

音源の暗号化

WAV ファイルにある操作を加え、オリジナルの音声を聞き取れないように加工する。

レベルを逆変数する

音声レベルを逆数にすることによって、オリジナルの音源を聞き取りにくくします。オリジナルのデータ値を v とすると、(最大値- v) へ変換します。引数は、入力の WAV ファイル名と出力用の WAV ファイル名を指定しなければなりません。以降に入力例を示します。

```
cript <入力ファイル名> <出力ファイル名>
```

以降に、実行例を示します。ステレオの 1002 [Hz] の Sin 波を変換します。

```
C:\>cript sin1002Hz8_10msec.wav cript01.wav
```

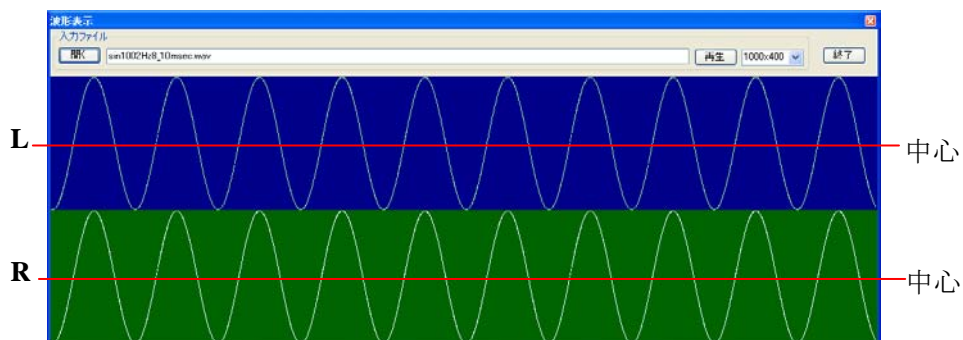
```
ファイル名[sin1002Hz8_10msec.wav]
"fmt" の長さ: 16 [bytes]
    データ形式: 1 (1 = PCM)
    チャンネル数: 2
    サンプリング周波数: 44100 [Hz]
    バイト数 / 秒: 88200 [bytes/sec]
バイト数×チャンネル数: 2 [bytes]
    ビット数 / サンプル: 8 [bits/sample]

"data" の長さ: 880 [bytes]

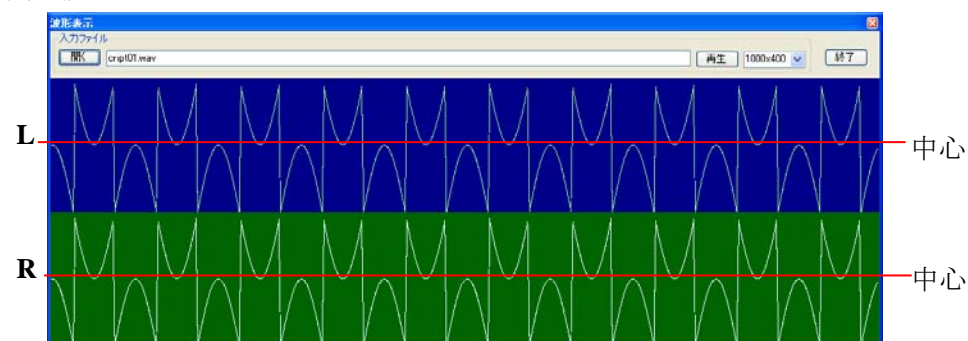
時間=0.010
```

[sin1002Hz8_10msec.wav] を [cript01.wav] へ変換しました。

入力波形:



出力波形:



中心(無音)を境にして、レベルが逆転しています。変換後のWAVファイルにはグリッチが多数含まれますので、スピーカの音を小さくして再生してください。変換後の音は非常に耳に良くない音となります。

