

Standard MIDI File Compiler mdcomp32 v1.01

for Windows 7 / 10 (GUI)

説明書

1.01.00-00



wnc develop

この説明書は mdcomp32 に標準添付しているヘルプファイルを元に作成しています.

はじめに	2
■ GUI 版と DOS 版について	2
■ 重要事項	2
動作環境	3
インストール・アンインストール	4
■ インストール方法	4
■ インストール方法(インストーラ付)	4
■ アンインストール方法	5
■ アンインストール方法 (インストーラ付)	5
mdcomp について	6
■ mdcomp の必要性	6
使い方	7
■ 初期画面 (画面の採取は Windows 7、一部 2000Pro で採取)	7
■ ボタンの説明	7
エディタの解説	8
使ってみよう	11
■ 使用例	11
命令解説	12
■ ソースで使用できる命令	12
■ データとして使用できる命令 (大文字小文字区別なし)	12
□ ドレミコード	13
データ認識について	14
■ 連符	14
■ op 命令	15
■ 書式	16
演奏用ソースのサンプル	17
技術的な情報	22
■ 自動左右配分での各チャンネル初期値	22
■ スタンダード MIDI ファイル (SMF)	23
■ ヘッダ (MThd)	24
■ トラック (MTrk)	24
■ トラック	26
■ ティックカウント	27
■ mdcomp エラー内容一覧	28
仕様	32
約束	33
■ mdcomp32 固有の約束	33
おしまいに	34
■ 参考書類	34
開発履歴・その他	35
開発元	36

■はじめに

wnc develop 製作のソフトウェアをご利用いただきましてありがとうございます。このソフトウェアはスタンダード MIDI ファイルを作成するために使用するソフトウェアで、昔の BASIC で使用していた記号 (MML) で作曲することができます。どちらかというと N88BASIC や MSXBASIC で作成されている PLAY 文の中身を mdcomp32 を使用して SMF に変換する作業に適しています。

mdcomp32 は以下のファイルで構成しています。

mdcomp32. EXE	プログラム本体です
mdcomp32. CFG	環境設定保持用です
mdcomp32. CHM	Windows HtmlHelp版説明書です
mdcomp32. PDF	pdf版説明書(pdf表示ソフトウェアが必要)

■GUI版とDOS版について

GUI 版は統合開発環境になっていますので、編集や演奏を mdcomp32 上から行うことができます(演奏は MCI デバイスを使用しますので特別なソフトウェアは必要ありませんが、MCI デバイス経由で MIDI が再生できることが条件です)。DOS 版はコンパイラ機能だけです。別途 text エディタや SMF を演奏するソフトウェアが必要です。

注) 出力ファイルは拡張子が .mid です

■重要事項

このソフトウェアを使用した結果、パソコンや USB メモリ・ハードディスク類に保存されているデータなどに影響を与え、損害を出しても作成者およびこのソフトウェアの配布者(掲載者)は一切責任を負いません、ということをご理解ください。

普通の使用法ならばこのような事態は起こりませんが、できるだけ重要データはバックアップをとるか、業務に全く影響を与えないパソコンでの使用をおすすめします。

■動作環境

Microsoft Windows 7 または Windows 10 が必要で、CPU は 80x86 完全互換で動作します。

Windows 10 は 64bit 版でのみ作動確認を行っています(WOW64 上で 32bit 起動)。
作動機械は、上記のOSが動作保証されている PC/AT 互換機です。

※XP, Vista, 8, 8.1は従来のVerでは作動確認できていましたが、諸事情により初期検査未最新の作動確認状況は Web 頁をご覧ください。

以下の機種で作動確認を行いました

NEC

PC-GL21DJ5 (Pentium P6200@2.13GHz,6GBmem,Windows 10 Pro 64b Ver1703)

PC-GL21DJ5 (Virtual Box 5.1.28 , Windows 2000 Pro SP4) *参考用

SONY

PCG-SRX7E/P (PentiumIII 800MHz,256M,Windows XP Pro SP3)

Windows は Microsoft 社の登録商標または商標です。
Pentium,Celeron は Intel 社の登録商標または商標です。
製品名および会社名は各社の登録商標または商標です。

■インストール方法

ハードディスクか USB メモリに

mdcomp32.EXE

mdcomp32.CHM

2つのファイルを同じフォルダに保存すれば完了です。実行するにはエクスプローラから **mdcomp32.EXE** をダブルクリックすると起動できます。必要であれば、ショートカットを作成したりスタートメニューに登録してください。

mdcomp32.EXE に必要な DLL などは OS 標準状態で供給されているものだけを使用しているので特別な DLL やコントロールモジュールは必要ありません。
(コンパイラの基本コンポーネントのみで製作しています)

■インストール方法(インストーラ付)

現バージョンではインストーラ付きを提供していません。

注) 過去のバージョンでは付けてました

※**mdcomp32**ではレジストリを操作する部分は一切ありません。

■アンインストール方法

レジストリなどを操作していないのでアンインストールは下記のファイルを削除するだけで完了です。

(スタートメニューやショートカットを作成した場合はそれも削除してください)

mdcomp32.EXE

mdcomp32.CFG

mdcomp32.CHM

mdcomp32.EXE は必要な機能を内部ですべてもっていますので勝手に DLL を作成したり SYSTEM フォルダに移したりしません。また、レジストリやその他の設定ファイルは一切触れていません。

■アンインストール方法(インストーラ付)

