

# *Wonder Swan Total Sound Driver*

*Version 1.0a*

*User's Manual*

*Digitalis*

# 始めに

この度は、Wonder Swan Total Sound Driver(以後、WTD と略します。)をご購入いただきありがとうございます。

本製品は、Wonder Swan(以後、WS と略します。)用開発ツール Wonder Witch において、統合的な音源環境を構築すべく開発しました。WS 内蔵 PCM 音源、外部 MIDI 音源の制御が可能ですが、今後の拡張性も備えています。

本製品のドライバーは、全てアセンブリ言語によって記述することで、処理速度を向上させ、できる限り割り込み処理が軽くなるように設計されています。演奏 20 トラック、効果音 3 トラックの同時発生にもドライバーの重さを感じさせません。(しかし、PCM Voice 発生時だけは頻度に割り込み処理を行う必要があります、処理速度が低下致しますが、なにとぞご了承下さい。)

本ドライバーを製作するに当たって多くの方にお世話になりました。

- ・ (株)バンダイ様、(有)キュート様

Wonder Witch という素晴らしい開発環境を与えていただきました。

- ・ 都立高専エレクトロニクス研究部様

測定器具の貸与、測定・実験方法の提案等をしていただきました。

この場を借りて、感謝の意を述べたいと思います。

2001.05.29 Digitalis

渡部 篤史 (S.W.)

渡辺 伊織 (FUN-X)

Wonder Swan は、株式会社バンダイの登録商標です。

Wonder Witch は、株式会社バンダイの登録商標です。

Freya BIOS , Freya OS, Trans Magic は、有限会社キュートの登録商標です。

MS-DOS,MS-Windows は、米国 Microsoft Corporation 及びその他の登録商標です。

Macro Assembler は、米国 Microsoft Corporation 及びその他の登録商標です。

LSI-C86 は、エル・エス・アイ・ジャパン株式会社の登録商標です。

# 1 . 目次

1 . 目次	-----	p. 2
2 . 付属品	-----	p. 3
3 . 仕様	-----	p. 4
3 - 1 . ドライバー制御		p. 4
3 - 2 . 演奏・効果音制御		p. 5
3 - 3 . PCM Voice 制御		p. 6
4 . 付属ツールについて	-----	p. 7
4 - 1 . プレイヤー		p. 7
4 - 1 - 1 . 準備		p. 7
4 - 1 - 2 . 操作方法		p. 7
4 - 1 - 3 . ビューモード		p. 7
4 - 1 - 4 . 注意		p. 7
4 - 1 - 5 . 戯言		p. 7
4 - 2 . MML コンパイラ	-----	p. 8
4 - 2 - 1 . コマンドライン書式		p. 8
4 - 2 - 2 . MML 文法		p. 9
4 - 2 - 3 . Wonder Witch への転送・演奏方法		p. 42
4 - 3 . ROM イメージメーカ	-----	p. 43
4 - 3 - 1 . WonderSwan エミュレータについて		p. 43
4 - 3 - 2 . コマンドライン書式		p. 43
5 . 参考文献	-----	p. 44

## 2 . 付属品

本製品は3つのエディションがございます。

・ **Standard Edition ... (SE)**

曲データの作成・演奏が可能です。開発にはご利用できません。

・ **Professional Edition ... (PE)**

主に、本製品を用いたアプリケーションの作製・開発にご利用頂けます。

・ **Developer's Edition ... (DE)**

本ドライバー開発時の資料一式です。

文書、電子データになっている資料のみです。

表 2-1 Wonder Swan Total Sound Driver 付属品リスト

	<i>SE</i>	<i>PE</i>	<i>DE</i>	再配布
Driver(内蔵 PCM のみ)				
Driver(内蔵 PCM + 外部 MIDI)				
MML Compiler				
Music Selector & Viewer for WTD				
User's Manual				
Developer's Manual				
WTD Software Development Kit				
WTD Source list				
開発時の資料(HCP Chart,他)				

( 各々の価格は、最後のページを参照して下さい。 )

## 3 . 仕様

### 3 - 1 . ドライバー

本ドライバーは、ライブラリ(.lib ファイル)及び、WonderWitch 用のインダイレクトライブラリ（以後、'IL'と略す）として作製されています。ユーザは、制御用関数を使用することで、本ドライバーの制御が可能となります。

本ドライバーは、アセンブリ言語にて作製することで、処理速度の向上・効率化を図っています。内部では、以下に示すデバイスを使用しており、ドライバーの常駐時にユーザはこれらのデバイスにアクセスをしてはいけません。

- ・ サウンド関連（サウンド DMA 含む）
- ・ シリアル通信関連（wtd\_h.fx の使用時は、アクセス可能です。）
- ・ HBLANK タイマー割り込み関連

IL 版のドライバーについては、ユーザが Freya OS 及び、Freya BIOS を使用することを前提に作製しています。故に、本ドライバーを使用するプログラムは Freya OS 及び Freya BIOS の規格に準拠したものである必要があります。又、制御用ワークエリアとしてユーザプロセス用 SRAM を要求します。制御用ワークエリアは、ヒープ領域の先頭から 4096byte の領域を使用します。

ライブラリ版のドライバーについては、WonderWitch を必要としません。DGROUP を I-RAM(DS=0x0000)に確保している場合は I-RAM 版、DGROUP を S-RAM(DS=0x1000)に確保している場合(WonderWitch 等)は S-RAM 版を使用してください。又、制御用ワークエリアは、セグメント DGROUP に 4096Byte のグローバル変数を確保します。

本ドライバーは、以下の機能を有します。

- ・ ドライバー常駐 / 常駐解除機能を有します。
- ・ WS 内蔵 PCM と、外部 MIDI(38.4kbps)の同時制御が可能
- ・ 音楽、効果音、PCM Voice の同時制御が可能
- ・ 音楽、効果音、PCM Voice について、個別に演奏開始及び停止、又、ソフトウェアボリュームを有します。

