

DESIGN AND DEVELOPMENT OF A MULTIRATE FILTERS IN SOFTWARE DEFINED RADIO ENVIRONMENT

L.C Loong, N.C Kyun, C.J Hui and N.K Noordin

Department of Computer and Communication Systems Engineering, Universiti Putra Malaysia, 43400 UPM Serdang, Selangor, Malaysia.

Email: nknordin@eng.upm.edu.my

ABSTRACT

Many well-known digital signal processing (DSP) technologies have been developed for communications, speech processing, radar and sonar systems. One of the basic elements in DSP is sampling rate. Many different sampling rates are used in digital communication applications. Software defined radio (SDR) concept is one of the techniques used to comply with any sampling rate, within a wide range of rates, in order to be compatible with many current protocols. This paper present the multirate filters design using SDR concept. Two SDR models are considered; one with System Generator and one which was absed on Matlab Simulink. The model of System Generator employs sampling rate concept, while the model of without utilizing System Generator employs multirate filter concept. Both models are developed and simulated to produce the desired audio sampling rate for comparison. In the sampling rate concept, a general filter is used to increase and decease the input rate from continuous signal to discrete signal. In contrast, the multirate filter concept uses decimator and interpolator to increase and decrease the input rate respectively. The results show that both models can produce the desired audio sample rate. The later model produces the output faster than the former model but cannot generate Very High Description Language (VHDL) codes. The present of System Generator in Simulink software is to generate the VHDL codes in order to produce a Bit file that can be uploaded to Xilinx FPGA XSV-300 board. Finally, the audio wave file is played from Xilinx FPGA XSV-300 board.

Keywords: Digital Signal Processing, Multirate filter, Sampling rate, Software Defined Radio.

INTRODUCTION

In the past, radio systems were normally designed and built to communicate using one or two modes of signal. If two groups of people have different types of legacy radios they will not be able to communicate. In fact, various systems coexist with different standards in mobile communication and wireless LAN. In transition between 2G and 3G mobile cellular systems, there are coexist many different standards such as Global System for Mobile Communications (GSM), GPRS, IMT-2000, and so on. In wireless LANs, there are standards of IEEE 802.11a, IEEE 802.11b and IEEE 802.11g but widely varying wireless LANs using direct sequence (DS), frequency hopping (FH), orthogonal frequency division multiplexing (OFDM), and other technologies coexist such as Bluetooth. These different specifications result in less interoperability among them. The need to communicate with people using different types of device can only be solved with software reprogrammable radios. One software radio can communicate with many different radios with only a change in software parameters. To satisfy such demands, many manufacturers have been developing various wireless systems with different specifications. Therefore, a software-defined radio (SDR) system with multi-mode and multifunctional capability is urgently needed [1-2].

Many different sampling rates are used in today's digital signal processing (DSP) applications. Telephone systems sampling speech at 8 KHz, 11.025 KHz rate is used in amplitude modulation (AM) radio system, and 44.1 KHz rate is the standard for CD quality digital music. The CD quality digital music with sampling rate of 44.1 KHz has the larger rate than about 40 KHz to fulfill the Nyquist criterion that requires sampling at twice the maximum analog frequency for human ear can hear tones out to about 20 KHz. On the other hand, digital audio tapes (DATs) use a sampling rate of 48 KHz. The sampling rate at 48 KHz is an easy multiple of frequencies used for other common formats, e.g. 8 KHz for telephones.

The problem of converting between different sampling rates has been discussed in numerous technical papers, and different methods have been proposed in [3-7]. A good converter must operate at any sample rate within a wide range of rates in order to be compatible with many protocols. This adaptive control is crucial and it can be implemented either with a hardware link to the converter or in software based controller.

This paper presents the comprehensive knowledge of SDR techniques to design and develop a multirate filter for future advanced radios. Two models are considered; the System Generator model that employs sampling rate concept, while the model of without utilizing System Generator employs multirate filter concept. Both models are developed and simulated to produce the desired audio sampling rate for comparison. This research consists of various phases as follows: introductory theoretical analysis, Simulink simulations, Xilinx System Generator is used to model and implement high-performance DSP systems in field-programmable gate arrays (FPGAs), and testing and troubleshooting the FPGA hardware device.