現在のバージョンではインストーラとアンインストーラはありません。

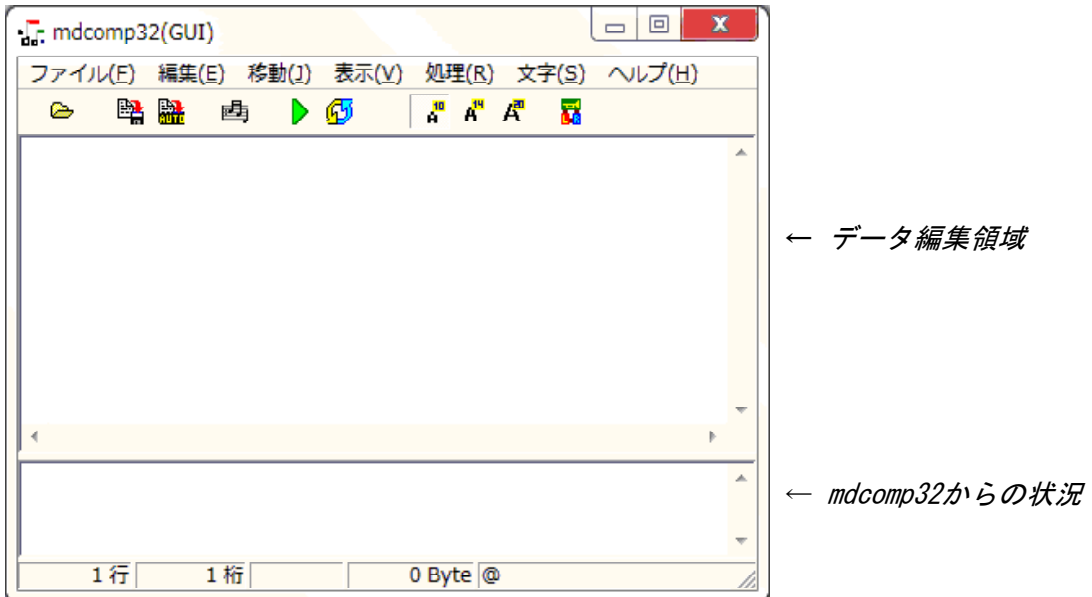
■mdcompの必要性

mdcomp32 は開発初期段階では DOS プロンプト版の FMDRV 用コンパイラ（コンパイラというよりアセンブラ）として稼働していました。2000 年に FMDRV という PC-9801 系専用の FM 音源ドライバを完成し、それにあわせて FMCOMP という FMDRV 用のデータを作成する専用コンパイラの開発に着手することになります。近年 GUI の OS 上で開発をすることが多くなり、旧式の機械を使用することが少なくなったため GUI の OS でも演奏および変換ができるようにと演奏データ出力部分を SMF 対応に交換したものを mdcomp としました。これは業務専用開発した必要最小限の機能しかもたない DOS プロンプト版コンパイラでした。後に GUI 版の開発を始めることになりますが、それにあわせてある程度の機能を持たせた SMF コンパイラとして強化することにしました。現在の mdcomp32 でテキストエディタのような画面構成であるのは、DOS 上での開発を意識したのもです。市販ソフトウェアや著名なオンラインソフトウェアに比べると簡素で機能も少なくとても対抗できるものではありませんが、他と比較してもかなり簡単に作曲することは可能になっています。

注) 2002 年当時の文章です

■初期画面(画面の採取はWindows 7、一部2000Proで採取)

@起動直後の画面



mdcomp32 は上記の画面でデータを作成し演奏するようになります。上側の文字が入力できるところはデータを入力し、下側の同じような枠の上下が狭まっているところにはメッセージを表示します。

■ボタンの説明

@ボタンの絵

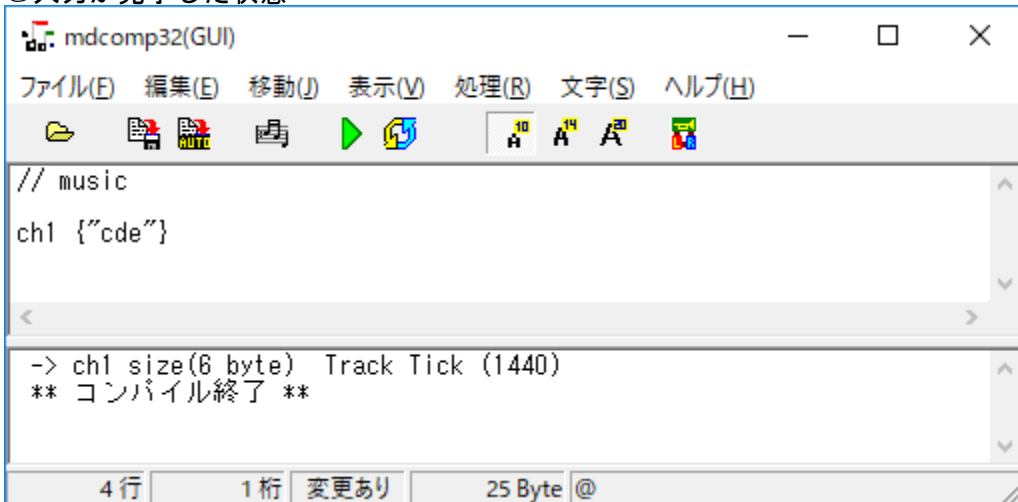


左から、開く・上書き保存・自動上書き保存・midi ファイル作成・コンパイル後再生・演奏ループ・文字小・文字中・文字大・自動左右配分

※文字小・文字中・文字大は3つのうち一つだけ選択できます（選択以外は解除）

■エディタの解説

@入力が完了した状態

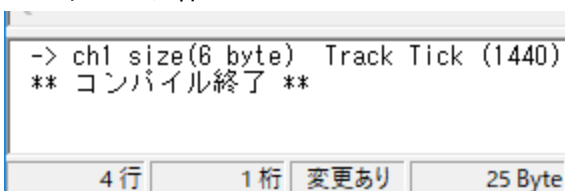


@入力枠



ここに演奏データを書き込みます。

@メッセージ枠



mdcomp32 が処理に応じて書き加えます。

何か問題が発生した場合はメッセージ枠に状況を表示します。また Track Tick (1440) となっている箇所がありますが、括弧内の 1440 は ch1 の長さを示しています。

@ファイル

ファイル(F)	編集(E)	移動(J)	表示(L)	開く	: ソースを開きます (テキスト形式)
				開く(O)...	
				閉じる(C)	閉じる : ソースを閉じます (新規作成状態にします)
				midi ファイル作成(M)	midi ファイル作成 : Standard MIDI File を作成します
				fmcomp データ出力(F)	fmcomp データ出力 : fmcomp 準拠データを出力し保存します
				上書き保存(S)	上書き保存 : 開いているソースを上書きします
				自動上書き保存(V)	自動上書き保存 : コンパイル後再生を行う時にソースを上書きします
				名前をつけて保存(A)	名前を付けて保存 : ソースに名前を付けて保存します
				終了(X)	終了 : mdcomp32 を終了します

※midi ファイル作成で作成した拡張子がmidのファイルはwindowsに標準でついているメディアプレイヤーで再生することが可能です。また、他のプレイヤーやシーケンサーでもスタンダードMIDIファイル(FORMAT 1)として使用できます。

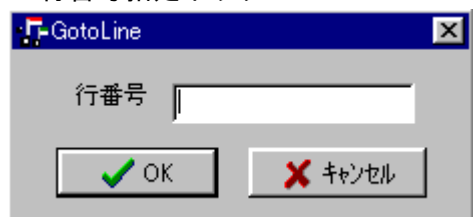
@編集

編集(E)	移動(J)	表示(L)	
元に戻す(U)	Ctrl+Z	元に戻す	: 現在行った作業を直前の状態に戻します
切り取り(T)	Ctrl+X	切り取り	: 範囲指定した文字列をクリップボードに切り取ります
コピー(C)	Ctrl+C	コピー	: 範囲指定した文字列をクリップボードにコピーします
貼り付け(P)	Ctrl+V	貼り付け	: クリップボードからカーソルの位置へ貼り付けます
削除(D)	Del	削除	: 範囲指定した文字列を削除します
すべて選択(L)		すべて選択	: 編集しているソースをすべて選択します

@移動

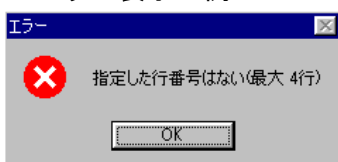
移動(J)	表示(L)	先頭	: カーソルをソースの先頭へ移動します
先頭(S)		最後	: カーソルをソースの最後へ移動します
行番号指定(G)		行番号指定	: 指定した行番号にカーソルを移動します
最後(E)			

@行番号指定ボックス



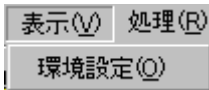
※行番号指定を選択すると右図のボックスが出現しますのでそこに行番号を入力してください。
行番号の指定に問題がある場合はエラーを表示します。