## 3 - 2 . 演奏・効果音制御

本ドライバーは、HBLANKタイマー割り込みを用いて演奏、効果音制御を行っている。演奏及び、効果音は、以下の機能を有す。<sup>1</sup>

- ・ 4トラックのWS内蔵PCMの制御
- ・ 16トラックの外部MIDI音源(38.4Kbps)の制御
- ・ MIDI音源に指定したトラックは、最大8和音まで発生することができる。
- ・ 16個のWS内蔵PCM用の音色をドライバーに記憶させることができる。
- ・ MIDI音源の、バンクセレクトとプログラムチェンジに対応。
- ・ WS内蔵PCMは、テーブルを用いた対数演算によって、128段階の音量及びパンポットを実現させ、MIDI音源にレンジを合わせた。
- ・ WS内蔵PCMは、音階テーブルを持たせることで、キーレベル(ドレミ...)での制御にし、MIDI音源と制御ルーチンを同一にした。
- ・ 音量、音程、パンポットの制御にそれぞれLFOとエンベロープ、ディチューンを有す。MIDIの音程制御はベンドによって行われ、音量制御はエクスプレッションによって行われる。ディチューン等の音程制御をMIDI音源のレンジに合わせる処理は、複雑な計算を必要とするため行っていない。
- ・ HBLANKタイマー割り込みの周期を用いたテンポ制御。効果音のテンポは、現在演奏している曲のテンポと同一になる。
- ・ MIDIに送信するデータを直接記述することができる。この時、エクスクループ等のチェックサムの計算はユーザが行うものとする。MMLコンパイラレベルではチェックサムの計算に対応している。

### <注意>

“wtd\_h.fx”は、MIDI機能を削除しユーザがシリアル通信をいつでも使用できるようにしたドライバーです。“wtd\_h.fx”の使用時は、MIDI機能を使用することができません。

1 効果音の場合は、以下に示す機能は無い。

- ・ MIDI制御関連
- ・ テンポ制御関連

### 3 - 3 . PCM Voice 制御

本ドライバーは、HBLANK タイマー割り込みを用いて、PCM Voice 処理も行っている。PCM Voice は、以下の機能を有す。

- ・ HBLANK タイマー割り込み周期を最速にし、WS で可能な最大のサンプリングレート(12[kHz))での PCM Voice が発生可能。この時、タイマー割り込み処理は非常に重くなるので、メインの処理速度に及ぼす影響が大きいことを留意してプログラムを作成すること。
- ・ PCM Voice 発生中は、演奏・効果音処理は擬似的なタイマーを作動させるため、独立した発生が可能。但し、演奏・効果音処理のテンポ落ちは発生する。
- ・ Microsoft 社の'wave'ファイル(12[kHz],モノラル,8[bit])に対応する。

## 4 . 付属ツールについて

### 4 - 1 . プレイヤー

このソフトはワンダーウィッチ用のサウンドドライバ「WTD」に対応したミュージックプレイヤー＆ビューアーです。

#### 4 - 1 - 1 . 準備

ワンダーウィッチに「WMSVWTD.FX」と「WTD.IL」と WTD の演奏ファイルを/ROM0/ディレクトリに転送してください。

#### 4 - 1 - 2 . 操作方法

X1 ~ X4	: 演奏ファイルの選択
Y2,Y4	: ビューモード変更
Y1,Y3	: トラックのスクロール
A	: 再生
B	: 停止
START	: 終了

#### 4 - 1 - 3 . ビューモード

##### a.) 鍵盤モード

鍵盤付きで音階を表示します。

##### b.) ステータスモード

演奏データのメモリの状態を表示します。

演奏を再生させるだけに使う方には、必要の無いモードです。

#### 4 - 1 - 4 . 注意

たぶん無いと思いますが、このソフトを使用して FreyaOS、FreyaBIOS 他が壊れてしまっても作者は一切の責任を取りません。大切なファイルはちゃんとバックアップを取っておきましょう。

#### 4 - 1 - 5 . 戯れ言

まだまだ Wonder Witch で曲データ作ってくれる人募集。



## 4 - 2 . MML コンパイラ

### 4 - 2 - 1 . コマンドライン書式

MS-DOS のコマンドプロンプト上で、以下のように記述します。

WTC [filename(.mml)]

すると、指定した MML ファイルを wtd で演奏できる形式に変換します。

曲データをコンパイルして作成したら、Wonder Witch 転送形式(“FX”ファイル)に変換する必要があります。MML 中の”FX”コマンドにて 1 を指定することで Wonder Witch 転送形式を出力する事が可能です。それ以外の場合は、Wonder Witch 付属の”mkfent.exe”を使用して、Wonder Witch 転送形式に変換しなければなりません。

Wonder Witch を所有していない方は、MML 中の”FX”コマンドにて 1 を指定して下さい。

MML コンパイラレベルで Wonder Witch 転送形式に変換可能なため、チーム開発において作曲チーム及び、サウンド・チームは Wonder Witch のカートリッジのみの購入で済み、開発費が安くなるメリットもあります。

WTV 形式の音色ファイルに対応しております。別途、FUN-X 氏の”wswa.exe”(Win32 アプリケーション)を使用することで、音色ファイルの編集を行うことができます。

#### 注意事項

MML は、テキストエディタ等で編集します。本製品にはテキストエディタは付属していません。

本 MML コンパイラは、ループ命令以外の文法チェックは行っていません。

廉価版は 4 パート、通常版は 20 パートまで演奏制御が可能です。

廉価版は、MIDI 音源を使用できません。

## 4 - 2 - 2 . MML 文法

---

### ヘッダー

---

<書式> 行の先頭に書いてください。

TI	<i>Title</i>	説明文(最大半角 24 文字)	
FI	<i>Filename</i>	ファイル名(最大半角 16 文字)	
TB	<i>Time Base</i>	( 4 分音符の分割数 )	ver,0.01
PA	<i>Part</i>	( 使用するトラック数 )	ver,0.01
FX	<i>Switch</i>	Wonder Witch 転送形式ファイルを生成するか?	

