Audio module to capture, store and reproduce sound

Módulo de audio para capturar, almacenar y reproducir sonido

Jason Leitón-Jiménez[1]

1 Instituto Tecnológico de Costa Rica. Costa Rica.
Correo electrónico: jleiton@tec.ac.cr
https://orcid.org/0000-0002-6271-6595

Tecnología en Marcha,
Vol. 34, especial. Noviembre
IEEE Latin American Electron Devices Conference 2021

32

Keywords

Digital filter, frequency , signals, signal sampling ,frequency response.

Abstract

This poster explains the design of an audio module, which is capable of capturing sound signals and storing them for later reproduction. The development platform is an embedded system (Beaglebone black), which has GPIO pins for user interaction. One of the objectives of the module is to keep production costs low, so the electronic components will be chosen taking quality criteria in operation as well as their cost. In the software component, the fdatool tool will be use to design the digital filter, getting the differences equation. It should be noted two major challenges that the development of this project has are the presence of noise in the analog signal and the sampling frequency with which the samples are taken to obtain the digital signal.

Palabras clave

Filtro digital, frecuencia, señales, muestreo, respuesta en frecuencia.

Resumen

Este póster explica el diseño de un módulo de audio, que es capaz de capturar señales de sonido y almacenarlas para su posterior reproducción. La plataforma de desarrollo es un sistema embebido (Beaglebone black), que tiene pines GPIO para la interacción del usuario. Uno de los objetivos del módulo es mantener bajos los costos de producción, por lo que los componentes electrónicos serán elegidos teniendo en cuenta tanto los criterios de calidad en operación como su costo. En el componente software, se utilizará la herramienta fdatool para diseñar el filtro digital, obteniendo la ecuación de diferencias. Cabe destacar que dos grandes retos que tiene el desarrollo de este proyecto son la presencia de ruido en la señal analógica y la frecuencia de muestreo con la que se toman las muestras para obtener la señal digital.

Introduction

LuTec is a laboratory within the Costa Rica Institute of Technology to help learn concepts related to science and technology. Its goals consider that knowledge is a critical factor for any society, and therefore, it promotes new ways to increase, acquire and transmit it [1]. One of the projects of this lab is KROTIC (Costa Rican Robotics Kit). It consists of modular robots with sensors, pins, and components for aiding children to develop their projects. The goal of the project is to motivate children in rural or low-income areas to learn about technology.

Therefore, this paper describes the design and implementation of an audio module for the robots in KROTIC. It looks to facilitate children to use recorded or downloaded sounds in their prototypes to make learning a more exciting way for them since they would be using ears in their senses.

Currently, most devices incorporate multimedia as an essential part of their functionality. One of the reasons is users' attraction when interacting with audio, and video [7]. The audio module project begins with the intention of attracting children's interest in developing their projects, thereby achieving entertained experimentation that generates new knowledge related to science and technology.

The audio module must capture analog signals (audio waves), process them to reduce noise, store them in secondary memory so that recorded sounds or some other audio type can be reproduced later or generated by other means such as songs. It also provides some mechanism for the user to interact with the system. Recordings are stored in secondary memory, in this case, in the embedded system's SD memory.

Due to legal issues, the module must also indicate to the user auditory signals when recorded (notification before starting to record) and visual signals (led on during recording).

KROTIC targets people with limited resources, so the module must be relatively low cost (\$ 15) to increase the total price of the programmable robot.

Background

In 2016, a first attempt was made to incorporate into the KROTIC project an audio module that would capture sound, store it and reproduce it to later transmit it wirelessly, however the objectives were not fully met, since the saved signal it had a very high noise component, so it was not fully intelligible. The research was done by Byron Rojas Valverde [2]. One of the conclusions of the investigation was that the hardware that was being used was very slow to fulfill the requirements, in that case the microprocessor that was used is the ATMEGA328, the great limitation was the loss of data, due to the rate of transmission of these, which implied a deteriorated signal with respect to the original and this was the reason why the sound was not heard clearly.

Most of the signals used in science and engineering are continuous, which is why they are classified as analog, some of them are voltage, electric current, light, temperature, among others. Analog-digital and digital-analog conversions allow today's computers to interact with the above signals. This research is related with processing digital signals and transforming analog signals. It should be noted that digital signals differ from analog signals in that the first ones are sampled and quantified, while the second ones are continuous [4].