@エラー表示の例



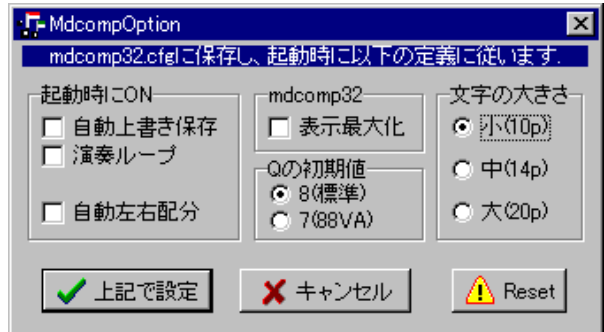
← 最大行数を越える数値を入力した場合

@表示



環境設定：起動時に反映させる項目を設定します。

@環境設定画面



起動時に反映させたい項目をクリックしてチェック ☒ している状態にしてください。

文字の大きさは小・中・大のうち一つしか選択できません。

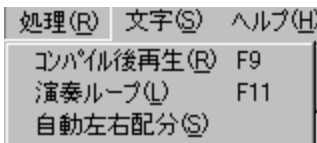
表示最大化は起動時にフォーム最大化ボタンを押したのと同じ効果があります。

設定が完了したら 上記で設定 ボタン

をクリックしてください。設定を mdcomp32.cfg に保存します。 キャンセル ボタンを押すと保存せずに設定画面を閉じます。 Reset ボタンを押すと初期設定になります。

※mdcomp32. cfg ファイルを削除しても同じように初期設定になります。

@処理



コンパイル後再生：コンパイル後演奏を開始します

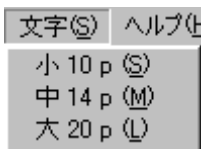
演奏ループ：演奏を繰り返します

演奏ループが ON の時は → に変化します。

自動左右配分：チャンネルごとに自動で音を振り分けます

自動左右配分が ON の時は → に変化します。

@文字

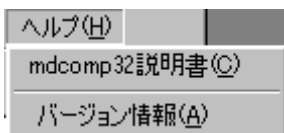


小：エディタの文字を小さく表示します（10 ポイント程度）

中：エディタの文字を通常表示します（14 ポイント程度）

大：エディタの文字を大きく表示します（20 ポイント程度）

@ヘルプ



mdcomp32 説明書：ヘルプを表示します

バージョン情報：著作権を表示します

■使用例

ドレミを演奏すると仮定してデータを作成してみます。

mdcomp32 では

```
ch1 { "cde" }
```

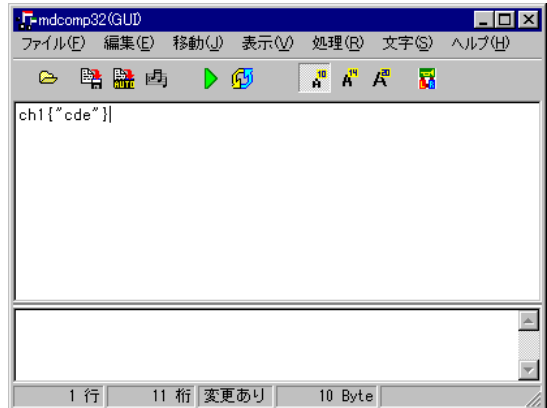
と、入力します。

入力が完了したら  をクリックしてください。
(または、メニューの 処理→コンパイル後再生)

ドレミの音はでましたか？もし何らかのエラーが表示した場合は入力したデータをもう一度よく見直してみてください。もし MCI デバイスのエラーが発生した場合は他で音源を使用中かまたは使用している本体に音源を搭載していないことが考えられます。

(ソフトシンセ (例：VSC-88 など) では演奏できます)

@画面



@mdcomp32からの結果表示


```
-> ch1 size(6 byte) Track Tick (1440)
** コンパイル終了 **
```

@演奏中の画面 (ボックスが出現)



トラックバーをマウスで移動させることにより演奏を進めたり戻したりできます。また、演奏中に編集する事も可能です。

注)トラックバーをマウスで移動させると各チャンネル楽器はピアノに戻ります

※演奏中にソースを変更しても影響はありません。変更を反映させるには  をクリックして再コンパイルするか、メニューからコンパイル後再生を選んでください。

■ソースで利用できる命令

命令	説 明	値
ch?	指定したチャンネルをオープンするための前手続き(?には数値) ※ ch9 はリズムチャンネル	1 ～ 9
{	ch 命令で指定したチャンネルをオープン	
}	{ 命令でオープンしたチャンネルをクローズ	
//	出現以降改行までコメントとする	
/*	コメントの開始 (/* が出現するまではコメントと見なします)	
*/	コメントの終了	
"	データ開始または終了	
op	各種設定をソース上から変更	
cr("文字列")	著作権表示文字列を組み込みます (最大 63Byte まで)	
sn("文字列")	シーケンス名文字列を組み込みます (最大 127 Byte まで)	

■データとして使用できる命令(大文字小文字区別なし)

命令	説 明	値の範囲	初期	例
o?	オクターブの変更(?には数値)	0 ～ 7	4	o5
L?	省略時に使用する音符長(.は 1.5 倍)	1,2,4,8,16,32,64	4	L32 L16.
R?	休符(.は 1.5 倍)	1,2,4,8,16,32,64		R8 R4.
@?	楽器番号の変更	0 ～ 127	0	@5
T?	テンポ設定(1 分間に四分音符を鳴らす数)	10 ～ 250	120	T160
#	半音上げ (単独で使用できません)			C#4 D#
+	半音上げ (#と同)			C+4 C+
-	半音下げ (単独で使用できません)			D-8 D-
&	音符を連結 (単独では使用できません)			c4&c8 c2&c2&c8
<	オクターブを 1 つ下げる			c<c<c
>	オクターブを 1 つ上げる			c>c>c
V?	音量を指定する	0 ～ 15	10	V15
*:?	音階の長さを直接指定 (単独では使用できません) [* に音階 ? に長さ]	1 ～ 255 ※技術的な情報 を参照		c:5 d#:255 g:-100
Q?	発音時間を指定する(?に割合 : ?/8 で計算)	1 ～ 8 (8=100%)	8	Q6
{ }	連符指定 (最大 32 連符まで)			{cde} 16
\$?	panpot(左右音の配分) を指定する	0 ～ 127 (64=中央)	64	\$0 \$70

ロドレミコード

命令	説明	値	例
c?	ド	?には音符、c4 は音階ドで四分音符という意味 音符省略時は L?で指定されている値を使用	c c4 c#4 c#4.
d?	レ		d d4 d#4 d#4.
e?	ミ		e e4 e#4 e#4.
f?	ファ		f f4 f#4 f#4.
g?	ソ		g g4 g#4 g#4.
a?	ラ		a a4 a#4 a#4.
b?	シ		b b4 b#4 b#4.

