

[Article](#) - [Disclaimer](#) - [About](#)

## 1. Introduction

- 1.1 [Why this project?](#)
- 1.2 [Pitch shifting: traditional methods](#)
  - 1.2.1 [Shift the pitch up](#)
  - 1.2.2 [Shift the pitch down](#)
- 1.3 [Pitch shifting: the digital method](#)
- 1.4 [Latency](#)
- 1.5 [Where to start?](#)

## 1. Introduction

- 2. [Pitch shifting](#)
- 3. [Algorithm](#)
- 4. [Hardware](#)
- 5. [Software](#)
- 6. [Performances](#)
- 7. [Matlab code](#)

## 1. Introduction

This article will explain you how to design a real-time pitch shifter for an electric guitar. In order to build and test my system, you need the following stuff:

- A **DSP starter kit from Spectrum Digital** (as shown below)
- **Code Composer Studio**
- **Matlab**
- Some knowledge regarding pitch shifting (provided in this website!)

Code Composer Studio comes with the DSP starter kit. These kits are really useful and it's worth buying one because you can reuse it afterwards for any DSP project. Every signal processing engineer should also have Matlab to test and perform simulations before programming the DSP chip. If you are a student, you can buy a Matlab license at a very cheap price.



**Figure 1.1:** DSK starter kit for the TMS320C6713 (Photo courtesy of Spectrum Digital, Inc.)

### 1.1 Why this project?

If you play the guitar, have you ever noticed how annoying it is to tune your guitar a few steps up or down to match the song you want to play? In my case it is even worse to tune it down since my guitar's got a floyd rose. This is the main reason why I and two colleagues undertook this project for our final year in Electrical Engineering. The goal was to

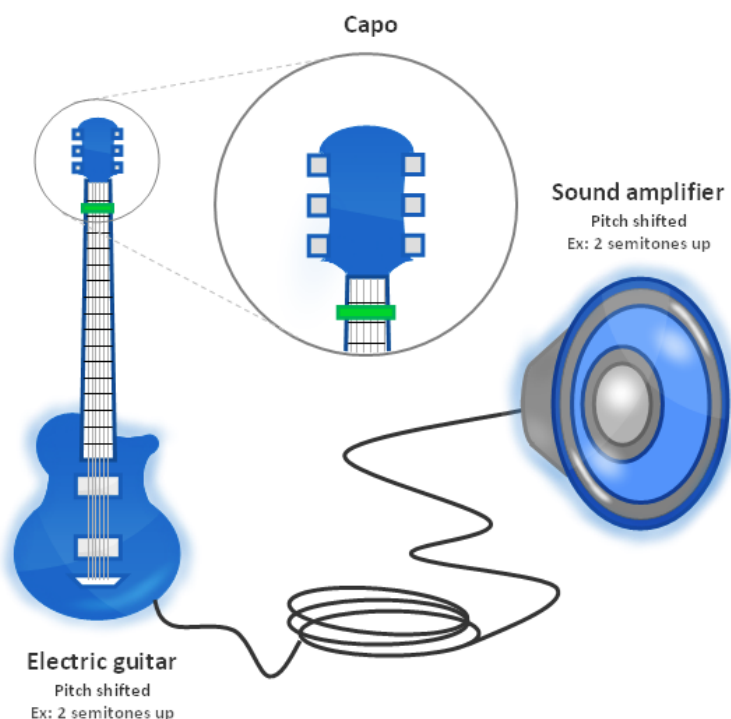
design a real-time pitch shifter that would allow the user to digitally shift the pitch up or down while his guitar stays in standard tuning.

This article will explain in details the different steps we went through in order to achieve this design. I will try to keep the explanations as simple as possible. However, I will assume you have some knowledge of signals and systems theory. Please bear with me: English is not my mother tongue so there might be some mistakes once in a while. Feel free to email me if you have any comments.

## 1.2 Pitch shifting: traditional methods

### 1.2.1 Shift the pitch up

To shift the pitch up, we usually use a capo to lock the strings at a specific spot on the neck. This reduces the length of each string and thus changes its fundamental frequency, which results in a new tuning at a higher pitch. This method works fine except that it reduces the number of frets available for playing.