さて、このWAVファイルを、再度criptプログラムで変換してみましょう。可逆性がありますので、元の音に戻るはずですが、以降に動作の様子を示します。

```

C:\>cript cript01.wav cript02.wav

ファイル名[cript01.wav]
"fmt "の長さ: 16 [bytes]
    データ形式: 1 (1 = PCM)
    チャンネル数: 2
    サンプリング周波数: 44100 [Hz]
    バイト数 / 秒: 88200 [bytes/sec]
バイト数×チャンネル数: 2 [bytes]
    ビット数 / サンプル: 8 [bits/sample]

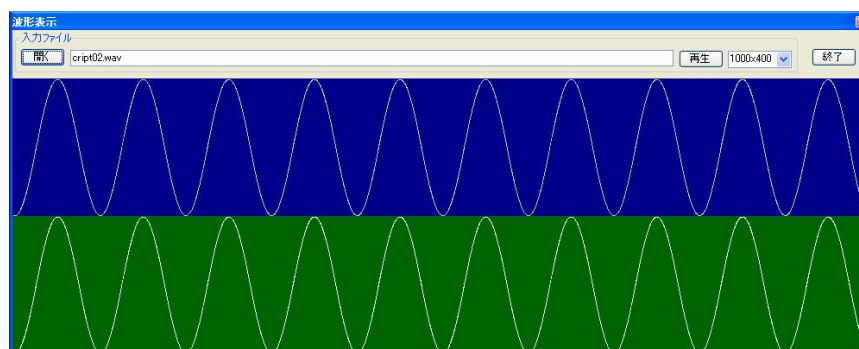
"data" の長さ: 880 [bytes]

時間=0.010

[cript01.wav] を [cript02.wav] へ変換しました。

```

出力波形:



二回変換を行うと完全に元に戻ります。cript02.wav は sin1002Hz8_10msec.wav と同じ内容になります。

続いて、ステレオの、ある曲を変換します。以降に動作の様子を示します。

```

C:\>cript 異邦人_10msec.wav cript03.wav
wav

ファイル名[異邦人_10msec.wav]
"fmt "の長さ: 16 [bytes]
    データ形式: 1 (1 = PCM)
    チャンネル数: 2
    サンプリング周波数: 44100 [Hz]
    バイト数 / 秒: 176400 [bytes/sec]
バイト数×チャンネル数: 4 [bytes]
    ビット数 / サンプル: 16 [bits/sample]

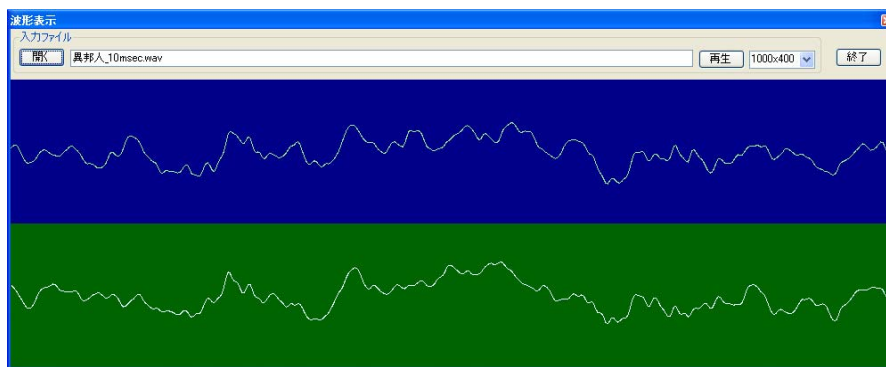
"data" の長さ: 1760 [bytes]

時間=0.010

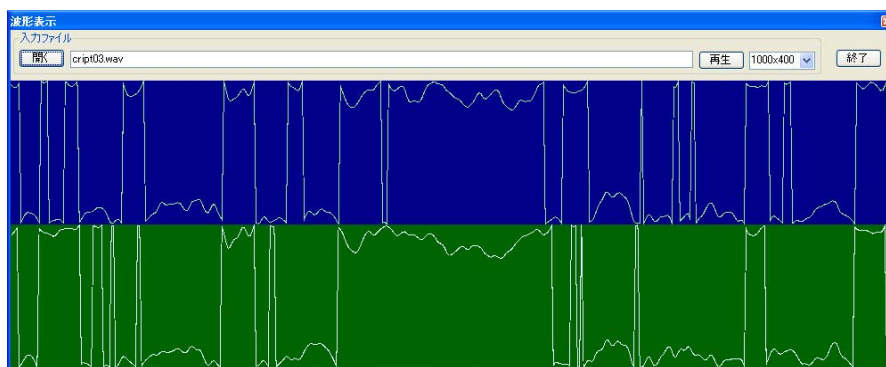
[異邦人_10msec.wav] を [cript03.wav] へ変換しました。

