o = \pi/L$. In this context, the low-pass filter is often called an interpolation filter.

Decimation

The following system is used for decimation.

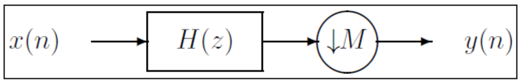

Fig. 2: Decimation

The combined filtering and down-sampling can be written as

$$y(n) = [\downarrow M]\,(x(n) \times h(n)) - \sum_k x(k)h(Mn - k)$$

The filter is designed to avoid aliasing. It should be a low-pass filter with a cut-off frequency $\omega_o = \pi/M$. In this context, the low-pass filter is often called an anti-aliasing filter.

B. Fractional Sampling Rate Conversion

A rate changer for a fractional change (like 2/3) can be obtained by cascading an interpolation system with a decimation system. Then, instead of implementing two separate filters in cascade, one can implement a single filter. Structure for rational rate changer:

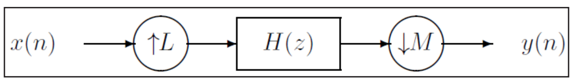

Fig. 3: Fractional SRC

The filter is designed to both eliminate spectral images and to avoid aliasing. The cascade of two ideal low-pass filters is again a low-pass filter with a cut-off frequency that is the minimum of the two cut-off frequencies. So, in this case, the cut-off frequency should be

$$\omega_o = \min\left\{\frac{\pi}{L}, \frac{\pi}{M}\right\}$$

When the ratio between the desired sample rate and the actual symbol rate is an integer, the up-sampling or down sampling process is straight forward. However, there are many applications where the amount by which the discrete time signal must be up-sampled or down-sampled is not always fixed at an integer. In this case, an SRC method capable of handling arbitrary conversion ratios is required.

These all constraints can be answered by the design of the filter H(z). Hence the major challenge tends to be the implementation of the filter H(z) and / or how they update the filter coefficients in order to be as efficient as possible.

C. Performance Limits of Sampling Rate Converter

The performance of the sampling-rate converter algorithm is determined by 5 design parameters:

1. The length L of the sub filters (or equivalently, the length of the prototype low pass filter).
2. The technique (Parks-McClellan, Kaiser, etc) used to design the prototype low pass filter.
3. Allowable pass band and stop band ripple in the prototype low pass filter.
4. The number of sub filters m.
5. The order of the polynomial interpolation.

III. Need of SRC in Wireless Sensor Networks

Wireless Sensor Networks (WSNs) have gained world-wide consideration in recent years, particularly with the proliferation in Micro-Electro-Mechanical Systems (MEMS) technology which has facilitated the development of intelligent sensors. These sensors are small, with limited processing and computing resources. These sensor nodes can sense, measure, and collect information from the environment and, based on some local decision process, they can transmit the sense data to the users. The sensors nodes consist of sensing, data processing, and communicating component, leverage the idea of sensors networks. The components of Sensor nodes are as shown in the Fig. 4.

A sensors network [3] is composed of a large number of sensor nodes that are densely deployed either inside the phenomenon or

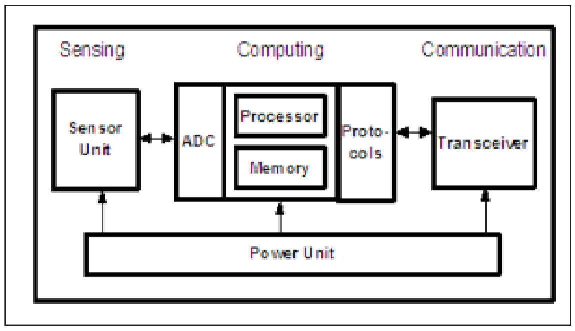

Fig. 4: Components of Sensor Nodes

very close to it. Intelligent sensor nodes are low power devices equipped with one or more sensors, a processor, memory, a power supply, a radio and an actuator. Experimental measurements have shown that generally data transmission is very costly in terms of energy consumption, while data processing consume significantly less [4]. Communication subsystem has energy consumption much higher than the computation subsystem. The energy expenditure of transmitting a single bit of information is approximately the same as that needed for processing a thousand operations in a typical sensor node [5]. Therefore, Communication should be traded for computation in WSNs.