<出力>

曲データのヘッダーに数値が出力されます。

Developer's Manual を参照

<初期値・範囲>

<i>Title</i>	初期値	無し	( <i>Switch</i> を 1 にした場合は必ず指定する。)
<i>Filename</i>	初期値	無し	( <i>Switch</i> を 1 にした場合は必ず指定する。)
<i>Time Base</i>	初期値	48	範囲 1 ~ 255
<i>Part</i>	初期値	0	範囲 1 ~ 20(廉価版は 4,効果音は 3)
<i>Switch</i>	初期値	0	範囲 0 (生成しない) / 1 (生成する)

<解説>

データのヘッダーを定義します。

また、これらの引数はコンパイル作業時にも影響を及ぼします。

これ以外のヘッダーは、コンパイラが自動的に計算し出力します。

---

---

---

## 拡張ヘッダ

---

現在は、使われていません。

---

---

## ファイル挿入

---

<書式>            行の先頭に書いてください。

IN     *Filename*(+改行)

ver,0.01

<出力>

コンパイル疑似命令です。

<解説>

コンパイル時にファイルを挿入します。

MS-DOS 形式のファイル名（ファイル名:8/拡張子:3 文字）で指定します。

ファイル名の終了は必ず、改行して下さい。

---

---

	WS 内蔵音色定義	PCM
<書式>	行の先頭に書いてください。	
@	<i>no</i> , <i>d0</i> , <i>d1</i> , <i>d2</i> , ... , <i>d31</i>	ver,0.01
@F	<i>no</i> , <i>no2</i> , <i>Filename</i>	ver,0.01

<出力>

Developer's Manual を参照

<書式>

曲データ中に音色データを埋め込みます。

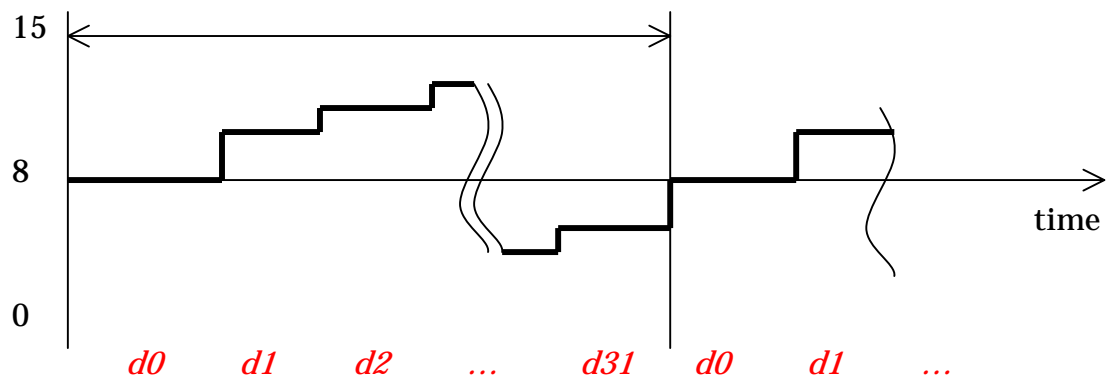
ドライバーは、演奏開始時にドライバー内部に音色データを転送します。

ドライバーの常駐解除、又は、再度データ定義するまで引き継がれます。

*no*                    0 ~ 15   音色番号  
*d0 ~ d31*           0 ~ 15   波形データ  
*no2*                 0 ~ 255 音色データ中の音色番号  
*Filename*           音色ファイル名(WTV 形式)

<図 PCM 音色波形>

Level                1 周期 T[sec]