```

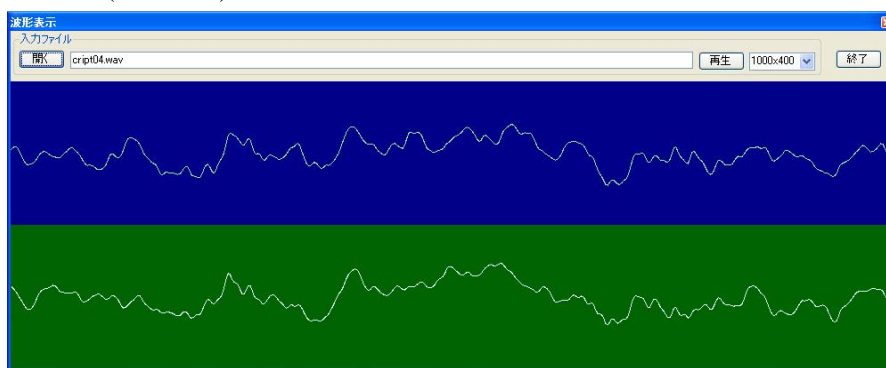
入力波形:



出力波形:



再変換後(元に戻る):



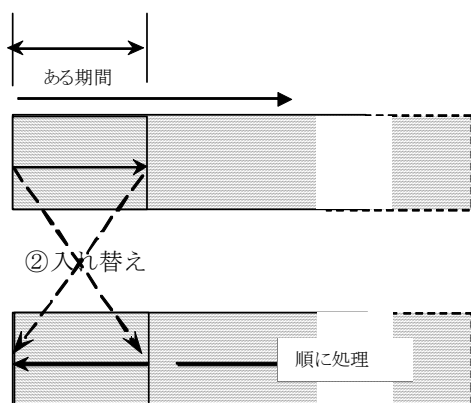
ある期間で逆再生

もうひとつのプログラムは、ある期間で周期的に逆再生する WAV ファイルを生成します。周期的に逆再生されるため、原音が何であるか分かりにくくなります。逆再生の周期が分かれば、元の WAV ファイルに戻すことが可能です。

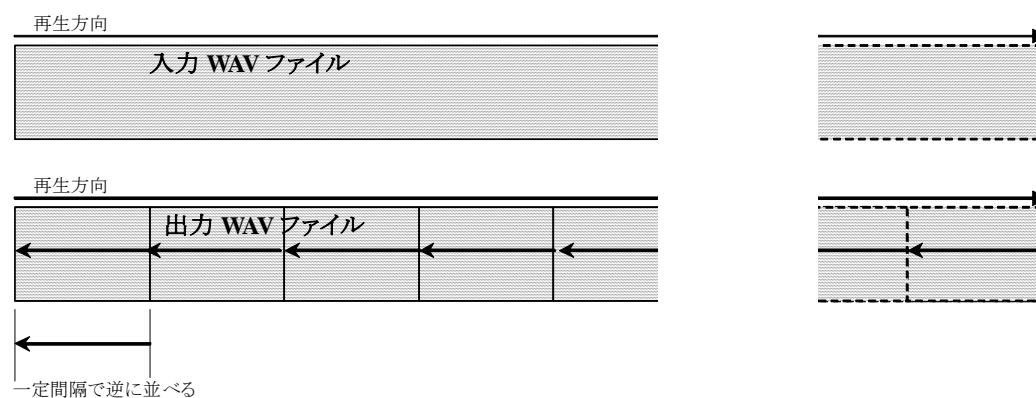
引数が二つ指定されているか調べます。本プログラムには、入力の WAV ファイル名と出力用の WAV ファイル名を指定しなければなりません。以降に入力例を示します。

```
cript2 <入力ファイル名> <出力ファイル名>
```