For example, consider a variety of mechanical, thermal, biological, chemical, optical, and magnetic sensors attached to the sensor node to measure properties of the atmosphere. Since the sensor nodes have limited memory and are typically deployed in difficult-to-access locations, a radio is implemented for wireless communication to transfer the data to the base station.

But as nodes utilize individual clock sources, sampling rate offsets are inevitable and may cause severe performance degradation.

For this purpose wireless transceivers are required that need to support various sampling rates in either a receive chain or transmit chain to accommodate different system bandwidths, or different operating bands used in such transceivers. One part of accommodating different bandwidths, such as in baseband processing of a transceiver, is through sampling rate conversion that converts one sampling rate of a signal to another sampling rate. Such conversion could be performed at the output of an Analog-to-Digital Converter (ADC), the input of a Digital-to-Analog Converter (DAC), or any other portions of baseband processing utilizing sampling of signals requiring conversion or adjustment of the sampling rates. Conventional sampling rate conversion (also termed herein as "re-sampling") used in transceivers to accommodate different bandwidths or bands may include integer down-sampling (i.e., decreasing the rate at which a signal is sampled by an integer factor) or up-sampling (i.e., increasing a sampling rate of a signal by an integer factor) or fractional sampling (i.e., changing the sampling rate according to a predetermined fractional value).

As in wireless sensor networks low power consumption is of utmost importance, our concentration can be on efficient structure of SRC which in turn can also influence the power consumption in communication subsystem of sensor node.

IV. Study of Two Sampling Rate Converters

We analyzed the new re-sampling algorithm proposed by Laurent de Soras, in his paper "The Quest for the Perfect Re-sampler" [6] against the most common poly phase sampling rate conversion method. We implemented a fixed point variant of the proposed algorithm with further enhancements to meet the memory and cycle requirements without compromising on the quality.

We used an open source code SoX Re-sampler library as reference for the Laurent de Soras-re-sampler implementation and compared its performance and quality against the benchmarks set by Qualcomm's re-sampler implementation which is based on poly phase sampling rate converter. [For comparison sake in theory and study we used an open source poly phase sampling rate converter – ScopeFIR / libresample].

Comparison of libresample vs SoX library [7] for sine sweep is shown in Fig. 5. and sine tone is shown in Fig. 6. SNR and aliasing effect in this case shows SoX resampler's superior quality.

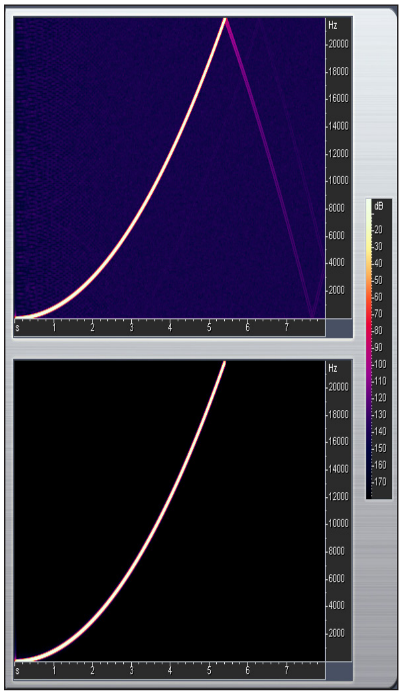

Fig. 5: Comparison of Libresample [Top] vs Sox [Bottom] Library for Sine Sweep-Downsampling