<注意>

効果音と共通のワークエリアを使用しているので、お互いに干渉し合わないよう注意して下さい。

エンベロープ定義	PCM	MIDI
----------	-----	------

<書式> 行の先頭に書いてください。

E *no, fl, ar, dr, dl, sr, rr* ver,0.01

Ex *no, fl, ar, as, al, dr, ds, dl, sr, ss, sl, rr, rs, rl* ver,0.01

<出力>

Developer's Manual を参照

<書式>

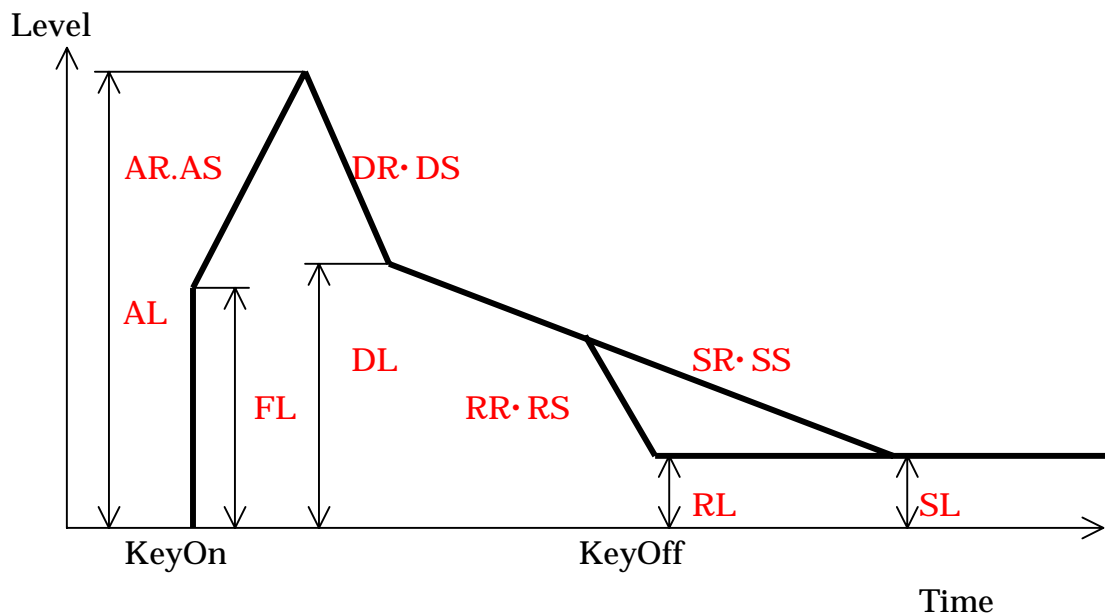
曲データ中にエンベロープデータを埋め込みます。

ドライバーは、演奏開始時にドライバー内部に音色データを転送します。

ドライバーの常駐解除、又は、再度データ定義するまで引き継がれます。

引数	名前	内容	範囲	初期値
<i>no</i>	number	音色番号	0 ~ 23	255
<i>fl</i>	First Level	Key On 時の出力レベル	0(小) ~ 255(大)	-
<i>ar</i>	Attack Rate	Key On 時立ち上がりの早さ	0(遅) ~ 255(早)	-
<i>as</i>	Attack Speed	Key On 時立ち上がりの早さ	1	1
<i>al</i>	Attack Level	Key On 時立ち上がりのレベル	0(小) ~ 255(大)	255
<i>dr</i>	Decay Rate	1 回目立下りの速さ	0(遅) ~ 255(早)	-
<i>ds</i>	Decay Speed	1 回目立下りの速さ	1	-1
<i>dl</i>	Decay Level	1 回目立下りのレベル	0(小) ~ 255(大)	-
<i>sr</i>	Sustain Rate	2 回目立下りの速さ	0(遅) ~ 255(早)	-
<i>ss</i>	Sustain Speed	2 回目立下りの速さ	1	-1
<i>sl</i>	Sustain Level	2 回目立下りのレベル	0(小) ~ 255(大)	0
<i>rr</i>	Release Rate	Key Off 後の立下りの速さ	0(遅) ~ 255(早)	-
<i>rs</i>	Release Speed	Key Off 後の立下りの速さ	1	-1
<i>rl</i>	Release Level	Key Off 後の立下りのレベル	0(小) ~ 255(大)	0
1	Speed について	増加方向	1(早) ~ 127(遅)	
		減衰方向	-1(早) ~ -128(遅)	

<図 エンベロープ形状>



<注意>

効果音と共通のワークエリアを使用しているため、お互いに干渉し合わないようご注意ください。

---

### MMLトラック指定

---

<書式> 行の先頭に書いてください。

**Cn(n,...)**

ver,0.01

<出力>

Developer's Manual を参照

<解説>

これ以降の MML は、この命令で指定した番号のトラックにコンパイルされます。

今まで指定されていた番号はすべてリセットされます。

番号は、','(カンマ)で区切ることで複数指定することができます。

トラック番号は、0 から 'PA' 命令で指定した数値分だけ使えます。

---

---

---

マクロ
-----

---

<書式>

**\$*Sn***    MML 文字列                      . . . マクロ定義 ( 行頭に書きます。 )  
          **\$*Sn***                                . . . マクロ呼び出し

<出力>

コンパイル擬似命令です。

<初期値・範囲>

***S*** . . .            範囲    a ~ z , A ~ Z ( 1 文字 )  
***n*** . . .            範囲    0 ~ 255                                省略時 0  
MML 文字列                      MML 文字列を書きます。改行まで有効です。

<解説>

行頭に書いた場合は、マクロを定義します。

MML 文字列中に書いた場合は、定義したマクロを呼び出します。

マクロのネストは、8 回までです。

---

---

ワーク・スイッチ	PCM	MIDI
----------	-----	------

---

<書式>

          !  
ver,0.01

<出力>

0x21

<解説>

次の命令の引数で、ワークを参照します。ワークの上位アドレスは"コマンドで指定し、下位アドレスは、コマンドの引数となります。

---

---

ワーク・アドレス上位 8bit の指定	PCM	MIDI
---------------------	-----	------

<書式>

"*address*

ver,0.01

<出力>

0x22 *address*

<初期値・範囲>

*address*

初期値 0

範囲 0 ~ 255

<解説>

ワークの上位アドレスを指定します。

0 以外の値を指定した場合の動作は保証しません。

アクセント	MIDI
-------	------

<書式>

'*velocity*

ver,0.01

<出力>

0x27 *velocity*

<初期値・範囲>

*velocity*

初期値 127

範囲 0 ~ 127

<解説>

次の Key のみに対して、ここで指定した数値の Velocity で KeyOn をします。



相対ボリューム	PCM	MIDI
---------	-----	------

#### <書式>

(	．．． Volume down	ver,0.01
)	．．． Volume up	ver,0.01

#### <出力>

0x28	．．． Volume down
0x29	．．． Volume up

#### <解説>

音量を相対的に変化させます。  
変化量は、'x'コマンドにて指定します。

LFO スイッチ	PCM	MIDI
----------	-----	------

#### <書式>

* <i>type</i> , <i>n</i>	ver,0.04
--------------------------	----------

#### <出力>

0x2a	<i>type</i> <i>n</i>
------	----------------------

#### <初期値・範囲>

<i>n</i>	初期値 0	範囲 0 (Disable) or 1 (Enable)
----------	-------	------------------------------

#### <解説>

ソフトウェア LFO のスイッチです。type は以下の通りです。  
(type は、ワーク参照ができません。)

0	音程 LFO
1	音量 LFO
2	パンポット LFO

エンベロープスイッチ		PCM	MIDI
<書式>			
	<i>/type , n</i>		ver,0.05
<出力>			
0x2f	<i>type    n</i>		
<初期値・範囲>			
<i>n</i>	初期値 0      範囲    0 (Disable)   or   1 (Enable)		
<解説>			
ソフトウェア・エンベロープのスイッチです。type は以下の通りです。 (type は、ワーク参照ができません。)			
0	音程 エンベロープ		
1	音量 エンベロープ		
2	パンポット エンベロープ		

ワーク演算	PCM	MIDI
-------	-----	------

# <書式>

*command, address, no*

ver,1.00

# <出力>

*command* + 0x30      *address*      *no*

# <範囲>

<i>command</i>	0 ~ 9	コマンド
<i>address</i>	0 ~ 255	アドレス
<i>no</i>	0 ~ 255	数値

# <解説>

ワーク演算を行う。

command については、以下の通り。

- |   |         |
|---|---------|
| 0 | 代入      |
| 1 | 加算      |
| 2 | 減算      |
| 3 | 論理積     |
| 4 | 論理和     |
| 5 | 排他的論理積  |
| 6 | ビットセット  |
| 7 | ビットリセット |
| 8 | コンペア    |
| 9 | ビットテスト  |

ループ	PCM	MIDI
-----	-----	------

#### <書式>

[	・・・ ループ開始位置	ver,0.03
:	・・・ 最後のループで抜ける。	ver,0.03
; <i>flag</i>	・・・ 条件一致で抜ける。	ver,1.00
] <i>count</i>	・・・ ループ終了位置	ver,0.03