処理の概念図を示します。



処理が終わったときの、入力ファイルと、出力ファイルの関係を以降に図で示します。



以降に、プログラム実行例を示します。モノラルの曲を変換します。

```
C:\>cript2 月16mono.wav cript2_01.wav

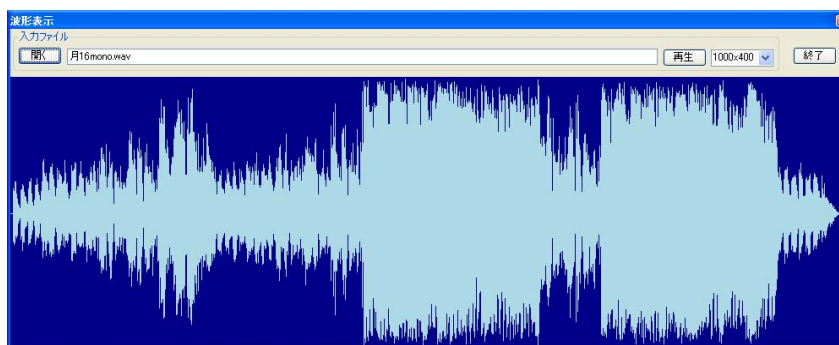
ファイル名[月16mono.wav]
"fmt "の長さ: 16 [bytes]
  データ形式: 1 (1 = PCM)
  チャンネル数: 1
  サンプリング周波数: 44100 [Hz]
  バイト数 / 秒: 88200 [bytes/sec]
  バイト数×チャンネル数: 2 [bytes]
  ビット数 / サンプル: 16 [bits/sample]
```

"data" の長さ: 27309072 [bytes]

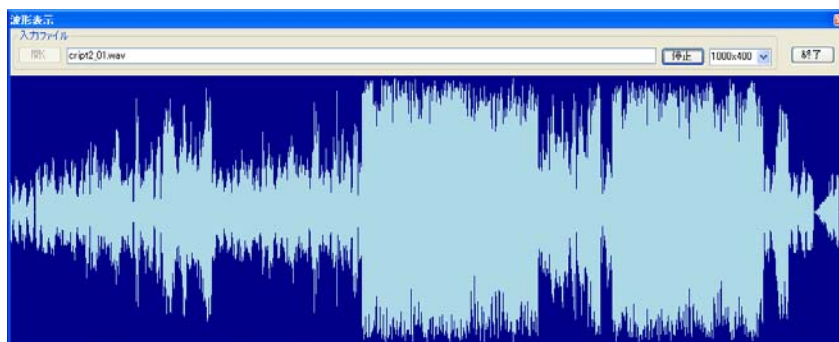
時間=309.627

[月 16mono.wav] を [cript2_01.wav] へ変換しました。

入力波形:



出力波形:



入れ替える期間に比べ全体が長いので、詳細に見ないと視覚上は分かり難いです。ただ、細かく見ていくと微妙に形が違うのが分かります。再生すると、原曲とはまるで異なっています。さて、先ほど変換した WAV ファイルを、再度 cript2 プログラムで変換してみましょう。可逆性がありますので、元の音に戻ります。

```
C:\>cript2 cript2_01.wav cript2_02.wav
```

ファイル名[cript2_01.wav]

"fmt" の長さ: 16 [bytes]

データ形式: 1 (1 = PCM)

チャンネル数: 1

サンプリング周波数: 44100 [Hz]

バイト数 / 秒: 88200 [bytes/sec]

バイト数 × チャンネル数: 2 [bytes]

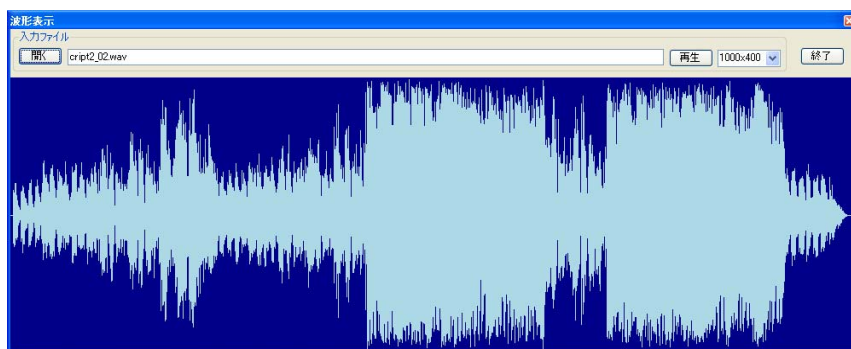
ビット数 / サンプル: 16 [bits/sample]

"data" の長さ: 27309072 [bytes]

時間=309.627

[cript2_01.wav] を [cript2_02.wav] へ変換しました。

出力波形:



二回変換を行うと完全に元に戻ります.