This paper is organised as follows: in section 2, the overview of SDR is described. The theories of sampling rate is depicted in section 3. The multirate system which exploits decimators and interpolators concept, and resampling concept is explained in section 4. In section 5, the two design models of multirate filter; with and without utilised System Generators are described. The results and performance analysis of such multirate filters are presented in sections 6 and 7 respectively. The paper is ended with a brief conclusion.

SOFTWARE DEFINED RADIO

According to SDR forum, SDR is a radio that provides software control of a variety of modulation techniques wide and narrowband operation, communication security function and waveform requirements of current and evolving standards over a broad frequency range [8]. While the concept of SDR is not new, the rapidly evolving capabilities of digital electronics are making many processes that were once only theoretically possible practical.

SDR also leads to new generations of radio, such as "smart radios" or "cognitive radios" that can observe the utilization of the radio frequency (RF) spectrum in the environment and adjust themselves for optimum performance. The development of SDR leads to universal communication devices. Multi-purpose programmable devices, allows easier usage of communication functions, as a single device that is able to satisfy multiple needs and requirements of the user, as well as providing flexible and advanced features [8].

The key components of SDR are shown in Figure 1, which play vital role to support SDR-based architecture. These key components have been identified as follows:

- Intelligent antenna
- Programmable RF modules
- Digital-to-Analog (DAC) and Analog-to-Digital Converters (ADC)
- Digital Signal Processing Techniques
- Interconnect Technologies

Each block has to provide a considerable amount of flexibility while maintaining performance.

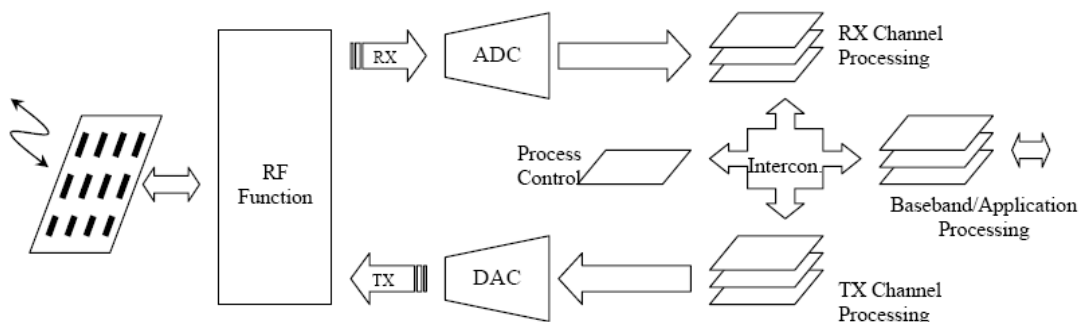


Figure 1: Main components of SDR-based system architecture

SAMPLING RATE

The sampling rate or sampling frequency is defined as the number of samples per second (or per other unit) taken from a continuous signal to make a discrete signal. For time-domain signals, it can be measured in hertz (Hz). The inverse of the sampling frequency is the sampling period or sampling interval, which is the time between samples. The concept of sampling frequency can only be applied to samplers in which samples are taken periodically. Some samplers may sample at a non-periodic rate. Sampling theorem, also called Nyquist theorem is a theoretically sufficient condition to allow an analog signal to be reconstructed completely from a set of uniformly spaced discrete-time samples. Nyquist theorem states that:

$$f_s \geq 2f_M \tag{1}$$

where f_s is sampling frequency of the signal and f_M is maximum frequency of the signal being sampled or also called Nyquist frequency. Aliasing will be occurred when $f_s < 2f_M$. Obviously, aliasing has the effect of producing sounds of lower frequency from sounds that are higher in frequency than the Nyquist frequency. Once aliasing has occurred, it is absolutely impossible to distinguish a component generated by aliasing from one that was actually present in the input signal. This effect is one of the most common sources of distortion in digitized waveforms [9].