※ c-やb#は使用できますがこの場合同じオクターブのbやcになります.
(BASICの説明にあるように仕様をあわせました)

■データ認識について

ch1 { "o3CDEF" } //オクターブ 3 のドレミファ

の時、CDEF はオクターブ 3 になる（ o?命令が出現しない限り 3 のまま）

ch1 { "L4CDE8FG" }

L4 は長さ四分音符で CD、E には 8 がついているので E のみ八分音符。FG は四分音符となる。音階の後に付いている数値はそこだけ音符を指定数値に変えるという意味がある。

■連符



このような連符が存在したとき以下のように定義します。

{fga} 4

{ } は連符を指定する意味になり、fga はそれぞれファソラです。4 は 4 分音符という意味になり、4 分音符の長さの中に fga の音符が存在するとコンパイラに指示します。



左図のようにオクターブが一段上がる場合は以下のように指定します。

{ab>c} 4

または、オクターブを固定する場合は（この場合 O4 から O5）

o4 {abo5c} 4

と、する事も可能です。ただし、連符の間に指定できる命令はオクターブ変更命令のみです。

省略入力に対応しているので以下の定義も有効になります。

L16 cde {fga} cde

音符解析で c16 d16 e16 {fga} 16 c16 d16 e16 というように認識します。

■op命令

GUI 版ではマウスで各ボタンをクリックすれば設定を変更することができますが、一部の命令についてソース上から変更することができます。この命令は DOS 版でコンパイルした場合にも同様の結果が得られるように追加した命令です。

op (スイッチ 1,スイッチ 2)

スイッチはいくつでも , (カンマ)で区切ることにより指定できます。指定できるスイッチ名は以下のようになります。

スイッチ名	意味
Q7DEF	Q7 を初期値とします(88VA 互換)
Q8DEF	Q8 を初期値とします
AUTOLR	自動左右配分を作動させます
MANULR	手動左右配分とします

op 命令はソースの先頭に入れておく方がバージョンアップしても互換を保つことができます。ソースの途中にあった場合 op 命令以降のコンパイルの音に変化がでます。これは Q7DEF や Q8DEF が出現したときにその時点で Q の値を再初期化するので、例えば Q5 に設定したあと op(Q7DEF) を処理したら次の命令から全チャンネル Q7 になります。また、スイッチ名は大文字小文字関係なく指定できます。

入力例 1

op (autoLR)

入力例 2

op(autoLR , Q7def)

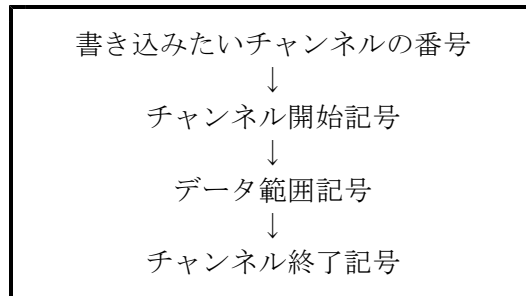
以下の入力では最後に渡したスイッチが有効になります

op(Q7def , Q8def)

Q7def と Q8def は同じスイッチを変更するので最後に渡した Q8def が有効になる。

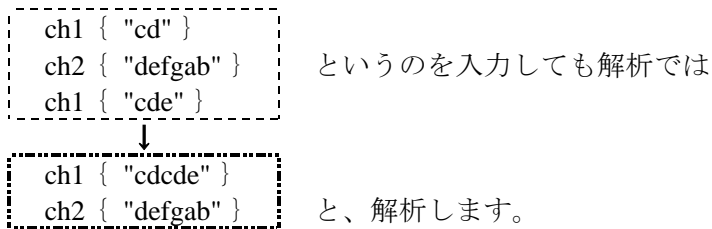
■書式

基本的には

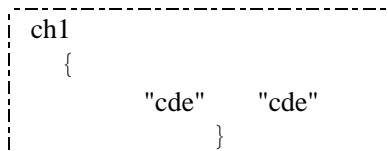


でひとつのチャンネルが完了しますが、同じチャンネル名は違うチャンネルデータの作成後に何度もあらわれても続きに書き込みます。

例えば、



1 行に必ずしも収まる必要はなく、以下のような形式でも作成できます。



使用する方が見やすい形で記入することをおすすめします。

(C 言語での書き方を参考にしています)

■ サンプル(ヘルプファイルに同等のサンプルがありますのでそれを利用してください)

//コメント(//が出現したら改行までコメントと見なす)

//The Last space

//(C) 1995,2017 wnc develop

//注意：転用不可

//1995. 3.24 (fry)

//1995.12. 9 (sat) rev

//2017. 9.23 (wed) 再調整/追加

cr ("1995,2017 wnc / t.w.")

sn ("Last Space")

ch1 { "@80 v15" }

ch2 { "@80 v15 \$40" }

ch3 { "@80 v15 \$90" }

ch4 { "@16 v12 \$120" } //オルガン

ch5 { "@25 v13 \$20" }

ch6 { "@25 v13 \$80" }

ch9 { "v13" }

ch1 { "o4l4c#d# c d " }

ch2 { "o3l4e f# d# f " }

ch3 { "o2l4g a f# g#" }

ch4 { "r1" }

ch5 { "L16 r4 o2 g# a a# b> c c# d d# e f f# g" }

ch6 { "L16 r4 o2 r4 g# a a# b>c c# d d#" }

ch9 { "o1L16 drdrdrdrdrdrdr" }

ch1 { "l8 o4 c r8 <a# e g a >c r" }

ch2 { "l8 o3 d#r8 c# <b d e g r" }

ch3 { "l8 o2 f#r8 e e g a >c r" }

ch4 { "l8 o3 d#efed#efe" }

ch5 { "r1" }

ch6 { "r1" }

ch9 { "o1L16 ddrdrdrdrdrdrdr" }

ch1 { "o3e d f d# f d# r f" }

ch2 { "o2b a >c <a# >c <a# r >c" }

ch3 { "o2e d f d# f d# r f" }

ch4 { "r1" }

ch5 { "L16 o3 erererererererer" }

ch6 { "L16 o2 brbrbrbrbrbrbrbr" }

ch9 { "o1L16 ddrdrdrdrdrdrdr" }

ch1 { "r f r f r4 r4" }

ch2 { "r c r c r4 r4" }

ch3 { "r f r f r4 r4" }

ch4 { "r1" }

ch5 { "r1" }

```

ch6{"r1"}
ch9{ "o1L16 ddrdrdrdrdrdr"}
//メイン
ch1 { "o3 l1 d "}
ch2 { "o2 l1 a "}
ch3 { "o2 l1 d "}
ch4 { "o3 L1 f#"}
ch5{ "L16 o3 f#r#f#r#f#r#f#r#f#r"}
ch6{ "L16 o2 ararararararar"}
ch9{ "o1L16 ddcr drcr ddcr drcr"}
ch1 { "c" } //7
ch2 { "g" }
ch3 { "c" }
ch4 { "f"}
ch5{ "L16 o3 grgrgrgrgrgrgr"}
ch6{ "L16 o2 brbrbrbrbrbrbr"}
ch9{ "o1L16 ddcr drcr ddcr drcr"}
ch1 { "<b  "}
ch2 { " f# "}
ch3 { "<b  "}
ch4 { "e"}
ch5{ "L16 o3 f#r#f#r#f#r#f#r#f#r"}
ch6{ "L16 o2 ararararararar"}
ch9{ "o1L16 ddr drr ddr drr"}
ch1 { "l4 >c <a2 a"} //9
ch2 { "l4 g e2 e"}
ch3 { "l4 >c <a2 a"}
ch4 { "e"}
ch5 { "L16 o3 crcr crcr crcr crcr"}
ch6 { "L16 o2 erer erer erer erer"}
ch9{ "o1L16 ddr drr ddr drr"}
ch1 { ">d2 L8 d <b >c d"} //10
ch2 { " a2 L8 a f# g a"}
ch3 { ">d2 L8 d <b >c d"}
ch4 { "f#"}
ch5 { "L8 o3 drdrdrdr"}
ch6 { "L8 o2 f#r#f#r#r"}
ch9{ "o1L16 drdr drr drdr drr"}
ch1 { "o3 L2 e d"} //11
ch2 { "o2 L2 b a"}
ch3 { "o2 L2 e d"}
ch4 { "o3 g"}
ch5 { "L8 o2 brbrbrbr"}
ch6 { "L8 o2 erer erer"}