**Figure 1.2:** Tuning a guitar with a capo

### 1.2.2 Shift the pitch down

In order to shift the pitch down, the strings need to be slacked. The tension in the strings is reduced by turning the tuning keys. This is very annoying because it takes a while and you have to make sure the tension in each string is correct such that the guitar is tuned properly. This is why in most live performances guitarists have many guitars with different tuning. It is in fact faster to pick another guitar than to tune it on the spot. However, for those like me who cannot afford to buy 9 electric guitars, this is not a good solution.

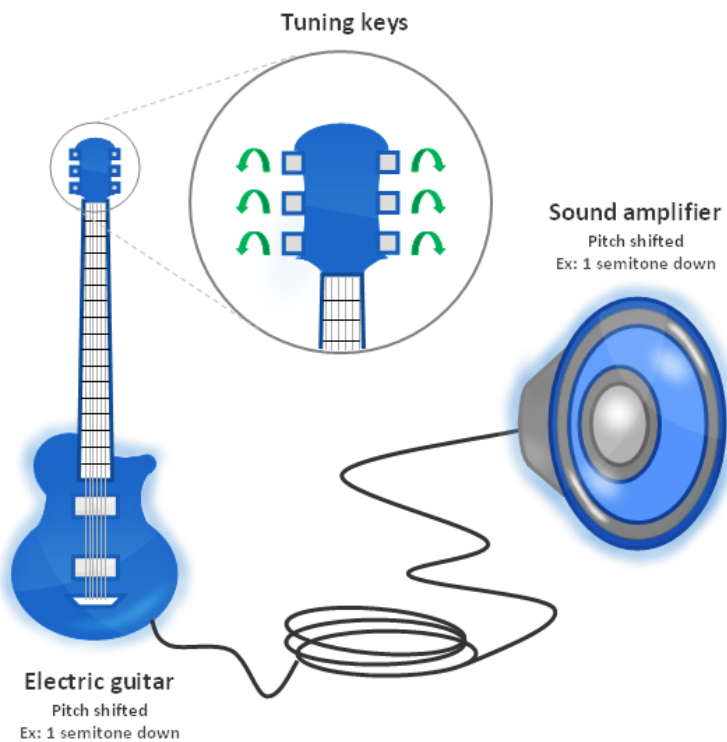


Figure 1.3: Tuning a guitar with tuning keys

1.3 Pitch shifting: the digital method

So let's see what we propose instead. Let's leave the guitar in standard tuning (E-A-D-G-B-E). Before we let the electric signal from the guitar reach the amplifier, we will process it with a Digital Signal Processor (DSP) board to change the frequencies of the note in order to make it sounds like the guitar was tuned differently.

