

Inside mml2wav.rb

2016/02/16

0x64 Tales

#05 Number

Livesense Inc.

HORINOUCHI Masato

デモ

- スペランカー メインテーマ ほげ
 - [spelunker.wav](#)
 - <http://lifecycle.s165.xrea.com/mml1.html> から引用
- FF4 通常戦闘曲
 - [ff4_buttle.wav](#)
 - http://www.geocities.jp/tifa_adiru/57ff4 から引用

MML

先程の音律デモの MML

1. T120 V15 04 L4 R
CDEF GAB>C< R C1& C1 F1& F1 G1& G1

2. T120 V15 04 L4 R
C1& C1& R R4E2.& E1 R4A2.& A1 R4B2.& B1

3. T120 V15 04 L4 R
R1 R1 R R2G2& G1 R2>C2& C1 R2D2& D1

track 1～3 で同時発音数 3 となる ← 誤用だけどいわゆる 3和音。

機能 (1)

- 量子化ビット数: 8bit / 16bit
- サンプリング周波数: 8kHz / 44.1kHz / 48kHz (任意変更可)
- 波形: 正弦波 / ノコギリ波 / 矩形波

機能 (2)

- 分解能(ppq): 48 / 96 / 480 (任意変更可)
- トラック数: 無限 (性能依存)
- ステップ数: 無限 (性能依存)
- 音高: A440 / A442 / A444 (任意変更可)
- 音律: 12平均律 / 純正律

できないこと

- ステレオ出力
- 三角波
- エンベロープ
- ノイズ出力
- デューティー比変更

対応コマンド

- MML の主なコマンドはほとんど対応。
 - 0 > <
 - C D E F G A B + -
 - R
 - [0-9]+ (数字) . (dot)
 - T L V Q &

処理フロー

1. MML → 簡易 MIDI シーケンスデータに変換
 - note number / steptime / gatetime / velocity
2. シーケンスデータ → $-1.0 \sim 1.0$ の波形データに変換
3. トラック毎の波形データを合成
4. 合成後の波形データを WAV にエンコード

まとめ

- MML のパースは簡単。
- 各処理を抽象化すると、機能を後付けしやすい。
 - そのため最適化がされておらず処理が冗長で重い。
- より正確にシーケンス化すると .mid への変換もできる予感。
- テーマから激しく脱線した。
 - とはいえとても楽しめたので大満足。

ご清聴ありがとうございました