```

```

ch9 { "o1L16 drdr drrr drdr drrr"}
ch1 { "c < b"} //12
ch2 { "g f#"}
ch3 { "c < b"}
ch4 { "f#"}
ch5 { "L8 o2 brbrbrbr"}
ch6 { "L8 o2 erererer"}
ch9 { "o1L16 drdr drrr drdr drrr"}
ch1 { "o2 L1 a"} //13
ch2 { "o2 L1 e"}
ch3 { "o1 L1 a"}
ch4 { "f#"}
ch5 { "L16 o3 crcrercrcrcrcr"}
ch6 { "L16 o2 erererererererer"}
ch9 { "o1L16 drcr drcr drcr drcr"}
ch1 { "a"} //14
ch2 { "e"}
ch3 { "a"}
ch4 { "e"}
ch5 { "L16 o2 ararararararar"}
ch6 { "L16 o2 c#rc#rc#rc#rc#rc#rc#r"}
ch9 { "o1L16 drcr drcr drcr drcc"}
ch1 { "g"} //15
ch2 { "d"}
ch3 { "g"}
ch4 { "e"}
ch5 { "L16 o2 grgrgrgrgrgrgr"}
ch6 { "L16 o1 brbrbrbrbrbrbrbr"}
ch9 { "o1L16 drcr drcr drcr drcr"}
ch1 { "o3 L4 c < b a b"} //16
ch2 { "o2 L4 g f# e f#"}
ch3 { "o2 L4 c < b a b"}
ch4 { "g"}
ch5 { "L8 o3 crcrercr"}
ch6 { "L8 o2 erererer"}
ch9 { "o1L16 drcc drcr drcc drcr"}
ch1 { "o2 L1 a"} //17
ch2 { "o2 L1 e"}
ch3 { "o1 L1 a"}
ch4 { "a"}
ch5 { "L8 o2 arararar"}
ch6 { "L8 o2 c#rc#rc#rc#r"}
ch9 { "o1L16 drcc drcr drcc drcr"}
ch1 { "a"} //18

```

```

ch2 { "e"}
ch3 { "a"}
ch4 { "a"}
ch5 { "L8 o2 arararar"}
ch6 { "L8 o2 c#rc#rc#rc#r"}
ch9 { "o1L16 drcr drcr drcr drcr"}
ch1 { "o3 e"}
ch2 { "o2 b"}
ch3 { "o2 e"}
ch4 { "g#"}
ch5 { "L8 o3 erererer"}
ch6 { "L8 o2 g#rg#rg#rg#r"}
ch9 { "o1L16 drcr drcr drcr drcr"}
ch1 { "L4 d <b >c d"} //20
ch2 { "L4 a f# g a"}
ch3 { "L4 d <b>c d"}
ch4 { "a"}
ch5 { "L16 o3 ararararararar"}
ch6 { "L16 o3 c#rc#rc#rc#rc#rc#rc#rc#r"}
ch9 { "o1L16 drdr drcr drdr drcr"}
ch1 { "o3 L1 g a e"} //21 22 23
ch2 { "o3 L1 d e <b"}
ch3 { "o2 L1 g a e"}
ch4 { "o3 L1 g f# g#"}
ch5 { "L16 o3 grgrgrgrgrgrgr"}
ch6 { "L16 o2 brbrbrbrbrbrbrbr"}
ch9 { "o1L16 drdr drcr drdr drcr"}
ch5 { "L16 o3 ararararararar"}
ch6 { "L16 o3 c#rc#rc#rc#rc#rc#rc#rc#r"}
ch9 { "o1L16 drcr drdr drcr drdr"}
ch5 { "L8 o3 erererer"}
ch6 { "L8 o2 g#rg#rg#rg#r"}
ch9 { "o1L16 drcr drdr drdr drcc"}
ch1 { "L2 d f e d4 c4"} //24 25
ch2 { "L2 a >c<b a4 g4"}
ch3 { "L2 d f e d4 c4"}
ch4 { "L1 a a"}
ch5 { "L8 o3 grgrg#rg#r"}
ch6 { "L8 o2 brbr>crdr"}
ch5 { "L8 o3 erergrgr"}
ch6 { "L8 o2 g#rg#r>drdr"}
ch9 { "o1L16 drcc drcr drcc drcr"}
ch9 { "o1L16 drcc drcr drcc drcr"}
ch1 { "L8 e c d1 f2 L4 g"} //26 27

```

```

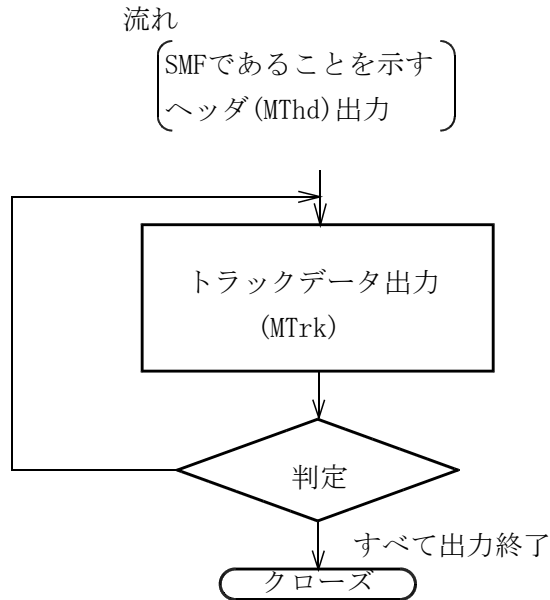
ch2 { "L8 b g a1>c2 L4 d"}
ch3 { "L8 e c d1 f2 L4 g"}
ch4 { "f#&f#4 f1"}
ch5 { "L8 o3 arararar"}
ch6 { "L8 o2 erererer"}
ch5 { "L8 o3 arararar"}
ch6 { "L8 o2 erererer"}
ch9 { "o1L16 drcr drcc drdr drdr"}
ch9 { "o1L16 drcc drcr drcc drcr"}
ch1 { "e g b2"} //28
ch2 { "<b >d f#2"}
ch3 { "e g b2"}
ch4 { "f#2&f#4"}
ch5 { "L16 o3 ererererererrr"}
ch6 { "L16 o2 brbrbrbrbrbrrrr"}
ch9 { "o1L16 drcc drcr drcc drcr"}
ch1 { "o3 L1 e"}
ch2 { "o2 L1 b"}
ch3 { "o2 L1 e"}
ch4 { "o3 L1 f#"}
ch5 { "L16 o3 crcrercrcrerrr"}
ch6 { "L16 o2 grgrgrgrgrgrrrr"}
ch9 { "o1L16 drdr drrr drrr drrr"}
ch1 { "<b4 a4 >c8 e2 b1 a1 r8 r2r4" } //30 31 32 33
ch2 { "f#4 e4 g8 <b2> f#1 e1 r8 r2r4" }
ch3 { "<b4 a4 >c8 e2 b1 a1 r8 r2r4" }
ch4 { "g&g8 a > g8&g2&g4 < L8 c#dc#cc#dc#c"}
ch5 { "L8 o3 c#rc#rc#rc#r"}
ch6 { "L8 o2 f#rf#rf#rf#r"}
ch5 { "L8 o3 erererer ererererr"}
ch6 { "L8 o2 grgrgrgr grgrgrrr"}
ch5 { "L16 o3 brbrbrbrbrbrbrbr"}
ch6 { "L16 o3 c#rc#rc#rc#rc#rc#rc#rc#r"}
ch9 { "o1L16 drrr drrr drrr drrr drrr drrr drrr drrr"}
ch9 { "o1L16 drrr drrr drrr drrr drcr drcr drcr drcr"}
ch1 { "r1" } //34
ch2 { "r1" } //34
ch3 { "r1" } //34
ch4 { "r1" } //34
ch5 { "r1" } //34
ch6 { "r1" } //34
ch9 { "o1L16 drdr drdr drdr drdr"}