When signal conversions are required from the analog domain to the digital domain and vice versa, it is inevitable to depend on the speed with which the arithmetic operations are carried out, mainly multiplication and addition, authors such as [5] mention that the quality of digital filters is directly related to the capacity of the processor, since the process generally involves the implementation of difference equations, which require many mathematical operations. The CPU frequency has an essential role due to it should be the double of the frequency of analog signal.

A typical problem that arises when the sound requires to be manipulated is the noise that the signal can take, so this effect must be counteracted with special circuits and suitable filters, in this case the filter is design in digital domain.

An audio signal processing circuit reduces noise in an incoming sound signal and is particularly useful in telephone communication systems that use one or more microphones. The audio signal processing circuit includes a pre-amplification circuit that receives the audio signal from a microphone or other transducer, an amplifier circuit that receives the pre-amplified audio signal. An output from the pre-amplifier circuit provides the processed audio signal that has an improved signal-to-noise ratio with minimal audible distortion [6]. The amplification process generates a significant increase in the magnitude of the sound signal itself, while in the noise signal the change made by is negligible [3].

Solution Description

The audio module should be capable to capture, process, and play sound, the signals are filtered in analog domain to reduce noisy and digital domain to eliminate some frequency components. It should be noted that it is necessary to eliminate the signals whose frequency exceeds 8000 Hz, since it does not correspond to the spectrum of the most common voice and sounds, so it is not interesting to store this type of audio in memory. The figure 1 shows the information flow chart that is proposed for whole processing of signals. The input requires a specific coupling circuit, since the microphone catches the audio signal with low magnitude. Digital filtering is necessary because this stage is responsible of deleting high frequencies.
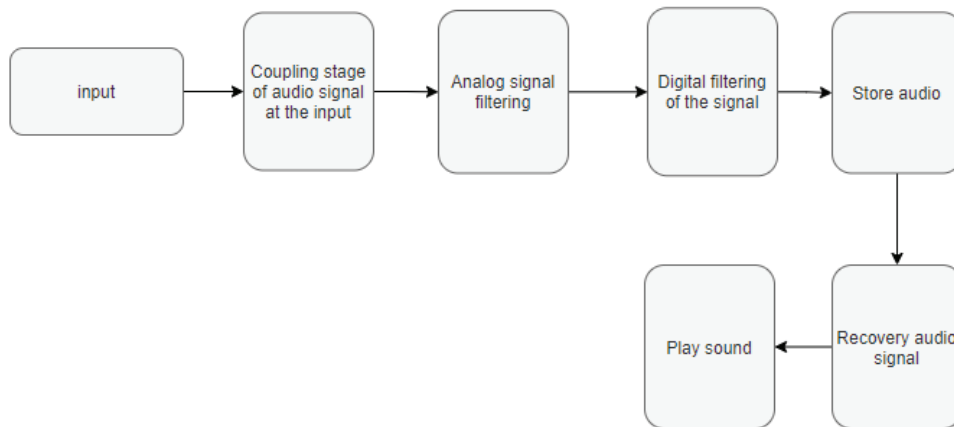
Figure 1. Information flow chart of audio signal.

The audio signal must be treated by different hardware and software modules. figure 2 shows what the process is from when the signal is captured until it is reproduced. It is observed that there is a signal acquisition stage, which corresponds to the microphone, then there is the conditioning stage, whose main function is to eliminate noise in an analog way and control the different physical signals provided by the user.
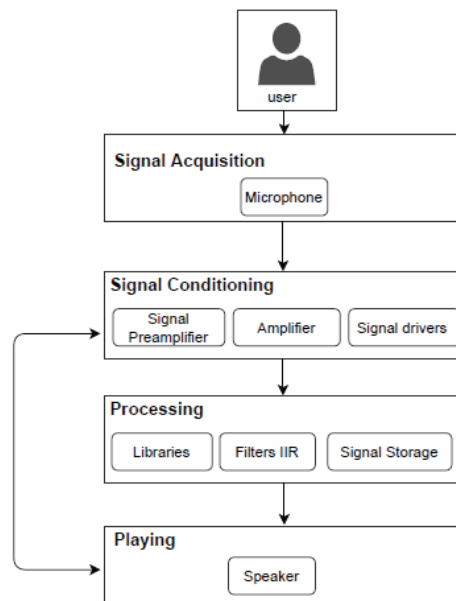
Figure 2. Processing stages.

