

# テキスト演奏ソフトウェアシンセサイザーの制作

渡邊陽平

令和7年10月5日

## 1 はじめに

### 1.1 テーマと動機

近年の音楽制作ではシンセサイザーが欠かせない物となっているが、その内部構造や音作りの原理はブラックボックス化している。本研究では、減算式シンセサイザーの機能の一部であるオシレータやフィルタと言った音響合成の基本要素を自力で実装し、その過程を通じてデジタル信号処理の基礎を深く理解することを目的とする。

本ソフトウェアはテキストファイルを利用して自動演奏を実現している。これは、自動演奏式の方が演奏技術よりもシンセサイザーの構造に関心を持ちやすくし、理解を深める目的がある。

### 1.2 技術的仕様

- 音声出力部
  - 言語:C(C17 規格)
  - 役割:PCM 信号を音声として出力する。低レベルなシステム IO やライブラリとの連携が容易であるため採用した。
  - コンパイラ:gcc 12.2.0
  - 主要ライブラリ:PulseAudio クライアントライブラリ 16.1
- 音響合成部
  - 言語:Fortran 2003
  - 役割:オシレータやフィルタを始めとした全て音響合成処理を行う。信号処理に最適化された言語特性から採用した。
  - コンパイラ: GNU Fortran 12.2.0

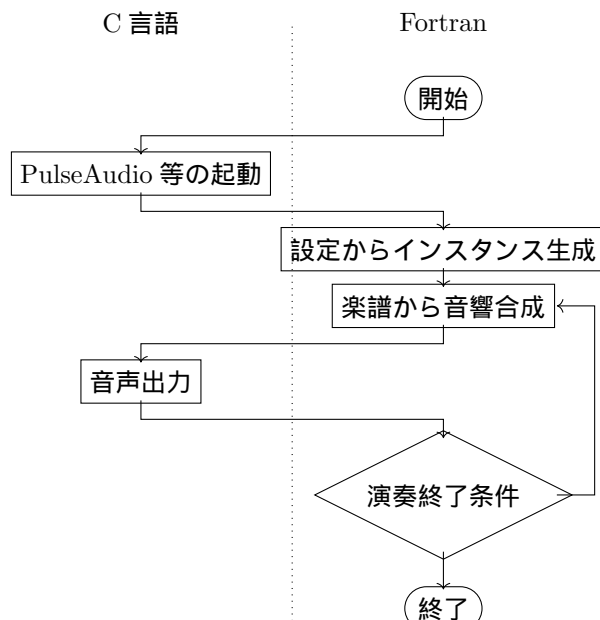
### 1.3 機能要件分析

本ソフトウェアが実現すべき機能要件は研究目的に基づき、以下の項目に定義する。

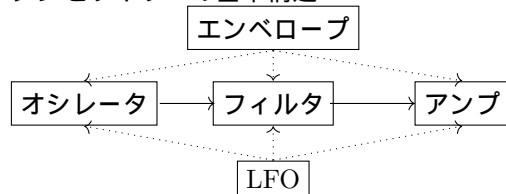
- IO 制御
  - PCM 出力:Fortran で生成した PCM データを PulseAudio を通じて途切れの少ないように出力すること。
  - シンセサイザーの操作: シンセサイザーの構成をテキストファイルで設定すること。
  - 自動演奏: 音符、休符などをテキストファイルから読み込み、オシレータ等に PCM データを生成するよう指示すること。
- 音響合成
  - 基本構成要素: オシレータ、フィルタ、エンベロープ、LFO の各音響要素をデジタル的に生成・制御すること。
  - 自動演奏実行: 読み込んだ音符や休符から音響合成の指示を自動的に行うこと。

### 1.4 システム設計

- 全体のフローチャート



## ● シンセサイザーの基本構造



## 2 音響合成部 (Fortran) の実装

本モジュールは、楽譜テキストファイルを解析し、設定されたシンセサイザパラメータに基づき、最終的に音響データを生成する純粋な計算処理を担当する。

### 2.1 オシレータ部

出力音の基礎となる簡単な形の波を発生させる。人間の発声器官のうち声帯に相当する。例として以下に矩形波を生成する Fortran の関数を示す。

```

1  IMPLICIT NONE
2  PURE FUNCTION OSC_SQUARE(X)
3      REAL, INTENT(IN)::X
4      REAL::OSC_SQUARE
5
6      REAL::I
7
8      I = X-AINT(X)
9
10     IF(I < 0.5) THEN
11         OSC_SQUARE=1
12     ELSE
13         OSC_SQUARE=-1
14     END IF
15 END FUNCTION OSC_SQUARE