#### <出力>

0x5B    *count*(byte)  
0x3A    *address*(word)  
0x3B    *flag*(byte)            ,*address*(word)  
0x5D    *address*(word)

#### <範囲>

*count*    1 ~ 255  
*flag*    bit 0:Carry                      bit 1:Zero                      (ワーク演算結果)

#### <解説>

'[]'で囲った MML を指定数値分繰り返します。  
':'がある場合は、最後のループ時にループを終了します。  
';'は、前回のループ演算時の結果に応じてループを抜けます。  
ループのネストは8回までです。  
ループ回数は、ワーク参照ができません。

相対オクターブ	PCM	MIDI
---------	-----	------

#### <書式>

<	Octave Down	ver,0.01
>	Octave Up	ver,0.01

#### <出力>

0x3C	Octave Down
0x3D	Octave Up

#### <解説>

相対的にオクターブを増減させます。

音色・プログラムの指定。	PCM	MIDI
--------------	-----	------

#### <書式>

@ <i>voice</i>	KeyOn 時の音色	ver,0.01
@F <i>voice</i>	KeyOn 時の音色	ver,1.05
@S <i>voice , decay</i>	KeyOn から decay 後の音色	ver,1.05
@R <i>voice</i>	KeyOff 後の音色	ver,1.05

#### <出力>

0x40	<i>voice (bit8 = 0)</i>
0x40	<i>switch (bit8 = 1) , voice ( , decay (switch="81h"の時) )</i>

#### <初期値・範囲>

<i>voice</i> (byte) 範囲	PCM: 0 ~ 15	@S , @R の初期値	255
	MIDI: 0 ~ 127		
<i>decay</i> (word) 範囲	0 ~ 65535		

#### <解説>

音色(PCM)及び、プログラム(MIDI)番号を指定します。

‘@S’,‘@R’コマンドは、128 以降の数値を指定すると向こうになります。

ベンド	MIDI
-----	------

<書式>

B *bend* (, *range*)

Ver,0.08

<出力>

0x42 *bend* <sup>1</sup> (*range*)

1 'range'を記述する場合は、bit15 は'1'になる。

<初期値・範囲>

*bend* (word) 初期値 8192 範囲 0(低) ~ 8192(中) ~ 16383(高)

*range* (byte) 初期値 2 範囲 1 ~ 127

<解説>

ベンド及び、ベンド・レンジを指定します。

音源、トラックの指定	PCM	MIDI
------------	-----	------

<書式>

C *track*

ver,0.01

<出力>

0x43 *track*

<初期値・範囲>

*track* 初期値 そのトラック番号 範囲 MIDI 0 ~ 15

PCM 128 ~ 131

<解説>

音源と、そのチャンネル又はトラック番号を指定します。

ディチューン（音程）	PCM	MIDI
------------	-----	------

<書式>

D *detune*

ver,0.01

<出力>

0x44 *detune*

<初期値・範囲>

*detune*(word) 初期値 0 範囲 -32768 ~ 32767

<解説>

音程のディチューン(ズレの度合い)を指定します。

エンベロープ	PCM	MIDI
--------	-----	------

<書式>

E *type , no*

ver,0.05

<出力>

0x45 *type no*

<初期値・範囲>

*no* 初期値 0 範囲 0 ~ 23

<解説>

エンベロープ形状を指定します。ヘッダー'E','Ex'命令にて指定したエンベロープが使用されます。"ty"は、以下の通りです。

(type は、ワーク参照ができません。)

- 0 音程
- 1 音量
- 2 パンポット

ボリューム	MIDI
-------	------

<書式>

F *volume*

Ver,0.08

<出力>

0x46 *volume*

<初期値・範囲>

*volume* 初期値 100

範囲 0 ~ 127

<解説>

音量を指定します。

チャンネル・プレッシャー	MIDI
--------------	------

<書式>

G *press*

Ver,0.08

<出力>

0x47 *press*

<初期値・範囲>

*press* 初期値 MIDIのマニュアル参照

範囲 0 ~ 127

<解説>

チャンネル・プレッシャーを指定します。



プログラム・バンクセレクト	MIDI
---------------	------

<書式>

H *lsb* , *msb*

Ver,0.08

<出力>

0x48 *lsb* *msb*

<初期値・範囲>

*lsb* 初期値 MIDI のマニュアル参照

範囲 0 ~ 127

*msb* 初期値 MIDI のマニュアル参照

範囲 0 ~ 127

<解説>

プログラムのバンクセレクトをします。

キーオン・ディレイ	PCM	MIDI
-----------	-----	------

<書式>

K *delay*

ver,0.01

<出力>

0x4B *delay*

<初期値・範囲>

*delay* (word) 初期値 0

範囲 0 ~ 65535(単位:Step)

<解説>

キーオンからの遅延時間を指定します。

注意：ソフトウェア LFO,エンベロープに対してのみ有効となります。

無限ループ位置指定。	PCM	MIDI
------------	-----	------

#### <書式>

L

ver,0.01

#### <出力>

- ・このコマンドのあったアドレスを記憶する。
- ・MML ファイルの[EOF]を検出したらコマンドを書き込み、コンパイルを終了する。

0x4C address('L'コマンドの記述がない場合は、0x0000 を出力する。)

#### <解説>

無限ループ時の戻り位置を指定します。

モジュレーション	MIDI
----------	------

#### <書式>

M *decay, mod*

Ver,0.0a

#### <出力>

0x4D *decay mod*

#### <初期値・範囲>

<i>decay</i>	初期値 0	範囲	0 ~ 65535
<i>mod</i>	初期値 MIDI のマニュアル参照	範囲	0 ~ 127

#### <解説>

モジュレーションを指定します。

---

ノン・レジスタード・パラメータ	MIDI
-----------------	------

---

<書式>

N *lsb* , *msb* , *data*

Ver,0.08

<出力>

0x4E *lsb* *msb* *data*

<初期値・範囲>

*lsb*

範囲 0 ~ 127

*msb*

範囲 0 ~ 127

*data*

初期値 MIDI のマニュアル参照

範囲 0 ~ 127

<解説>

ノン・レジスタード・パラメータを設定します。

---

ソヌーテート	MIDI
--------	------

---

<書式>

O *s*

Ver,0.08

<出力>

0x4F *s*

<初期値>

*s*

初期値 0

範囲 0 (Disable) or 1 (Enable)