```

※一番左が 0 で、64 が中央、127 は一番右から音を出力することになります

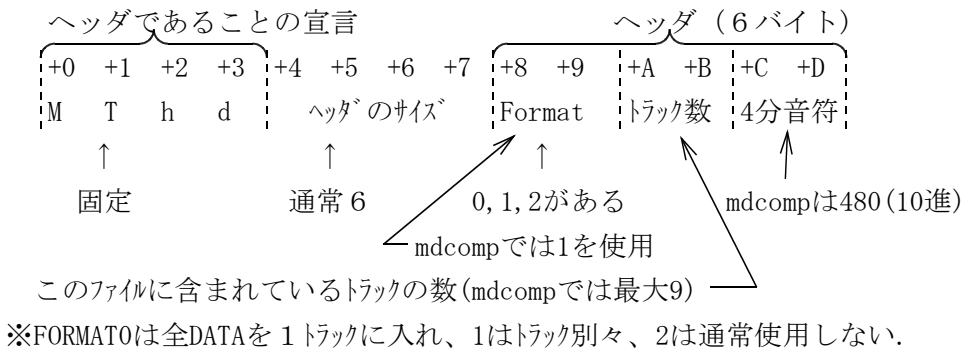
■スタンダードMIDIファイル(SMF)

mdcomp(mdcomp32)は FORMAT 1 の形式で出力しています。FORMAT 0 は1トラックに全パートのデータを組み込む形式で、FORMAT 1 は各チャンネルごとにトラックをもうける形式です。以下に mdcomp で作成する SMF について説明します。

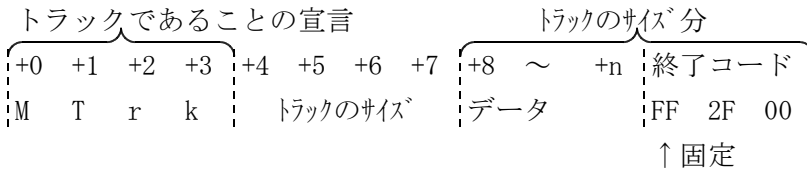


トラックデータ出力は、データ出力後サイズ計算を行いトラック開始直後にあるサイズ保管場所にシークし書き直します。また、サイズ書き込み後はファイル最後へ再シークします。

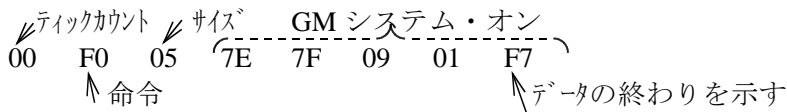
■ヘッダ(MThd)



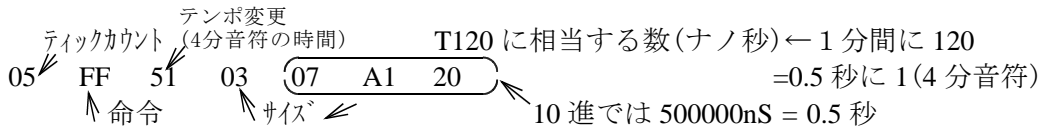
■トラック(MTrk)



mdcomp では一番はじめのトラックに GM システム・オンを入れます。



さらに、GM システム・オン後にはテンポも 120 に相当するスピードに初期化するため以下のコードを実行します。



テンポを計算する方法

$$\frac{T}{60} \times 1000000$$

例えば

$$\begin{aligned}
 & T=120 \text{ の時} \\
 & \quad \frac{120}{60} \times 1000000 \\
 & = 0.5 \times 1000000 \\
 & = 500000 = 07A120(16)
 \end{aligned}$$

著作権表示用文字列を指定している場合は、GM システム・オンより前に以下のコードを入れます。

```
+0 +1 +2 +3 +4 .. +(n+3) +(n+4)
FF 02 n <-- 著作権文字列 --> 00
```

+0	命令
+1	著作権表示文字列指定
+2	サイズ
+3 ~ (n+3)	文字列
+(n+4)	ティックカウント

また、文字列は SMF の構造上 255 までであるが OS 標準のプレイヤーで動作検証を行った結果 63 を最大にすることにより誤動作しないことを確認した（大きいサイズでは誤認識）。SMF ではティックカウントが開始から 0 地点でこの命令が出現しなければならない。

■トラック

二番目以降に出現したトラックにはダミーデータを先頭に入れます。

00 B? 78 00

ダミーデータ (?にはチャンネル)

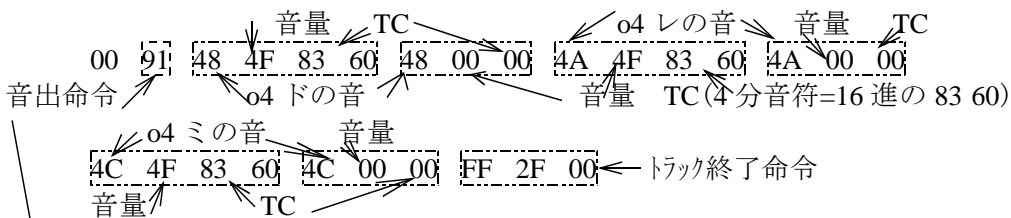
最初のトラック

初期化命令 (GM システム・オン、テンポ 120) が完了したら

音階コードを入れる

↓TC ← この数値がチャンネル番号 (0 から F までの 16) * TC はティックカウントの略
 00 C1 00 ← 楽器コード ! 先頭に楽器変更命令を入れる (すべて 00 番)
 ↑チャンネル指定 ↑通常ピアノ

音階ドレミを発声する時の例 (4 分音符)



音を出すには音階コードと音量および TC を指定する。

↑音を終わらせるには発声した音階コードと音量 0 を指定する。これで 1 音。

音を出せという命令は 9? となっており ? にはチャンネル番号が入ります。命令がつながる場合は、9? を先頭に 1 つおき、音階コードを連続することが可能です。なお C? 命令により楽器番号変更時はもう一度 9? 命令を出さないと音がそこで停止します。

■ティックカウント

通常時間の流れを示すもので4分音符がティックカウント 480(mdcomp32 では)と定義し、 $T=120=4$ 分音符は 0.5 秒とすればティックカウント 1 は

$$\frac{0.5\text{秒}}{480} \times 1 \div 0.0010417\text{秒となります。}$$

ティックカウントは7ビットコードで上位から保存します。
例えば、128(10)と 127(10)の2つを7ビットコードで示すと

	8ビット	7ビット	7ビット
	↓		
128(10)	1000 0000 _B	0000001	0000000 _B
127(10)	0111 1111 _B		1111111 _B

