

# **Alphtech DSP - Design Document**

**by Dang Duc Minh - 105559691**

<https://github.com/mikedeelman/Alphtech-DSP>

## **Overview:**

Alphtech DSP is a real-time digital audio processing application that simulates a guitar amplifier with multiple audio effects. The program provides a digital signal processing chain to process live audio input with various effects and EQ controls.

## **Purpose:**

The application serves as a virtual guitar amplifier and effects processor, allowing users to:

- Process live audio input in real-time with minimal latency
- Adjust amplifier settings like volume, gain, 3-band EQ (treble, mid, bass)
- Apply various audio effects (delay, chorus, tremolo, overdrive, distortion)
- Record processed audio input
- Play backing tracks alongside live processed audio
- Select a preloaded guitarist's amp presets, or save and load custom presets
- The UI is well-designed and easy to use, making it more accessible for users to use the application

## **Key Technologies Used:**

### **● Core Framework:**

- **C# .NET:** Primary programming language and runtime environment
- **Window Forms:** GUI framework for desktop application interface

### **● Audio Processing:**

- **NAudio:** .NET audio library for real-time audio I/O operations
- **NAudio.Dsp:** Digital signal processing components (BiQuad filters)
- **File I/O:**
  - **System.IO:** File operations for preset management and audio recording
  - **WAV Format:** Standard audio file format for recording and backing tracks
- **Additional Libraries:**
  - **System.Collection.Generic:** Collection classes for managing effects lists
  - **System.Math:** Mathematical operations for DSP calculations

### Audio Processing Flow:

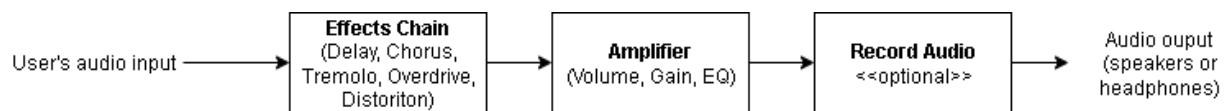


Figure 1: Flow of audio processing

### Program Breakdown:

- **Amplifier:** A digital simulation of a guitar amp that shapes the core sound and tone of a real guitar amplifier.
  - **Volume:** Final output level control (0-10 scale) applied after all processing stages
  - **Gain:** Input amplification (0.1x to 5.0x) that affects signal strength and adds harmonic saturation
- **3-Band EQ System:**
  - Bass (200Hz Low-Shelf): Boosts/cuts low frequencies
  - Mid (1000Hz Peaking): Boosts/cuts midrange frequencies
  - Treble (5000Hz High-Shelf): Boosts/cuts high frequencies

- **Delay:** An effect that records the input signal and plays it back after a set time, creating an echo-like repetition.
  - **Delay Time:** Adjustable from 0.1 to 5.0 seconds
  - **Feedback:** Controls echo repetitions (0-99%)
  - **Mix Control:** Balances dry (original) vs wet (delayed) signal levels
- **Chorus:** A modulation effect that thickens the sound by duplicating the input signal, slightly delaying and pitch-modulating it using an LFO (Low-Frequency Oscillator), then blending it back with the original. This creates a rich, ensemble-like sound.
  - **Rate:** LFO speed (0.1-5.0 Hz) from gentle wobble to vibrato effects
  - **Depth:** Delay time variation (0-10ms) controlling modulation intensity
  - **Base Delay:** Center delay time (1-20ms) around which modulation occurs
  - **Mix:** Wet/dry balance control that blends the original signal with the chorused signal
  - **Feedback:** Routes delayed signal back into delay buffer for enhanced resonance and complexity



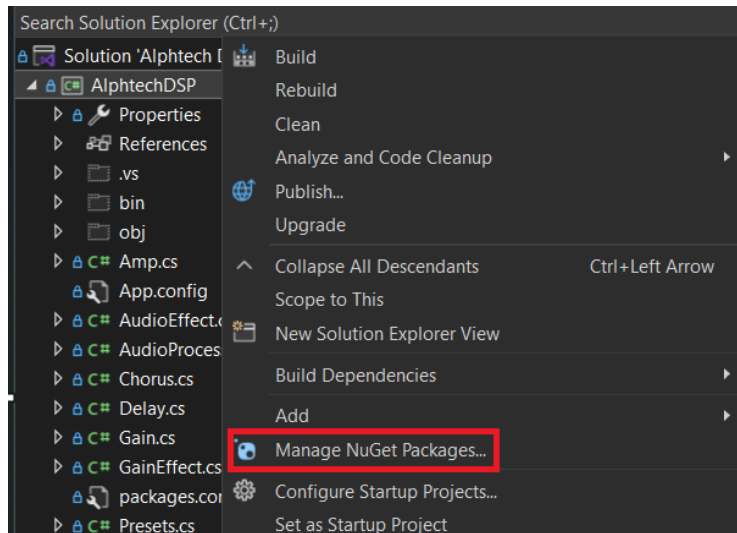
Figure 2: Example of LFO sinewaves

- **Tremolo:** A dynamic effect that modulates signal amplitude using LFO, producing rhythmic volume fluctuations without affecting pitch.
  - **Rate:** Adjusts the speed of modulation (0.1–20.0 Hz)
  - **Depth:** Modulation intensity (0-100%) controlling volume variation amount
- **Overdrive:** Adds gentle saturation by boosting the signal and applying soft clipping, simulating the sound of a cranked tube amp.
  - **Gain Range:** 1.0× to 20.0× for varying levels of drive.
- **Distortion:** Produces heavy, aggressive saturation by boosting the signal and applying hard clipping, resulting in a more intense and compressed tone.
  - **Gain Range:** 4.0× to 23.0× for high levels of distortion.
- **Record:** Captures processed audio in real time, saving WAV files with timestamped names in a dedicated folder.
- **Backing Track:** Plays WAV audio files alongside live input with volume and play/pause controls.
- **Preset:** Choose a preloaded guitarist's amp preset, or save a custom preset to a text file and load it for quick recall of amp settings.

### **How To Run Program:**

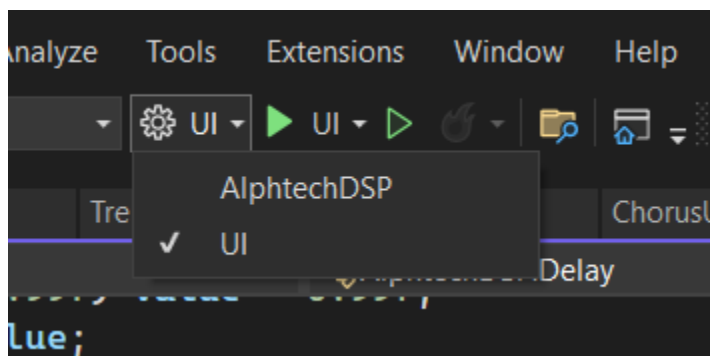
**Step 1:** Open the **Alphtech DSP** solution with **Visual Studio**

**Step 2:** Right-click on **AlphtechDSP** project and select “**Manage NuGet Packages**”



**Step 3:** Browse for **NAudio** by **markheath** and install it

**Step 4:** After installation is done, in the debugger, choose **UI** and run



**Step 5:** Rock and Roll!

**UML class diagram:**