<解説>

ソヌーテート・ペダルを指定する。

---

モードスイッチ・ダンパースイッチ	PCM	MIDI
------------------	-----	------

<書式>

P *switch*

ver,0.03

<出力>

0x50 *switch*

<初期値・範囲>

*switch*      初期値 0      範囲 0 (Disable) or 1 (Enable)

<解説>

モード及びダンパー・スイッチを設定します。

PCM	ch0	0 固定の事	PCM	ch1	0 固定の事
PCM	ch2	Sweep Mode	PCM	ch3	Noise Mode
MIDI		Dumper Switch			

ゲートタイム 8 分率	PCM	MIDI
-------------	-----	------

<書式>

Q *gate time*

ver,0.01

<出力>

0x51 *gate time*

<初期値・範囲>

*gate time*      初期値 8      範囲 1 ~ 8

<解説>

音のなる長さの割合を指定します。'4'などを指定することでスタッカートの効果が得られます。内部パラメータは、'U'コマンドと共通です。('U'コマンドと排他使用となります。)

ブレス・コントローラ	MIDI
------------	------

<書式>

R *bless*

ver,0.08

<出力>

0x52 *bless*

<初期値・範囲>

*bless*

初期値 MIDI のマニュアル参照

範囲 0 ~ 127

<解説>

ブレスコントローラを指定する。

ソフト・ペダル	MIDI
---------	------

<書式>

S *soft*

ver,0.08

<出力>

0x53 *soft*

<初期値・範囲>

*soft*

初期値 0

範囲 0 (Disable) or 1 (Enable)

<解説>

ソフト・ペダルを指定する。

フット・コントローラ	MIDI
------------	------

<書式>

T *foot*

Ver,0.08

<出力>

0x54 *foot*

<初期値・範囲>

*foot*

初期値 MIDI のマニュアル参照

範囲 0 ~ 127

<解説>

フットコントローラを設定する。

ゲートタイム 100 分率	PCM	MIDI
---------------	-----	------

<書式>

U *gate time*

ver,0.01

<出力>

0x55 *gate time*

<初期値・範囲>

*gate time*

初期値 100

範囲 1 ~ 100

<解説>

音のなる長さの割合を指定します。'50'などを指定することでスタッカートの効果が得られます。内部パラメータは、'U'コマンドと共通です。('U'コマンドと排他使用となります。)

ディチューン（音量）	PCM	MIDI
------------	-----	------

<書式>

V *detune*

ver,0.01

<出力>

0x56 *detune*

<初期値・範囲>

*detune*(word) 初期値 0

範囲 -32768 ~ 32767

<解説>

音量のディチューン(ズレの度合い)を指定します。

エクスクルーシブ	MIDI
----------	------

#### <書式>

X{*data(1),data(2),...,data(n)*} ver,0.05

X{*data(1),data(2),..., (data(3),...,data(4)),...,data(n)*} ver,0.05

#### <出力>

0x58    *0xF0*    *data(1) data(2)....*            *data(n) 0xF7*

0x58    *0xF0*    *data(1) data(2)...*            *data(3)... data(4) CRC data(n) 0xF7*

#### <範囲>

*data()* 0 ~ 127

#### <解説>

MIDI にエクスクルーシブを送信します。数列は、'{'で囲んでください。  
数列を'()'で囲んだ場合は、自動的にコンパイラ側で'CRC'を演算して出力します。  
この命令は、ワーク参照ができません。

バイナリーデータ・送信	MIDI
-------------	------

#### <書式>

Z{*data(1),data(2),...,data(co)*} ver,0.05

#### <出力>

0x5A    *co*            *data(1) data(2) ...*            *data(co)*

#### <範囲>

*data()* 0 ~ 127

#### <解説>

MIDI にバイナリ・データを送信します。数列は、'{'で囲んでください。  
この命令は、ワーク参照ができません。



移調		PCM	MIDI
<書式>			
_ <i>trance</i>		ver,0.01	
<出力>			
0x5F <i>trance</i>			
<初期値・範囲>			
<i>trance</i> 初期値	0	範囲	-128 ~ 127
<解説>			
移調度を指定します。			

音符・休符	PCM	MIDI
-------	-----	------

## <書式>

音階コード 半音 音長 付点 タイ ver,0.01

### 音階コードについて

c	ド	g	ソ
d	レ	a	ラ
e	ミ	b	シ
f	ファ	r	休符

### 半音について

#	シャープ	-	フラット
+	シャープ	=	ナチュラル

### 音長

%n      ステップ指定(1 ~ 65535)  
n          n 分音符指定(1 ~ 65535)  
0 の場合、次の音階と和音を演奏する。(MIDI のみ)

### 付点

.          付点(何個でも可)

### タイ

&          タイ(KeyOff 処理をしない。)

## <出力>

[com](#)    [\(lengs\)](#)      詳細は、Developer's Manual 1 を参照。

## <初期値・範囲>

- ・音長 'l' コマンドで指定した値。
- ・半音 'h' コマンドで指定した音階に対して自動的に半音のフラグが追加される。

## <解説>

音符・休符を指定します。

キーオン・ヴェロシティ	MIDI
-------------	------

<書式>

k *velocity* ver,0.01

<出力>

0x6B *velocity*

<初期値・範囲>

*velocity* 初期値 0 範囲 0(小) ~ 127(大)