となりますが、SMF では7ビットコードの最上位に8ビット目の続有無ビットを付加し8ビットとしている上記の128(10)をSMFの7ビットコードであらわすと

128(10)	1000 0001	0000 0000 _B
	↑	↑
	続きがある	続きはない
	000 000	1 0000000 _B

となるためティックカウントが128以上になるときは注意が必要です(16384以上で同じようなことがまたおきる)。

127(10)	0111 1111 _B	〔ティックカウント以外では8ビットコードの 時があるので注意〕
	↑	
	続きはない	

■mdcompエラー内容一覧

番号	エラー時の表示内容
1	解説できない命令があります
10	&は連続で並べられません
11	&を単独では利用できません
12	命令 N? の範囲は 0 ～ 96
13	命令 O? の範囲は 0 ～ 7
14	命令 L? で不可数値を使用
15	命令 @? の範囲は 0 ～ 127
16	命令 V? の範囲は 0 ～ 16
17	命令 T? の範囲は 10 ～ 250
18	命令 : の範囲は 1 ～ 255
19	命令 Q? の範囲は 1 ～ 8
20	命令 {} 連符指定に誤りがある
21	命令 \$? の範囲は 0 ～ 127
22	op は ')'で終わる必要があります
23	op で 該当する制御命令はありません
24	cr は ')'で終わる必要があります
25	文字列は 63Byte まで(超過違反)
26	sn は ')'で終わる必要があります
27	文字列は 127Byte まで(超過違反)
999	ch をオープンしていません
他	その他のエラー発生

mdcomp で GM 規格音源使用時の対応表 (ch1 ~ ch8 で使用可)

番号	mdcomp	楽 器 名	番号	mdcomp	楽 器 名
1	@0	ピアノ系	36	@35	バス系
2	@1	ピアノ系	37	@36	バス系
3	@2	ピアノ系	38	@37	バス系
4	@3	ピアノ系	39	@38	バス系
5	@4	ピアノ系	40	@39	バス系
6	@5	ピアノ系	41	@40	バイオリン
7	@6	ハープシーコード	42	@41	ビオラ
8	@7	クラビコード	43	@42	チェロ
9	@8	チェレスタ	44	@43	コントラバス
10	@9	グロッケンシュピール	45	@44	ストリング系
11	@10	オルゴール	46	@45	ストリング系
12	@11	ビブラフォン	47	@46	ハープ
13	@12	マリンバ	48	@47	ティンパニ
14	@13	シロフォン	49	@48	ストリング系
15	@14	チューブラーベル	50	@49	ストリング系
16	@15	ダルシマー	51	@50	ストリング系
17	@16	オルガン系	52	@51	ストリング系
18	@17	オルガン系	53	@52	声
19	@18	オルガン系	54	@53	声
20	@19	オルガン系	55	@54	声
21	@20	オルガン系	56	@55	オーケストラヒット
22	@21	アコーディオン	57	@56	トランペット
23	@22	ハーモニカ	58	@57	トロンボーン
24	@23	タンゴアコーディオン	59	@58	チューバ
25	@24	ギター系	60	@59	トランペット系
26	@25	ギター系	61	@60	ホルン
27	@26	ギター系	62	@61	ブラス系
28	@27	ギター系	63	@62	ブラス系
29	@28	ギター系	64	@63	ブラス系
30	@29	ギター系	65	@64	サックス系
31	@30	ギター系	66	@65	サックス系
32	@31	ギター系	67	@66	サックス系
33	@32	バス系	68	@67	サックス系
34	@33	バス系	69	@68	オーボエ
35	@34	バス系	70	@69	ホルン系

mdcomp で GM 規格音源使用時の対応表 (ch1 ~ ch8 で使用可)

番号	mdcomp	楽 器 名	番号	mdcomp	楽 器 名
71	@70	バスーン	106	@105	バンジョー
72	@71	クラリネット	107	@106	三味線
73	@72	ピッコロ	108	@107	琴
74	@73	フルート	109	@108	カリンバ
75	@74	リコーダー	110	@109	バグパイプ
76	@75	フルート系	111	@110	フィドル
77	@76	瓶	112	@111	シャーナイ
78	@77	尺八	113	@112	ティンクルベル
79	@78	口笛	114	@113	アゴゴ
80	@79	オカリナ	115	@114	ドラム系
81	@80	リード系	116	@115	木琴
82	@81	リード系	117	@116	太鼓
83	@82	リード系	118	@117	メロディックタム
84	@83	リード系	119	@118	ドラム系
85	@84	リード系	120	@119	シンバル系
86	@85	リード系	121	@120	ギターフレットノイズ
87	@86	リード系	122	@121	ブレスノイズ
88	@87	リード系	123	@122	波の音
89	@88	パッド系	124	@123	鳥のさえずり
90	@89	パッド系	125	@124	電話
91	@90	パッド系	126	@125	ヘリコプター
92	@91	パッド系	127	@126	UFO
93	@92	パッド系	128	@127	鉄砲
94	@93	パッド系			
95	@94	パッド系			
96	@95	パッド系			
97	@96	FX系			
98	@97	FX系			
99	@98	FX系			
100	@99	FX系			
101	@100	FX系			
102	@101	FX系			
103	@102	FX系			
104	@103	FX系			
105	@104	シタール			

mdcomp で GM 規格音源使用時の対応表 (ch9 で使用)

キー	mdcomp	N	パーカッション名		キー	mdcomp	N	パーカッション名			
35	o0	B	12	ドラム系		70	o3	A#	47	マラカス	
36	o1	C	13	ドラム系		71	o3	B	48	ホイッスル系	
37	o1	C#	14	サイドスティック		72	o4	C	49	ホイッスル系	
38	o1	D	15	スネア系		73	o4	C#	50	ギロ系	
39	o1	D#	16	手		74	o4	D	51	ギロ系	
40	o1	E	17	スネア系		75	o4	D#	52	クラベス	
41	o1	F	18	タム系		76	o4	E	53	ウッドブロック系	
42	o1	F#	19	ハイハット系		77	o4	F	54	ウッドブロック系	
43	o1	G	20	タム系		78	o4	F#	55	クイーカ系	
44	o1	G#	21	ハイハット系		79	o4	G	56	クイーカ系	
45	o1	A	22	タム系		80	o4	G#	57	トライアングル系	
46	o1	A#	23	ハイハット系		81	o4	A	58	トライアングル系	
47	o1	B	24	タム系							
48	o2	C	25	タム系							
49	o2	C#	26	シンバル系							
50	o2	D	27	タム系							
51	o2	D#	28	シンバル系							
52	o2	E	29	シンバル系							
53	o2	F	30	ベル系							
54	o2	F#	31	タンバリン							
55	o2	G	32	シンバル系							
56	o2	G#	33	カウベル							
57	o2	A	34	シンバル系							
58	o2	A#	35	ビブラスラップ							
59	o2	B	36	シンバル系							
60	o3	C	37	ボンゴ系							
61	o3	C#	38	ボンゴ系							
62	o3	D	39	コンガ系							
63	o3	D#	40	コンガ系							
64	o3	E	41	コンガ系							
65	o3	F	42	ティンバル系							
66	o3	F#	43	ティンバル系							
67	o3	G	44	アゴゴ系							
68	o3	G#	45	アゴゴ系							
69	o3	A	46	カバサ							