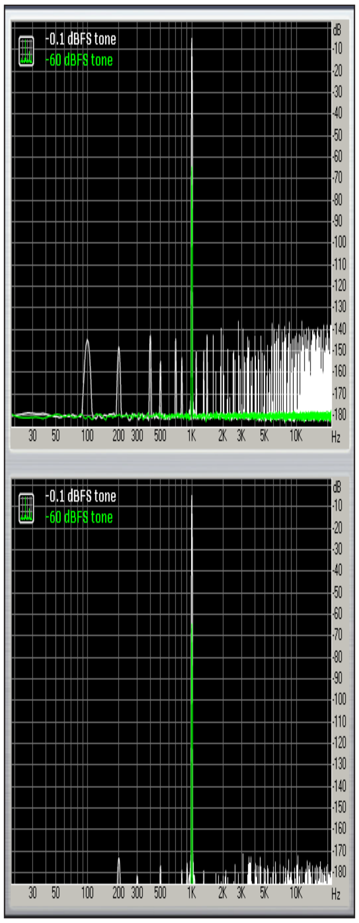

Fig. 6: Comparison of Libresample [Top] and Soxresampler [Bottom] for 1khz Sine Sweep

V. Theory and Results for the Fixedpoint Resampler

The implementation below provides signal evaluation at an arbitrary time, where time is specified as an unsigned binary fixed-point number in units of the input sampling (assumed constant).

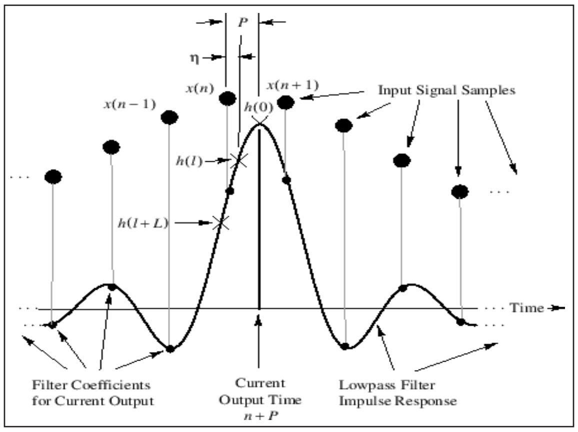

Fig. 7: Sinc Interpolation

In a fixed point re-sampler the time register is divided into three fields: The leftmost field gives the number *n* of samples into the input signal buffer, the middle field is an initial index *l* into the filter coefficient table *h(l)*, and the rightmost field is interpreted as a number *n* between 0 and 1 for doing linear interpolation between samples *l* and *l+1* (initially) of the filter table.

The concatenation of l and n are called *P= (0,1]* which is interpreted as the position of the current time between samples *n* and *n+1* of the input signal.

Let the three fields have n_n, n_l, and n_{nn} bits, respectively. Then the input signal buffer contains $N=2^{n}{}_{n}$ samples, and the filter table contains $L=2^{n}{}_{l}$ ``samples per zero-crossing.'' (The term ``zero-crossing'' is precise only for the case of the ideal low pass; to cover practical cases we generalize ``zero-crossing'' to mean a multiple of time $t_c=0.5/f_c$, where f_c is the low pass cutoff frequency in cycles per sample.) For example, to use the ideal low pass filter, the table would contain *h(l)=Sinc(l/L)*.

The existing reference code from soxr library has the following characteristics

Worst case Signal-to-Noise Ratio: 145.68dB

Measured -3dB roll off point: 96.08 %.

Half length of sinc function: 340238 (Float values).

The fixed point re-sampler based on soxr library for the embedded systems has shorter filter sinc values saving memory. The internal main filter kernel has been optimized to fixed-point arithmetic for faster computation time on DSP or ARM based processors. The quality metrics obtained for the fixed point implementation are as below.

Worst case Signal-to-Noise Ratio: 100.43dB.

Measured - 3dB roll off point: 80.23 %.

Half length of sinc function: 2464 (q31 fixed point format).

Frequency Response for sine sweep tone to 19kHz sampled at 44100Hz upsampled to 48kHz - for the reference soxr library and the proposed optimized version of soxr library.