<解説>

ヴェロシティを設定します。

音長省略時の音長	PCM	MIDI
----------	-----	------

<書式>

l *length* ver,0.01  
l% *step*

<出力>

0x6C *( time base \* 4 ) / length*  
0x6C *step*

<初期値・範囲>

*length* 初期値 4 ( time base = 48 の時 ) 範囲 1 ~ 65535  
*step* 初期値 48 範囲 1 ~ 65535

<解説>

音符・休符で音長を省略したときの音長を指定します。  
この命令は、ワーク参照ができません。

LFO	PCM	MIDI
-----	-----	------

# <書式>

m *type , decay , speed , level , count*

ver,0.04

# <出力>

0x6D *type decay speed level count*

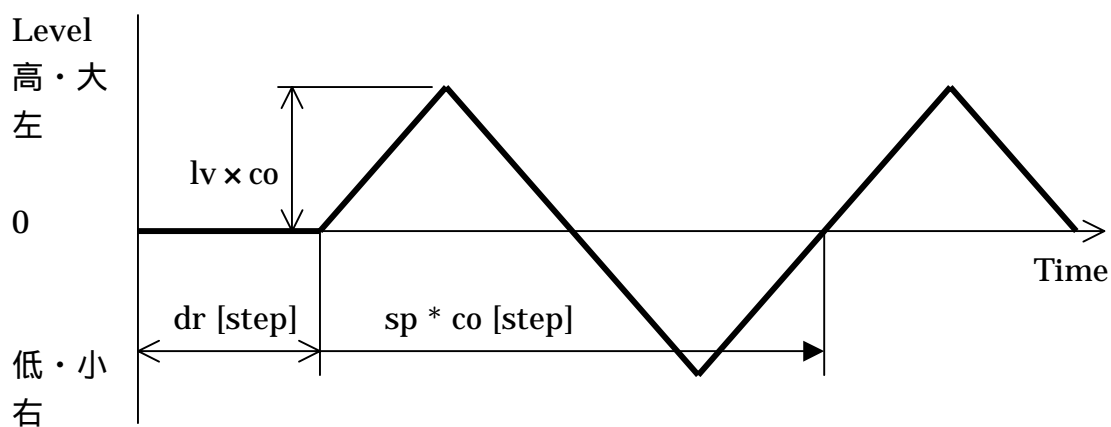
# <範囲>

*type*            0 : 音程            1 : 音程            2 : パンポット  
*decay*           0 ~ 255    ( 単位:Step )  
*speed*           1 ~ 255    ( 単位:Step )  
*level* (word)    -32768 ~ 32767  
*count*           1(One Shot LFO),2 ~ 255(Triangle LFO)

# <解説>

ソフトウェア LFO の形状を指定します。  
 (type は、ワーク参照ができません。)

# <図 LFO 形状>



ノイズ	PCM
-----	-----

<書式>

n *noise*

ver,0.01

<出力>

0x6E *noise*

<初期値・範囲>

*noise* 初期値 0

範囲 0 ~ 7

<解説>

PCM ch4 のノイズ・モードを指定する。

オクターブ	PCM	MIDI
-------	-----	------

<書式>

o *octave*

ver,0.01

<出力>

0x6F *octave*

<初期値・範囲>

*octave* 初期値 4

範囲 -2(低) ~ 4(中央) ~ 10(高)

<解説>

オクターブを指定します。

パンポット	PCM	MIDI
-------	-----	------

<書式>

p *pan* (, *mode*)

ver,0.01

p%*pan*%(, *mode*)

mode=1 使用時は、ver,1.07

<出力>

0x70 *pan* + *mode*\*128

0x70 *pan*%\*4 + *mode*\*128 (*pan*%が32の場合は、計算値から1を引く)

<初期値・範囲>

*pan* 初期値 64

範囲 0(左) ~ 64(中央) ~ 12(右)

*mode* 初期値 0

範囲 0:一次関数変化, 1:対数変化

<解説>

パンポットを指定します。

ゲートタイム	PCM	MIDI
--------	-----	------

<書式>

q *gate time*

ver,0.01

<出力>

0x71 *gate time*

<初期値・範囲>

*gate time* (word)

初期値 0

範囲 0 ~ 65535 (単位:Step)

<解説>

キー・オフを指定した数値分早めます。

スウィープ	PCM
-------	-----

<書式>

s *level, rate*

ver,0.05

<出力>

0x73 *level rate*

<初期値・範囲>

*level* 範囲 0(小) ~ 255(大)

*rate* 範囲 0(早) ~ 31(遅)

<解説>

スウィープモードを設定します。

テンポ	PCM	MIDI
-----	-----	------

<書式>

t *tempo*

ver,0.01

t% *tempo%*

ver,0.01

<出力>

0x74 *720000 / (tempo × timebase)*

0x74 *tempo%*

<初期値・範囲>

*tempo* (word) 初期値 120 ( timebase = 48 の時 ) 範囲 1 ~ 65535

<解説>

テンポを指定します。

ゲートタイム	PCM	MIDI
--------	-----	------

<書式>

u *gate time*

ver,0.01

<出力>

0x75 *gate time*

<初期値・範囲>

*gate time* (word)

初期値 0

範囲

0:無効

1 ~ 65535 ( 単位 : Step )

<解説>

キーオンから音を鳴らす時間を指定します。

エクスプレッション	PCM	MIDI
-----------	-----	------

<書式>

v *expression*

ver,0.01

v% *expression%*

ver,0.01

<出力>

0x76 *expression*

0x76 *expression% \*8*

<初期値・範囲>

*expression*

初期値 0

範囲

0(小) ~ 127(大)

<解説>

音量を指定します。MIDI の場合は、エクスプレッションとして扱われます。



相対音量の増減値	PCM	MIDI
----------	-----	------

<書式>

x *dev*

ver,0.05

<出力>

0x78 *dev*

<初期値・範囲>

*dev* 初期値 1 範囲 0(小) ~ 127(大)

<解説>

相対音量コマンド'(',')'の差分値を指定します。

コントロールチェンジ	MIDI
------------	------

<書式>

y *reg, data*

ver,0.05

<出力>

0x79 *reg data*

<初期値・範囲>

*reg* 範囲 0 ~ 127

*data* 初期値 MIDIのマニュアル参照 範囲 0 ~ 127

<解説>

コントロールチェンジを送信します。

半音指定	PCM	MIDI
------	-----	------

# <書式>

{ 半音 音階コード}

Ver,0.08

半音について

#	シャープ	-	フラット
+	シャープ	=	ナチュラル

音階コードについて

c	ド	g	ソ
d	レ	a	ラ
e	ミ	b	シ
f	ファ	r	休符

# <出力>

0x7B *data*      詳細は、Developer's Manual を参照。

# <初期値・範囲>

すべての音符は、半音指定はされていません。

# <解説>

指定した音符に対して、ナチュラル'='が指定されていない限り自動的に半音が追加されます。八長調でない曲を打ち込み時に、役に立ちます。  
この命令は、ワーク参照ができません。