※ ch9 は Panpot 命令で変更しなくても標準の状態です。
※ N 列は mdcomp で使用できる直音命令の対応番号

※ ch9 は Panpot 命令で変更しなくても標準の状態です。値が決められています。

※ N 列は mdcomp で使用できる直音命令の対応番号

都合上詳細を記載しませんが、別途インターネット等で General MIDI の規格をご覧ください。

ソフトウェア名	mdcomp32 (Standard MIDI File Compiler) 32bit アプリケーション
対応OS	Microsoft Windows 7 / 10 (10 のみ 64bit 版も作動確認済)
最大チャンネル数	9 チャンネル(ch9 はリズム音として使用) ※ MIDI チャンネルは 2 から 10 まで使用(MT-32 等で演奏可)
出力形式	スタンダード MIDI ファイル(FORMAT 1)
対応形式	テキストファイルのみ
入力限界	OS 標準のメモ帳限界と同等
出力限界	作成限界は各チャンネルの作成限界を加算したもの (各チャンネル 60 キロバイト程度まで:約 30000 命令)
設定保持	cfg ファイルに設定を保存可能
処理方式	オリジナルコード分解方式(入力解析) FM コンパイラ方式(DOS 版)
技術情報 (搭載)	FMCOMP アルゴリズム (DOS 版/16bit および 32bit) DEC2ASC (KMSWX32 v2.00) RD1LINE エミュレータ 分母・余り交換アルゴリズム (v0.00.05 ~)

商品名および社名は各社の登録商標または商標です

■約束

著作権およびその他の権利は wnc / t.w.にあります。自由に配布または複製および使用・利用できます。

mdcomp32 は無償で提供しています。

mdcomp32 は無制限に複製・配布可能です（ご連絡は不要です）。

不具合については次回更新時に可能であれば修正します。商用ソフトウェアではありませんのでいかなる損害や動作・修正を保証しません。すべて使用者の責任で使用してください。

インターネットメール wncdevelop@yahoo.co.jp

Web 頁 (wnc inu goya) <http://hp.vector.co.jp/authors/VA055892/>

■mdcomp32固有の約束

mdcomp32 (スタンダード MIDI ファイルコンパイラ) で作成したデータ (SMF) の著作権は作曲者および音楽ソースの作成者にあります。よって各個人の責任において自由に配布（販売を含む）および複製することができます。作曲者と音楽ソース作成者が異なる場合は作曲者の権利を侵害しないように注意してください。特に、作成したデータを商用または個人的以外で使用および配布する場合は作曲者に書面による承諾や JASRAC の使用許諾が必要になることがあります。

wnc / t.w.は一切関与しません。

■曖昧回避のための項目

※ 1:商用利用可（重要事項および約束をご覧ください）

※ 2:商用利用には学校・塾・企業・パソコンスクールでの使用、雑誌・書籍の付録を想定しています（連絡不要）

※ 3:画面のハードコピー・キャプチャも使用可（連絡不要）

※ 4:mdcomp32 説明書に使用している楽曲は転用不可

※ 5:犯罪になることはダメです

*この項目は都合により変更する場合があります

□バンドル（プリインストール）

ないとは思いますが、mdcomp32 をインストールして販売や配布しても良い（※ 1,5 厳守）（連絡不要）。

※諸事情により約束の内容が変更になる場合があります。

■おしまい

mdcomp32 v1.01 は
Borland TurboC++2006 Explorer で開発しています。

htmlhelp を製作するに当たって
HelpMaker VII

This is Licensed Material and the Property of Vizacc.
Copyright c 2008-2017 by Vizacc. All rights reserved.
Copyright c 1998-2007 by Vizacc. All rights reserved.
The product version is: 7.4.4 Beta Release
を使用しています。

[注*]2017 年 9 月現在 Vizacc がいないため確認できませんでした。

■参考書類

PC98-NX VS26 添付 VSC-88 のオンラインマニュアル (ローランド社)
B# MIDI 規格 1.0 のページ (Jun Arai さん製作) <2001 年 9 月 5 日時点>
<http://www.b-sharp.com>

注) 2017 年 9 月に閲覧したところ表示できませんでした

その他多数...
ありがとうございます。

■mdcomp32.EXE履歴

<2001. 01. 20p>v0. 00. 00 完成
 <2001. 01. 27p>v0. 00. 00 追加(/* */ および &)
 <2001. 02. 04p>v0. 00. 01 1Mまで編集対応, compile高速化, 編集機能改良, 画面一部変更
 <2001. 02. 12p>v0. 00. 02 自動上書き保存機能追加
 <2001. 03. 04a>v0. 00. 03 長さ直接指定機能追加, ch関係修正
 <2001. 03. 09a>v0. 00. 04 行番号表示機能追加, RichEdit関係修正
 <2001. 03. 10p>v0. 00. 04a 設定保持機能追加, GM SYSTEM ON強制出力追加
 <2001. 03. 16p>v0. 00. 04b RichEdit関係修正(スクロールバーバグ封じ)
 <2001. 03. 26p>v0. 00. 05 Q命令追加, 連符機能追加, 初期設定修正
 <2001. 08. 28p>v0. 00. 05c ファイルのドラッグ&ドロップによる読み込み追加
 <2001. 08. 30p>v0. 00. 06 Panpot命令に対応(音を左右に振り分け), 連符修正
 <2001. 09. 06p>v0. 00. 07 自動左右配分機能追加
 <2001. 09. 08p>v0. 00. 08 op命令追加(AUTOLR, MANULR, Q7DEF, Q8DEF)
 <2001. 12. 31p>v0. 00. 09 著作権表示文字列組込可能(cr命令追加)
 <2002. 02. 12p>v0. 00. 10 シーケンス名組込可(sn命令追加), 2000Pro対応
 <2002. 06. 11p>v0. 00. 10a 自動backup機能(180秒毎), 更新前bak機能追加
 <2013. 02. 22p>v1. 00. 00 再構築(RichEdit関連と行番号表示削除)
 <2017. 10. 04p>v1. 01. 00 fmcompデータ出力追加, SMF出力修正

■mdcomp32.chm(HLP)履歴

<2001. 01. 20p>v0. 00. 00-00 完成(v0. 00. 00同時配布用)
 <2001. 01. 27p>v0. 00. 00-00 追加
 <2001. 02. 04p>v0. 00. 01-00 変更
 <2001. 02. 12p>v0. 00. 02-00 機能追加による絵差し替え
 <2001. 03. 04a>v0. 00. 03-00 機能追加による文面追加
 <2001. 03. 09a>v0. 00. 04-00 機能追加による文面及び絵追加
 <2001. 03. 10p>v0. 00. 04-01 説明及び絵追加
 <2001. 03. 16p>v0. 00. 04-02 履歴等追加
 <2001. 03. 26p>v0. 00. 05-00 機能追加による文追加
 <2001. 09. 23p>v0. 00. 08-00 機能追加による文追加
 <2001. 12. 31p>v0. 00. 09-00 機能追加による文追加
 <2002. 02. 13p>v0. 00. 10-00 機能追加および改良による文追加
 <2002. 06. 11p>v0. 00. 10-01 機能追加による文追加
 <2013. 02. 25p>v1. 00. 00-00 htmlhelp再構築
 <2017. 10. 05p>v1. 01. 00-00 修正変更

■開発元

win32 (GUI) 版 : wnc develop

MADE IN JAPAN.

mdcomp32 取扱説明書

初 版	平成 1 3 年 3 月 5 日
改 訂 8	平成 2 9 年 10 月 5 日

製品名および会社名は各社の登録商標または商標です.