```

便宜上、一周を  $2\pi$  ではなく 1 としている。

Fortran では繰り返しを SIMD に自動的に変換することがあるが、純粋関数にすることで SIMD 命令への変換が起こりやすくなるため、結果的に効率の向上に繋がる。

### 2.2 ゲイン

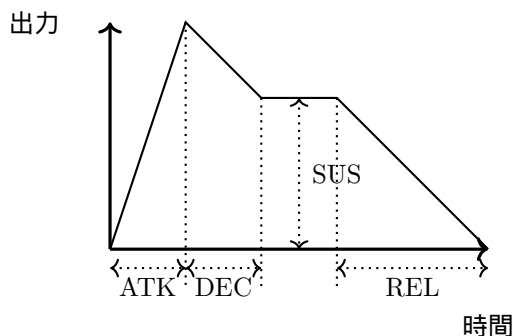
最終的な音量を変化させる。一般的に、波の式は以下の式で表される。

$$A \sin 2\pi \left( \frac{t}{T} - \frac{x}{\lambda} \right)$$

ゲインはこの式の  $A$  に相当し、いわゆる振幅、音の大きさである。

### 2.3 エンベロープ

出力の大きさを鍵盤が押されてからの時間で変化させる。以下にエンベロープの構成要素と役割をグラフで示す。



ただし、REL は鍵盤から離れた後の出力であり、いわゆる余韻。

## 3 実装計画

本プロジェクトで構築した音響合成システムは、基本機能の実装を優先したため、音色の複雑な加工や表現力の向上に繋がる以下の機能については、今後の課題として実装を計画する。

### 3.1 フィルタ

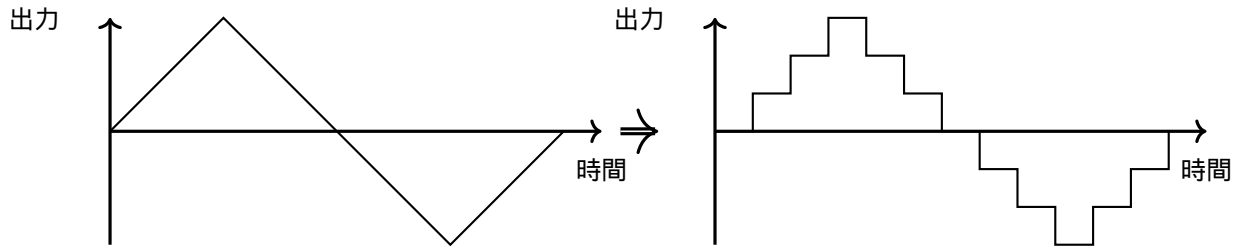
一定の周波数帯や倍音を増幅または減衰させ、音色を変化させる機能。本プロジェクトでは、アナログ回路のシミュレーションなどを利用して実装を目指す。

### 3.2 LFO

連続的に出力を変化させる機能で、ビブラートやトレモロ等の効果を演出する。

### 3.3 サンプリング周波数通減

擬似的にサンプリング周波数を下げることにより、レトロゲーム機に似た音色を演出する。  
以下に、この機能が波形をどのように変化させるかをグラフで示す。



## 4 終わりに

本研究では、Fortran と C 言語のハイブリッド構成によるトーキング・モジュレータ音響合成システムを構築した。本システムの構築における最大の目的は、Fortran の持つ計算処理能力の高さを活かして開発を行い、音響合成についての知見を深めることであった。

本研究を通じて、以下の成果と技術的知見を得ることができた。

- ISO\_C\_BINDING について

ISO\_C\_BINDING モジュールを使うことにより、Fortran と C 言語の間でデータの受け渡しが可能になった。

- 音響合成

オシレータやエンベロープなど減算式シンセサイザーの基本的な機能の実装を通して、音声合成技術の理解に近づいた。また、SIMD 最適化や、効率的なデータ構造の設計などの知見が深まった。

一方で、本システムには未だに実装できていない機能も多く、音色の多様性と言う点で課題が残されている。これらの実装に向けてこれからも努力を続けていく。