MULTIRATE SYSTEM

In recent years there has been tremendous progress in multirate processing of digital signals. Unlike the single-rate system, the sample spacing in a multirate system can vary from point to point. This often results in more efficient processing of signals because the sampling rates at various internal points can be kept as small as possible. There are two basic building blocks in a multirate DSP system: decimators and interpolators concept and resampling concept [10].

Decimators and Interpolators Concept

The decimator is characterized by the input-output relation:

$$y_D(n) = x(Mn) \tag{2}$$

which says that the output at time n is equal to the input at time Mn . Figure 2a shows block diagram of this building block.

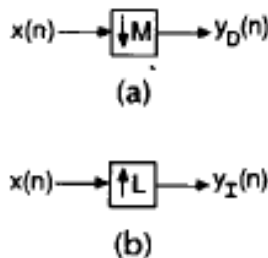


Figure 2: Building blocks, (a) M-fold decimator (b) L-fold interpolator

As a consequence, only the input samples with sample numbers equal to a multiple of M are retained. This sampling-rate reduction by a factor of M is demonstrated in Figure 3 for the case of $M = 2$. The L-fold interpolator as shown in Figure 2b is characterized by the input-output relation:

$$y_I(n) = \begin{cases} x(\frac{n}{L}) & \text{If } n \text{ is a multiple of } L \\ 0 & \text{Otherwise} \end{cases} \tag{3}$$

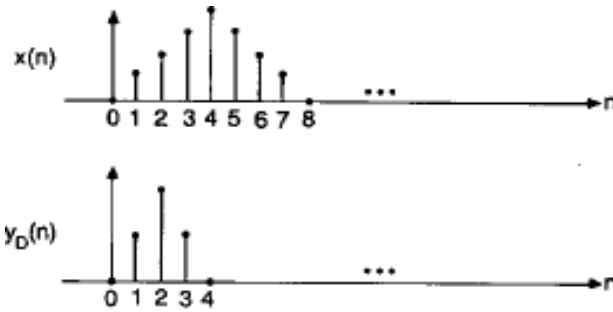


Figure 3: Demonstration of decimation for $M = 2$

That is, the output $y_I(n)$ is obtained by inserting $L - 1$ zero-valued samples between adjacent samples of $x(n)$, as demonstrated in Figure 4 for $L = 2$. The decimator and interpolator are linear systems even though they are time-varying [11].

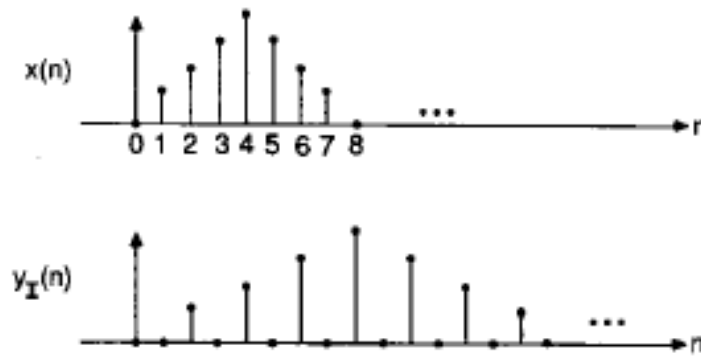


Figure 4: Demonstration of interpolation for $L = 2$

Resampling Concept

Resampling is usually done to interface two systems which have different sampling rates in multirate processing. If the ratio of two system's rates happens to be an integer, decimation or interpolation can be used to change the sampling rates, otherwise interpolation and decimation must be used together to change the rate [12].

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 tape to a CD, 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 the factoring must be stopped at that point. Therefore, in this example, it would interpolate by a factor of 147 then decimate by a factor of 160. The resampling factor of the example is $147 / 160 = 0.91875$.