The processing stage is where digital filtering is performed and unwanted frequencies are removed, in addition to creating the file for storage. Finally, the reproduction stage consists of providing an output system to reproduce the sounds that the user indicates by means of the buttons. This stage is directly related to the conditioning stage, since the signal coming from the embedded system needs an amplification so that it can be heard in an adequate way.

The general architecture of the solution needs some electronic components like a microphone, a coupler and amplifier circuit, an embedded system and an output system, which in this case is a speaker. Each one is related with each processing stage, e.g, output system is related with playing stage.

One of the most important processing stages is recording the signal of sound. The figure 3 shows the algorithm proposed to record and generate the audio file (.wav). The variable "Signal" corresponds to the one that comes from the record button of the circuit, then the fundamental idea is that as long as the user keeps the button pressed, the system will take samples until it is released, once it has been released it proceeds to apply filter to later generate the .wav audio file. The box call "Apply Filter" corresponds to digital filter implemented through differences equation (IIR filter).
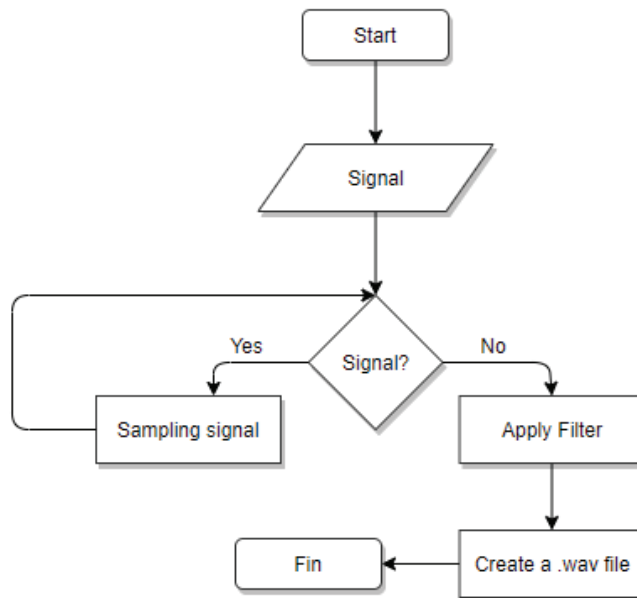

Figure 3. Proposed algorithm to store and recording audio signal.

Filtering implementation is proposed in the time domain, since less processing was needed than doing it in the frequency domain, so a function is required in the code whose objective is to apply the difference equation characteristic of the filter, whose mathematical representation is:

$$y(n) = a_0 x(n) + a_1 x(n-1) + a_2 x(n-2) + a_3 x(n-3) + a_4 x(n-4) + b_1 y(n-1) + b_2 y(n-2) + b_3 y(n-3) + b_4 y(n-4)$$

From the above equation it can be deduced that the filter is of order 4, and therefore it has 5 constants a_n for input $(x(n))$ and 4 constants b_m for data output ($y(n)$).

It should be noted that it is an IIR filter, which needs memory to store past samples, in order to calculate the corresponding output.

References

[1]   Bartolomé Pina, A. R. (1994). Multimedia interactivo y sus posibilidades en educación superior. *Pixel-Bit. Revista de Medios y Educación*, 1, 5-14.

[2]   Byron, Rojas (2016). Modulo de procesamiento de Audio.

[3]   Jiang, X., Song, J., & Chen, J. (2014). U.S. Patent No. 8,705,752. Washington, DC: U.S. Patent and Trademark Office.

[4]   Mitchell, J. L. (2004). Introduction to digital audio coding and standards. *Journal of Electronic Imaging*, 13(2), 399.

[5]   Smith, S. W. (1997). The scientist and engineer's guide to digital signal processing.

[6]   Stroud, R. S., & McQuilling, G. M. (2004). U.S. Patent No. 6,807,280. Washington, DC: U.S. Patent and Trademark Office.

[7]   Valero, C. C., Redondo, M. R., & Palacín, A. S. (2012). Tendencias actuales en el uso de dispositivos móviles en educación. *La educación digital magazine*, 147, 1-21.

Ilustration 1. Presented Poster at LAEDC 2021.