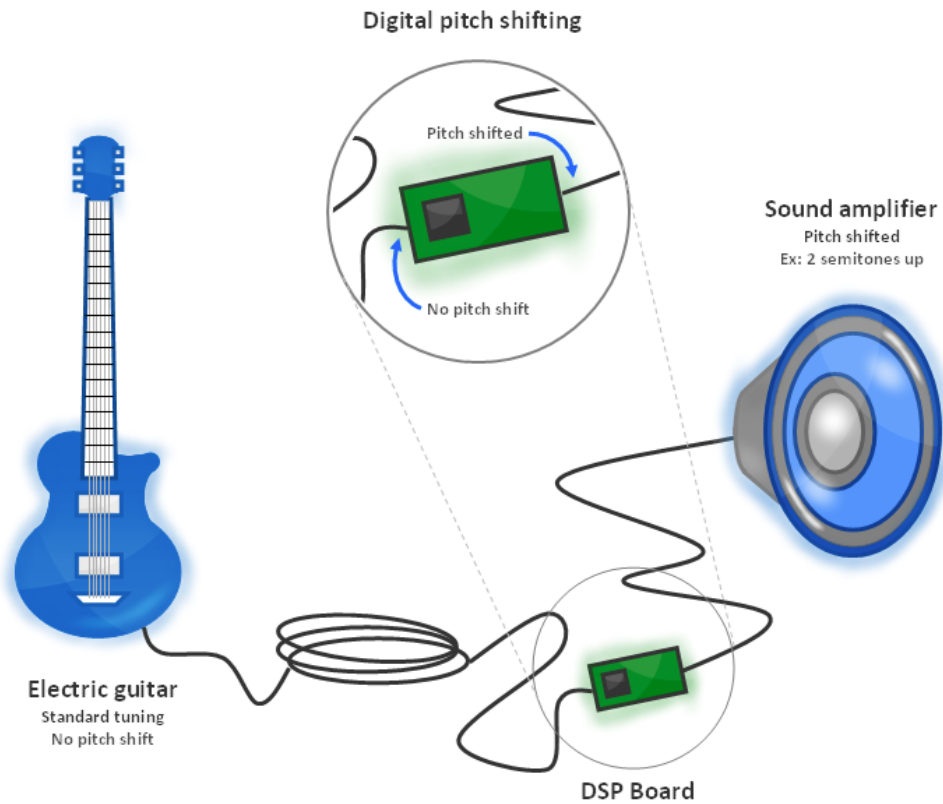
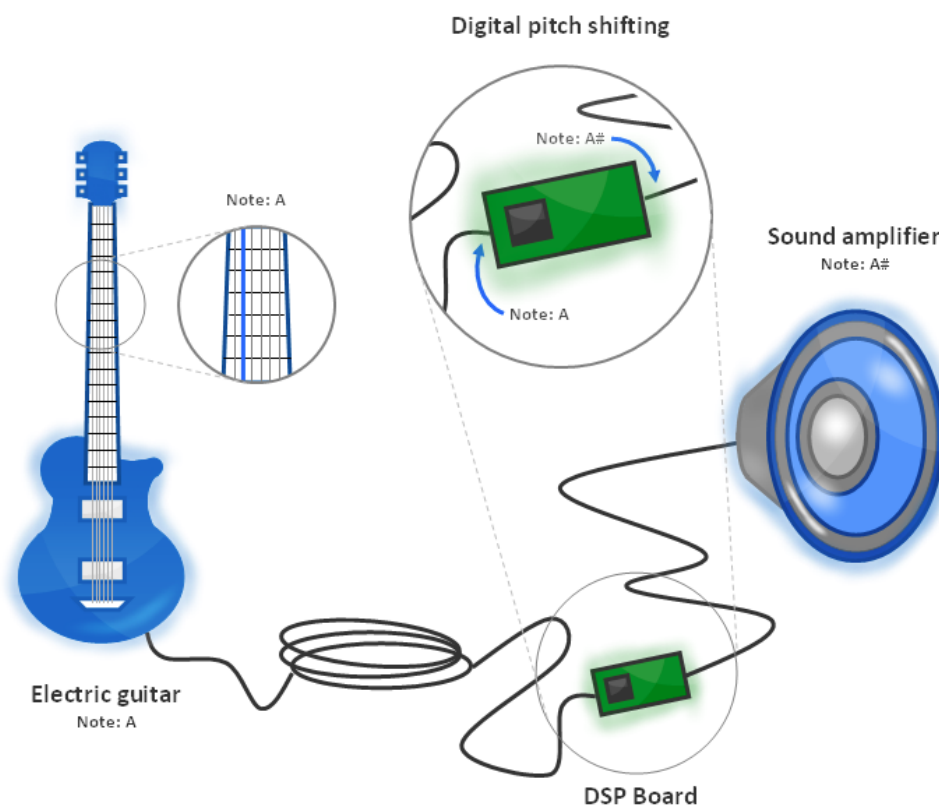


Figure 1.4: Setup for a digital pitch shifter

For instance, suppose we would like the pitch to be shifted up by one semitone. When the player picks the 5<sup>th</sup> string, the note A is produced. The processor takes the electric signal with frequencies that represent the note A. It performs some computations and output a new set of frequencies that now represent the note A#. This new signal is then sent to the amplifier and therefore the note A# is perceived by listeners. From the listener's point of view, it seems that the pitch was shifted up by one semitone. That's a smart trick that works fine, as long as an appropriate algorithm is used in order to preserve the sound quality of the guitar.

Figure 1.5: The way a digital pitch shifter works

## 1.4 Latency

As we design our algorithm, it is very important to keep in mind that the delay resulting from signal processing must be below 50 milliseconds. If we exceed this threshold, the player will perceive the delay. This is extremely annoying and it affects synchronization within the band.

## 1.5 Where to start?

We will first look at the algorithm we designed and tested with Matlab. Then we will look at the hardware we chose to implement the algorithm and the real-time program structure.

But first of all, let's learn a bit more about music theory and pitch shifting.

**Next:** [Pitch shifting](#)