SYSTEM MODEL DESIGN

Sampling rate conversion method is widely used in most application nowadays included audio sampling rate conversion. Developing the hardware implementation of real-time DSP applications for the sampling rate conversion on the reconfigurable logic platform using Xilinx System Generator and Matlab Simulink is presented here. There are two methods in designing the system model. First, multirate filter concept is used without utilizing System Generator. Second, sampling rate concept is used with utilizing System Generator. The type file of input and output audio signal is in WAV format. Both methods are used for sampling rate conversion to produce desired output sampling rate file. The audio wave file is then played in Xilinx FPGA XSV-300 board. In order to play the song using the Xilinx board, it is necessary to implement in VHDL coding to produce Bit File generated by ISE 9.2i software. Then, the Bit File is used to communicate with XSV-300 board by connecting the parallel port cable between the computer and the board. The GSXload software is used to load the Bit File into XSV-300 board. There are switch buttons embedded in the board used to play the song.

Multirate Filter Concept Without Utilizing System Generator

A multirate filter reduces or increases the input sample rate, resulting in an output rate different from the input rate. The higher the frequency, the more samples need to be evaluated per unit time. A typical use and reason for a multirate filter is to reduce the output frequency of one system to an acceptable value for input by another system. A filter that reduces the input rate is called a decimator. A filter that increases the input rate is called an interpolator. Multirate filter usually wise to break up the filter into several multirate stage, each comprising a multiple of the total decimation or interpolation factor. This process is viewed in Figure 5.

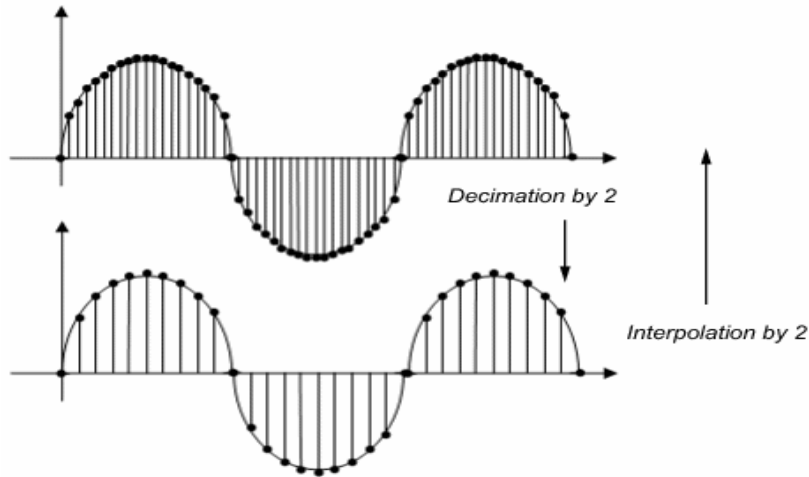


Figure 5: Demonstration of interpolation and decimation

There are many ways to convert a digital audio stream from 48 KHz to 44.1, 22.05, 11.025, 8 KHz. One of the methods is to use three multirate conversion filters in cascade to achieve the required overall fractional resampling ratio, while preserving the signal integrity between 0 and 20 KHz. For sampling rate conversion from 48 KHz to 44.1 KHz, the sequential ratios 8:7, 5:7, and 4:3 are used as shown Figure 6. These ratios are obtained by first noting that the ratio 48:44.1 is equivalent to 160:147. Therefore the three sequential ratios 8:7, 5:7, and 4:3 can be used to achieve an overall ratio equivalent to 160:147 or 48:44.1. The original 48 KHz audio stream is first resampled at 7/8 the rate yielding a 42 KHz sample audio stream. The audio stream is then resampled at 7/5 the rate yielding a 58.8 KHz sampled audio stream. Finally, the 58.8 KHz sampled audio stream is resampled at 3/4 the rate yielding the desired 44.1 KHz output. The possible solutions of sequential ratios in cascaded three-stage sampling rate conversions are shown in Table 1.