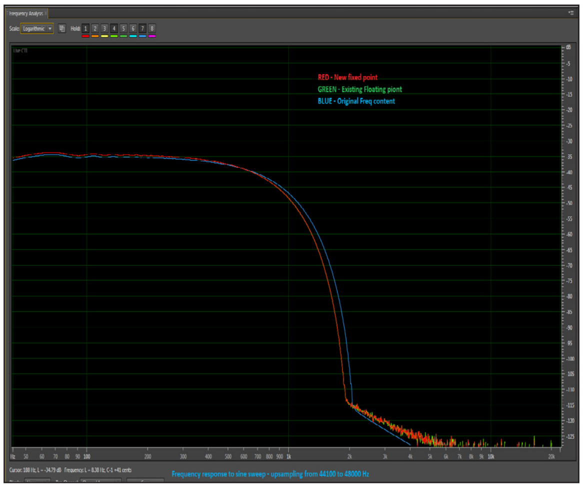

Fig. 8: Frequency Comparison of Sine Sweep Input

Frequency Response for sine tone of 997Hz sampled at 44100Hz upsampled to 48kHz – for the reference soxr library and the proposed optimized version of soxr library.

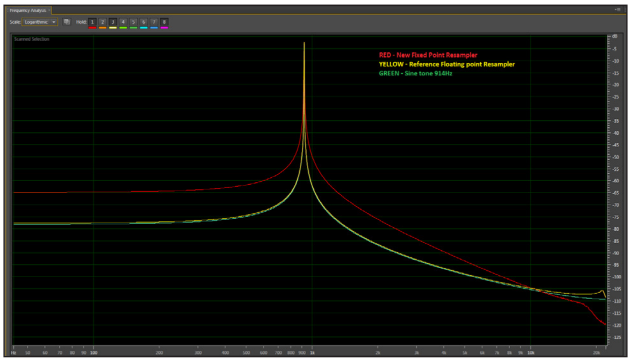

Fig. 9: Frequency Comparison of 914Hz

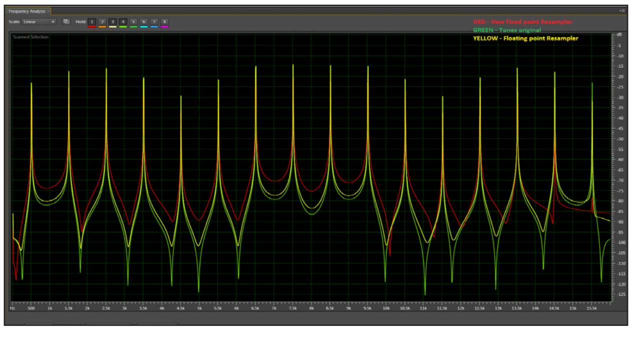

Fig. 10: Frequency Comparison for Multiple Tones

Frequency response to multiple tone inputs up sampled from 32000 to 48000Hz.

VI. Conclusion

In this paper we have reviewed the concept of Sampling Rate Converter and its significance in Wireless Sensor Network. It has been observed that a need exists for more efficient and flexible sampling rate conversion for arbitrary sample rates but with less cost in terms of hardware and power consumption. As a case study comparison of two sampling rate converters has been included followed by simulation results of a fixed point re-sampler, where it is noticed that sinc memory length is efficiently reduced, though due to higher noise floor it doesn't meet the industry standard of 110dB SNR. More tweaking and optimizations would be needed to make it viable solution.

References

[1] R. Crochiere, and L. Rabiner, "Multirate digital signal processing," Englewood Cliffs, NJ: Prentice-Hall, Inc., 1983.

[2] K. Rajamani, Y.-S. Lai, and C. W. Farrow, "An efficient algorithm for sample rate conversion from CD to DAT," *IEEE Signal Processing Letters,* vol. 7, no. 10, pp. 288-290, 2000.

[3] C. Shen, C. Srisaththapornphat, and C. Jaikaeo, "Sensor information networking architecture and applications," *IEEE Personal Communications,* vol. 8, no. 4, pp. 52-59, Aug. 2001.

[4] C. Schurges, O. Aberthorne, and M. B. Srivastava, "Modulation scaling for energy aware communication systems," *International Symposium on Low Power Electronics Design,* pp. 96-99, 2001.

[5] R. Min, and A. Chadraksan, "A framework scalable communication in high density wireless networks," *International Symposium on Low Power Electronics Design,* 2002.

[6] L. D. Soras, "The quest for the perfect re-sampler," 2003. Available http://ldesoras.free.fr

[7] Comparison of various re-samplers. Available: http://src.infinitewave.ca/