

# lilcaSoundDriverFC 仕様書

lilca reload(Masayuki Ise)

2025-12-08

## Contents

負荷	1
メタ情報	2
変換されたコードのコメントとして出力されます	2
いずれのメタ情報もキーワード直後の区切り文字以降の文字列を前後トリミングして値とする	2
マクロ	2
マクロパラメータ値	3
マクロ値	3
ボリューム, 音色, アルペジオ, ピッチ共通	3
ビブラート	4
DPCM	4
チャンネル	5
内部フォーマット (構文解析のみ)	5
音階	6
長さ (音符, 休符)	7
デフォルト値	7
タイマー値の計算	7
参考文献	7

## 負荷

VBlink : 20 x 341 = 6820[cyc]  
OAM転送 : 512[cyc]

## マクロ

---

サウンドドライバ: 820 + 55 x 2 [cyc]  
曲のデータサイズ: 159[byte] (The Theme of lilca)

## メタ情報

変換されたコードのコメントとして出力されます

- 書式

```
#<META_NAME> <string>
```

- FC 側で使う可能性のあるメタ情報

項目	キーワード	備考
曲名	#TITLE	
作曲者	#COMPOSER	
プログラマ	#PROGRAMER	MML やサウンドデータに起きた人
ラベル	#LABEL	プログラム上で使われるラベル名 (*1)

(\*1 mck mml の仕様にない)

いずれのメタ情報もキーワード直後の区切り文字以降の文字列を前後トリミングして値とする

例:

```
"#COMPOSER    lilca reload    "
```

```
Meta Value = "lilca reload"
```

## マクロ

- 一覧

マクロ名	定義	使用	ループ指定	終端子	有効チャンネル	備考
ボリューム	@v [n]	@v [n]	あり (*2)	-	12N (*1)	フレーム単位でボリュームを変化させる
音色	@[n]	@@[n]	あり (*2)	-	12N (*1)	フレーム単位で音色を変化させる

マクロ名	定義	使用	ループ指定	終端子	有効チャンネル	備考
アルペジオ (*3)	@EN [n]	EN[n]	あり (*2)	ENOF	12TN (*1)	フレーム単位で音階を変化させる
ピッチ (*3)	@EP [n]	EP0	あり (*2)	EPOF	12TN (*1)	フレーム単位で指定方向と量で音階を変化させる
ビブレート (*3)	@MP [n]	MP[n]	-	MPOF	12TN (*1)	指定された周期と振幅のSin波で音階を変化させる
DPCM 割り当て (*3)	@DPCM [n]	音階	-	-	D (*1)	n=0,1,2,3,4 …-> c,c+,d,d+,e …に割り当てる

\*1 チャンネル名: 1= 矩形波 1, 2= 矩形波 2, T= 三角波, N= ノイズ波, D=DPCM

\*2 “|”以降をループする

\*3 未対応

## マクロパラメータ値

### マクロ値

ビット	範囲	説明
1000-0000	-128	ループバックシンボル (次の 1 バイトが戻る値)
1000-00010111-1110	-127 ~ 126	マクロ値
0111-1111	127	未使用

ボリューム, 音色, アルペジオ, ピッチ共通

## マクロパラメータ値

値	範囲	説明
ボリューム値	0~15	
音色	0~3	矩形波 =duty 比ノイズ波 = ノイズモード
相対音階	-127~126	c に対して 1 は c+1 は b
増減値	-127~126	矩形波と三角波 -> 11bit タイマーノイズ波 -> 4bit ピリオド

@v0={ 15 8 4 2 0} ; 1フレーム目 15 ... 5フレーム目 0 それ以降 0

@v1={ 15 |8 4 } ; 1フレーム目 15, 2フレーム目 8, 3フレーム目 4, それ以降 8と4を繰り返す

@v2={|15 15 12 11 10 8 8 8 } ; '|' 以降の数値を繰り返す

@ep0={2 |4 0} ; 1フレーム目 +2, 2フレーム目 +6, 3フレーム目 +6, 以降 4,0増加させる (\*)

@ep1={-16}; 1フレーム目 変化なし, 2フレーム目 16減少, 3フレーム目 さらに16減少, 以降 16現

(\* 最大値を超えたなら最大値に、最小値を下回ったら最小値にする)

## ビブラート

値	範囲	説明
遅延	0~255	Sin 波を適用するまでのフレーム数
速さ	1~255	Sin 波の周期
深さ	0~255	Sin 波の振幅矩形波と三角波 -> 11bit タイマーノイズ波 -> 4bit ピリオド

@mp={4 8 6} ; 4フレーム変化ないし, それ以降 周期8フレーム, 振幅6のSin波を適用する

## DPCM

値	範囲	説明
ファイルパス	文字列	実行ディレクトリからの相対パス
パラメータ 1	0~15	ピッチ 15= ノーマル (*)4010b3 - 0RateIndex  2 0 4081  (16byte ) < br > 0 = (*)  3 0 15  ( )(*) < br >FF 推奨この値から波形が増減する
パラメータ 4	0~2	再生モード (*)0=1 ショット, 1= ループ