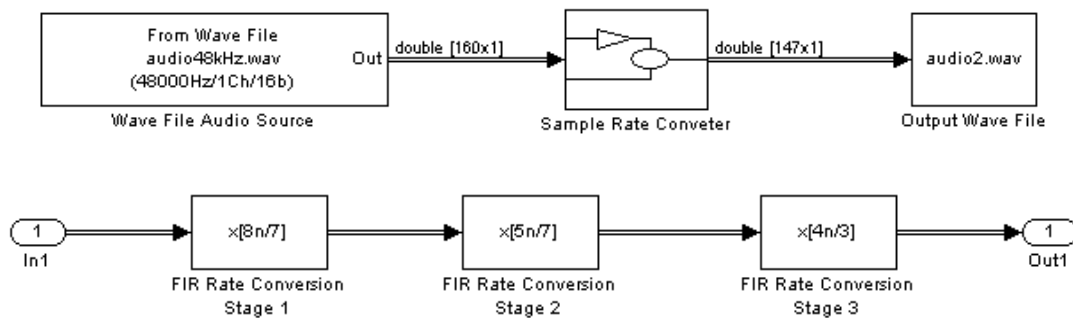


Figure 6: The cascaded three-stage sampling rate conversion from 48 KHz to 44.1 KHz

Table 1: The possible solutions of sequential ratios for sampling rate conversions

Audio Sample rate (KHz)		Overall Ratio equivalent	FIR Rate Conversion		
From	To		Stage 1	Stage 2	Stage 3
48	44.1	160:147	8:7	5:7	4:3
48	22.05	320:147	8:7	5:7	8:3
48	11.025	640:147	10:7	8:7	8:3
48	8	6:1	3:1	2:1	1:1
44.1	48	147:160	7:8	7:5	3:4
44.1	22.05	2:1	2:1	1:1	1:1
44.1	11.025	4:1	2:1	2:1	1:1
44.1	8	441:80	9:8	7:5	7:2
22.05	48	147:320	7:8	7:5	3:8
22.05	44.1	1:2	2:2	1:2	1:1
22.05	11.025	2:1	2:1	1:1	1:1
22.05	8	441:160	9:8	7:5	7:4
11.025	48	147:640	7:10	7:8	3:8
11.025	44.1	1:4	2:2	1:2	1:2
11.025	22.05	1:2	2:2	1:2	1:1
11.025	8	441:320	9:8	7:8	7:5
8	48	1:6	2:3	1:2	1:2
8	44.1	80:441	8:9	5:7	2:7
8	22.05	160:441	8:9	5:7	4:7
8	11.025	320:441	8:9	8:7	5:7

Sampling Rate Concept With Utilizing System Generator

Another method of developing multirate filter for the hardware implementation of complex and real-time DSP applications on a reconfigurable logic platform is by using Xilinx® System Generator. The gateway In and gateway Out blocks shown in Figure 6 can be compiled by System Generator advance’s features. This is shown in an example of Figure 7. In this figure, there are three blocks of input wave file either in one channel (mono) or two channels (stereo). The user can choose one of them to convert into the desired sampling rate. Matrix concatenate blocks arrange the multiple input wave files in an array and submatrix block to multiplex wave file to be sampled by returning the selected portion of input matrix.

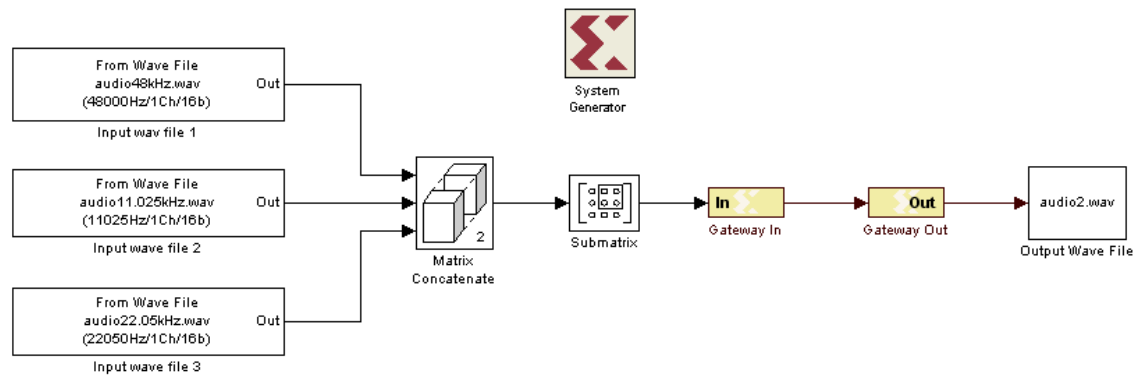


Figure 7: The design model of sample rate conversion with utilizing System Generator blocks