(\* 後方省略可能だが、中間のパラメータは省略できない)

- (著者用のメモ)

```

<Macro> ::= "@" <MacroName> <M_Number> "=" "{" <NumberList> "}"

<NumberList> ::= <Number>+          -- ループなし
|   <Number>+ "|" <Number>+    -- ループあり
|   "|" <Number>+           -- 冒頭からループ
|   <Delay> <Speed> <Depth>  -- ビブラートのとき

<MacroName> ::= "v" | "@" | "en" | "ep" | "mp"

<M_Number> ::= 0-127
|   0-63           -- ビブラートのとき

<Number> ::= 0 ~ 15      -- ボリューム
|   0 ~ 3          -- 音色
|   -127 ~ 126     -- アルペジオ
|   -127 ~ 126     -- ピッチ

<Delay> ::= 0 ~ 255
<Speed> ::= 1 ~ 255
<Depth> ::= 0 ~ 255

```

## チャンネル

チャンネル名	シンボル
矩形波チャンネル 1	A
矩形波チャンネル 2	B
三角波チャンネル	C
ノイズチャンネル	D
DPCM チャンネル	E

## 内部フォーマット（構文解析のみ）

コマンド値	意味	サイズ	備考
\$0_nn_ff	音符	nn = ノート番号 ff = フレーム数 (*1)	
\$1m_ff	休符	m = モード (0= 音止める, other= 音止めない) ff = フレーム数 (*1)	
\$2v	音量	v = 音量値 (0-15)	

## 音階

---

コマンド値	意味	サイズ	備考
\$3t	音色	t = 音色: 矩形波 (0-3) ノイズ (0-1)	
\$4_ pp pp	ピッチ	pppp = ピッチ量 (-32768~32767)(*3)	
\$5o	オクターブ	o = オクターブ値 (0-5 => o2-o7)	
\$6_	オクターブ UP	引数なし	
\$7_	オクターブ DOWN	引数なし	
\$80(*4)	ボリュームマクロ	マクロ番号	(*2)
\$81	音色マクロ	マクロ番号	(*2)
\$82	アルペジオマクロ開始	マクロ番号	(*2)
\$83	ピッチマクロ	マクロ番号	(*2)
\$84	ビブラートマクロ開始	マクロ番号	(*2)
\$8C	アルペジオマクロ終了	なし	
\$8D	ピッチマクロ終了	なし	
\$8E	ビブラートマクロ終了	なし	
A_ (FF を設定)			
\$Br	リピート終了	r = リピート回数 (0-15)	
\$FF	終端子	引数なし	

---

## 廃止

\$80 (無くなるかも)	ボリュームマクロ	マクロ番号	(\*2)
\$81	音色マクロ	マクロ番号	(\*2)
\$82	アルペジオマクロ開始	マクロ番号	(\*2)
\$83	ピッチマクロ	マクロ番号	(\*2)
\$84	ビブラートマクロ開始	マクロ番号	(\*2)
\$92	アルペジオマクロ終了	なし	
\$93	ピッチマクロ終了	なし	
\$94	ビブラートマクロ終了	なし	

\*1 省略時は“-1”が入る

\*2 バイナリ変換時に番号は振り直すので mml の番号とは一致しない

\*3 ピッチ量は通常 0

\*4 D0 DF は

## 音階

- 音階 n: C1=0 - A8=93

a4の周波数 = 440

周波数f: = 440 x 2 xx (n / 12)

- 算出式

```
f = fn(note, octave)
= 440 * 2 ** ((note-9) + (octave-4) * 12 - )/12)
タイマー値T: = -1 + 1789773 / 16 / f
```

## 長さ（音符、休符）

- 定義

テンポ t (例 = 160)  
符長 len (例 = 32 = 32分音符)

\* 符長：音符や休符の長さ

- 定数 A

A = 60fps x 60秒 x 4分音符 = 14400

- 符長のフレーム数 F

F = Int(A / t / len)

\* Int()で切り捨てられた少数点以下の値は累積され、  
"1"を超えると音符や休符を1フレーム長くして調整している

## デフォルト値

項目	デフォルト値
テンポ	120
長さ	4
ボリューム	12
オクターブ	4

### タイマー値の計算

f = clock / (16 \* (timer + 1)) timer = clock / (16 x f) - 1

# ##d 未整理メモ f ノイズチャンネル @v1={12 15 14 14 13 13 | 8 8 8 8 3 3 3 3 0} ; 音階 c - b  
0-4 5 6 7 8 9 a b c d e f ; \$400E b3-0 = 0 1 2 3 4 5 6 7 8 9 a b 音階 c d e f g a b 周期 5 6 7 8 9  
a b 0-4, c-f が使えない

## 参考文献

Ultimate PPMCK MML Reference

MCK/MML BEGINNERS GUIDE