RESULTS

This section will present the results of sampling rate conversion of two models: multirate filter model without utilizing system generator, and sampling rate model with utilizing system generator. The results are compared by the converted file's property of these models. This file property will show the details of file's format.

Design Model Without Utilizing System Generator

In Matlab's window of this model, the information of the audio file is displayed in the command window as shown in Figure 8. The user needs to enter the desired sampling rate for conversion. When the user enters the sampling rate 44.1 or 22.05 or 11.025 or 8 KHz, for example 44.1 KHz, then it is sampling from 48 KHz to 44.1 KHz. Finally the design model diagram is loaded and the necessary variables are assigned to the model.

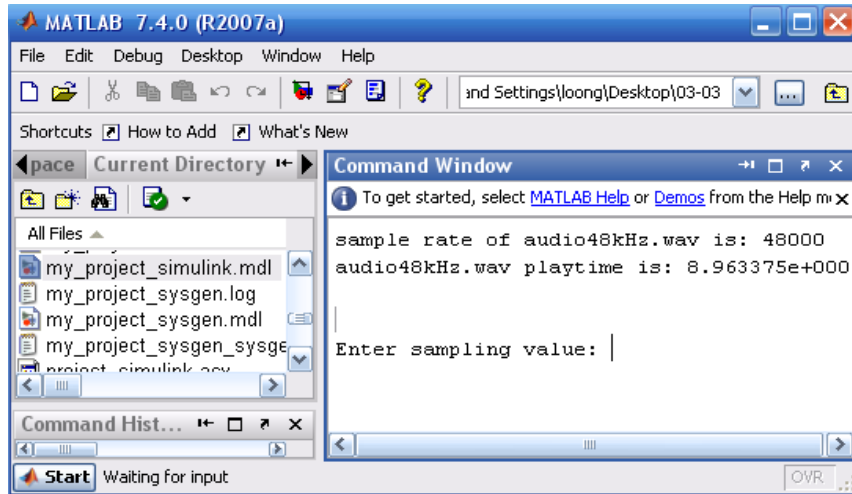


Figure 8: The Matlab's window of without utilizing System Generator model

After the conversion is succeed, the wave file of the desired sampling rate is saved in the current directory of Matlab. The properties of the converted wave file are provoked in order to view the sampling rate of generated audio file. The result is shown in Figure 9. The generated audio sampling rate is 44 KHz which is rounded from 44.1 KHz.

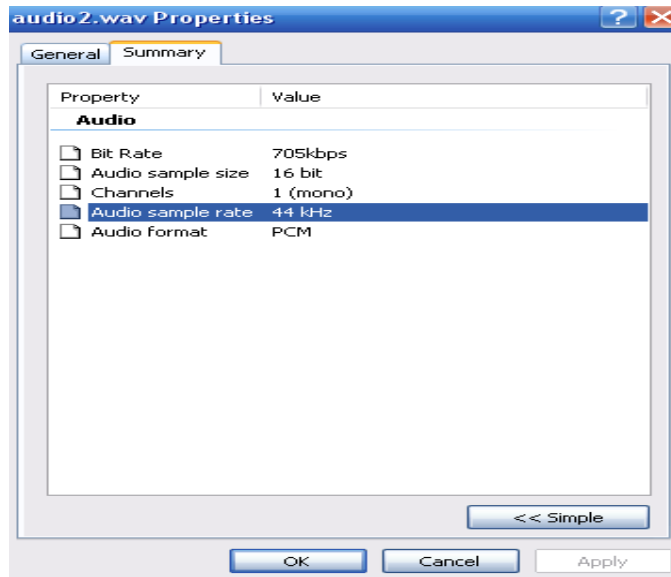


Figure 9: The properties of generated audio wave file in designing model without utilizing System Generator

Design Model With Utilizing System Generator

In the model with utilizing System Generator, the model's block shown in Figure 7 is simulated. The command window displays the three wave file selection options as shown in Figure 10. After the conversion is succeed, the wave file of the desired sampling rate is saved in the current directory of Matlab. The selected option is then being simulated. After the conversion is succeed, again the wave file of the desired sampling rate is saved in the current directory of Matlab. The properties of the converted wave file are shown in Figure 11. In this example, the conversion from 44.1 KHz to 22.05 KHz is preferred.

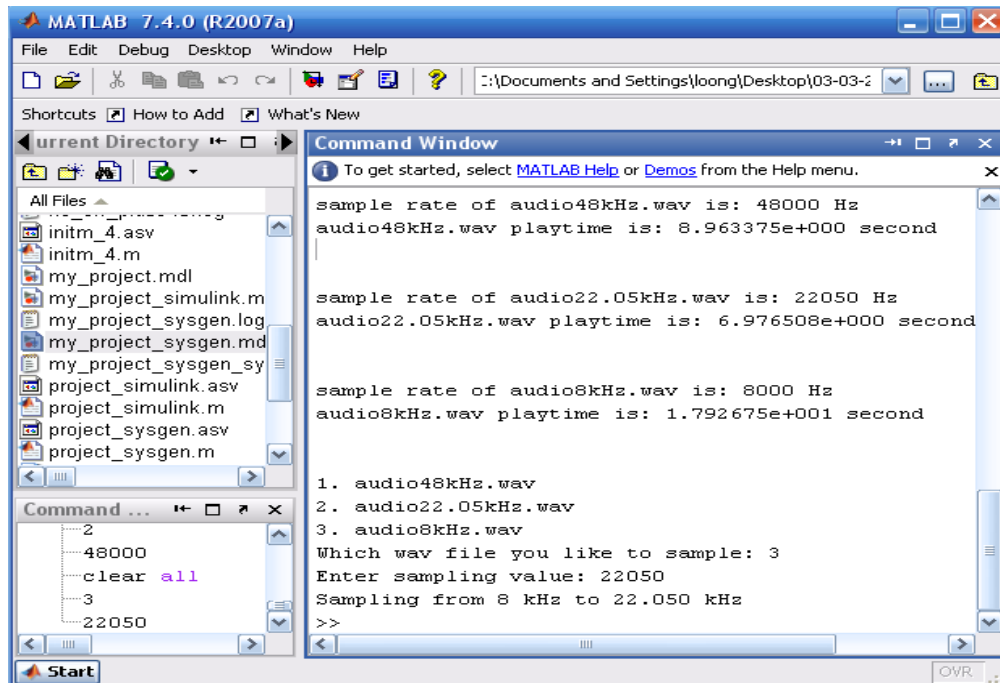


Figure 10: The Matlab's window of with utilizing System Generator model

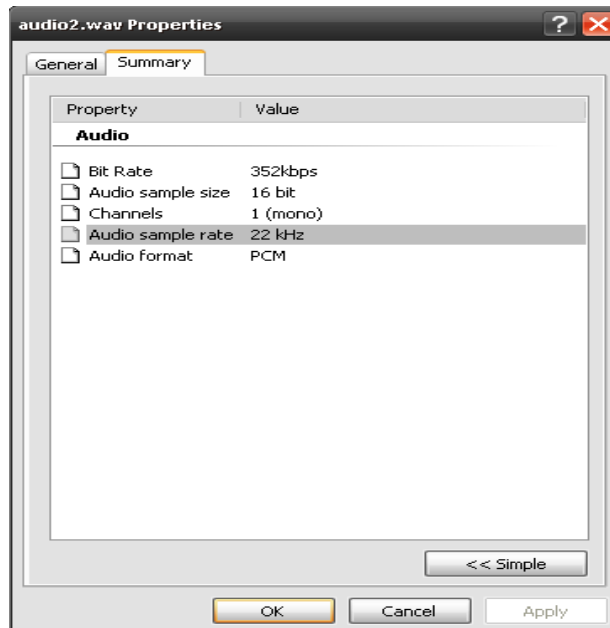


Figure 11: The properties of generated audio wave file in designing model with utilizing System Generator

PERFORMANCE ANALYSIS

Two multirate conversion models have been presented in previous sections. The multirate filter model is only using Simulink and without utilizing System Generator. Thus, it cannot produce VHDL code. On the other hand, the sampling rate model is utilizing System Generator beside simplified with Simulink. This model is producing the VHDL codes in order to generate produce a Bit file that can be uploaded to Xilinx FPGA XSV-300 board. Although both models are able to produce the conversion, it can be seen that the multirate filter model produces the output faster than the sampling model.

CONCLUSIONS

SDR concept is one of the techniques used to deal with multirate conversion. The designs of audio multirate conversion for two models of utilizing System Generator and without utilizing System Generator blocks have been developed. The model of System Generator employs sampling rate concept. A general filter is used to increase and decrease the input rate from continuous signal to discrete signal. In contrast, the model of without utilizing System Generator employs multirate filter concept. This model uses decimator and interpolator to increase and decrease the input rate respectively. After simulating these two models, the desired sample rates of digital audio wave were produced. As a result, both models can produce the desired audio sample rate. However, the multirate filter model that only uses Simulink produces the output faster than the sampling rate model with utilizing System Generator. Only the model that utilizing System Generator can generate VHDL codes. Finally, the audio wave file is played using Xilinx FPGA XSV-300 board.

REFERENCES

- [1] S. P. Reichhart, B. Youmans and R. Dygert, The Software Radio Development System, IEEE Personal Communications, August 1999.
- [2] N. Nakajima, R. Kohno and S. Kubota, Research and Developments of Software Defined Radio Technologies in Japan, IEEE Communications Magazine August 2001.
- [3] R. E. Crochiere and R. Hilson. (1985), *Multirate Filter in Software Defined Radio*, Prentice-Hall.
- [4] J. O. Smith and P. Gosset. (1984), A flexible sampling-rate conversion method, Proceedings, IEEE International Conference on Acoustic Speech Signal Processing, 2: 19.4.1 - 19.4.4.
- [5] S. Cucchi, F. Desinan, G. Parladori, and G. Sicuranza (1991), DSP Implementation of Arbitrary Sampling Frequency Conversion for High Quality Sound Application, International Conference on Acoustic Speech Signal Processing, 5: 3609 - 3612.
- [6] S. Park, G. Hillman, and R. Robles. (1991), A Novel Structure for Real-Time Digital Sample-Rate Converters with Finite Precision Error Analysis, International Conference on Acoustic Speech Signal Processing, 5: 3613 - 3616.
- [7] K. Rajamani, Y. Lai, and C. W. Farrow. (2000), An Efficient Algorithm for Sample Rate Conversion from CD to DAT, IEEE Signal Processing Letters, 7: 288 - 290.
- [8] Blossom, Eric. (2004). GNU Radio: Tools for Exploring the Radio Frequency Spectrum, Linux Journal. <http://www.linuxjournal.com/article/7319>.
- [9] R. E. Crochiere and L. R. Rabiner. (1981), Interpolation and Decimation of Digital Signals: A Tutorial Review, IEEE Proceedings, 69: 300 - 331.
- [10] G. Oetken, T. W. Parks, and H. W. Schussler. (1975), New Results in the Design of Digital Interpolators, IEEE Transactions. Acoustic Speech Signal Proceeding, 23: 301 - 309.
- [11] P. P. Vaidyanathan. (1988), A Tutorial on Multirate Digital Filter Banks, IEEE International Symposium on Circuits and Systems Proceeding, 2241 - 2248.
- [12] Resample Concept <http://www.dspguru.com/info/faqs/multrate/resamp.htm>.