#### 4 - 2 - 3 . Wonder Witch への転送・演奏方法

Wonder Witch の機材を使用せずに Wonder Witch に転送する方法は、以下の機材が別途必要になります。

- ・ Wonder Swan 本体(又は、上位互換)
- ・ Wonder Witch のカートリッジ
- ・ Wonder Wave
- ・ 赤外線シリアル通信機能を備えた、MS-DOS が使用可能なコンピュータ
- ・ MS-DOS Ver,3.30c ~ Windows XP Professional (NT Server は不可。)  
64bit 版の Windows では使用できません。
- ・ X-modem に対応したシリアル通信ソフト。(例、KTX , WTERM 等)

Wonder Witch のカートリッジには、Freya OS と呼ばれるカーネルが入っており、FTP と同等のコマンド (一部使用できないものもあります) が使用可能です。Wonder Witch のカートリッジにファイルを転送するには、

Wonder Swan と、Wonder Wave を接続して、電源を投入する。

(Wonder Witch の通信速度は 9600bps モードに設定する。)

通信ソフトを起動し、以下のように設定する。

- |           |          |
|-----------|----------|
| ・ 通信速度    | 9600 bps |
| ・ データビット  | 8bit     |
| ・ ストップビット | 1bit     |
| ・ パリティ    | 無し       |
| ・ フロー制御   | 無し       |

“send 「送りたいファイル名」”と打ち込み、[ENTER]を押す。

Wonder Swan の”A”ボタンを押す。

X-modem でファイルを送信する。

曲データを演奏するには、

“WTSSV.FX”と、“WTD.FX”を Wonder Witch に転送する。(4 - 1 項参照)

Wonder Swan の”B”ボタンを押し、ランチャー・モードにする。

“wtdsv”を選択して”A”ボタンを押す。

(WTD 標準のプレイヤー & セレクターが起動します。)

演奏したいファイルを”X”キーで選択して”A”ボタンを押します。

プログラム開発を行う際には、必ず Wonder Witch の正式なライセンスを手して下さい。

## 4 - 3 . ROM イメージ メーカー

### 4 - 3 - 1 . WonderSwan エミュレータについて

作曲及び曲の打ち込みをする際に、WonderWitch へ転送しながら曲をチェックするのは非常に大変です。そこで、このツールでは WonderSwan エミュレータ用の ROM イメージファイル(128kByte)を作成し、WonderSwan エミュレータ上で曲データの演奏を可能にするものです。

このツールにて作成される ROM イメージは、イメージ内に WTD 本体及び、曲データを含みます。尚、WonderWitch の OS, BIOS は ROM イメージ中に含まないので、インターネットでの配付は問題ありません。

尚、"ROM\_IMG.BIN"ファイルは WTD 本体とプレイヤー・ビューアーを含む WonderSwan 用の独自 BIOS です。ROM イメージ作成時、"WSMAKE.COM"は同じフォルダに置かれている"ROM\_IMG.BIN"ファイルを ROM イメージファイルに埋め込むので、"WSMAKE.COM"と同じフォルダに置いて下さい。

動作確認しているエミュレータは以下のとおりです。

Oswan

作者      toyo 氏

URL      <http://php.nm.land.to/>

### 4 - 3 - 2 . コマンドライン書式

MS-DOS のコマンドプロンプト上で、以下のように記述します。

WSMAKE [filename(.wtd)]

すると、指定した WTD 形式の曲データファイルを、WonderSwan エミュレータ用の ROM イメージファイルに変換し、WonderSwan エミュレータ上で演奏が可能になります。

## 5 . 参考文献

Wonder Swan 説明書		発行: (株) BANDAI
Wonder Witch 説明書		発行: (有) QUTE
LSI-C86 説明書		発行: (株) LSI Japan
Roland SC88 ~ 8850 説明書		発行: (株) Roland
YAMAHA Mu80,128 説明書		発行: (株) YAMAHA
JIS X 6054-1		発行: (財) 日本工業規格協会
JIS X 6054-2		発行: (財) 日本工業規格協会
MIDI 規格 ver. 1.0		発行: (財) 電子音楽事業団
Oh! X 88 年 8 月号		発行: (株) Soft Bank
Oh! X 88 年 12 月号		発行: (株) Soft Bank
試験にでる X1	著: 祝一平	発行: (株) Soft Bank
PC9800-86 説明書		発行: (株) NEC
PC9800 Series Technical Data Book		発行: (株) ASC
Z80 の使い方	著: 横田英一	発行: (株) オーム社
続 Z80 の使い方	著: 横田英一	発行: (株) オーム社
初めての MASM	著: 薄池輝尚	発行: (株) ASCII
初めての 4 8 6	著: 薄池輝尚	発行: (株) ASCII
C 言語入門	著: 大角盛広	発行: 西東社
C 言語辞典	著: 平林雅栄	発行: 技術評論社
MML2MID 説明書	著: 門田暁人, 藤井秀樹, 黒田久泰, 新出尚之	
PMD 説明書	著: 梶原正裕	発行: (同) 緑水ソフトウェア

誠に申し訳ありませんが、敬称は省略させていただきました。

*m e m o*

*m e m o*

*m e m o*



## 作者略歴

### 渡部 篤史

昭和 54      2 月 4 日生  
平成 11      東京都立工業高等専門学校 機械工学科卒業  
                半導体メーカーに就職 半導体技術本部  
平成 12      同人チーム Digitalis に参加

## 主な作品

「蛸 Project」( 都立高専エレクトロニクス研究部 )  
「Fm Sound Player (FSP)」( 個人 )

# Wonder Swan Total Sound Driver User's Manual

---

2001 年 5 月 4 日    Version 0.01  
2007 年 7 月 19 日    Version 1.0a

発行    Digitalis  
発行者 大串 公德  
作者    渡部 篤史  
          渡辺 伊織

印刷    グランド印刷(株) TUM 事業部

Standard Edition      無料  
Pro.Edition 製本無し 定価 ( 本体    ¥700 / 即売会特価    ¥500 )  
Professional Edition 定価 ( 本体    ¥1,600 / 即売会特価    ¥1,200 )  
Developer's Edition 定価 ( 本体    ¥20,000 )  
製本マニュアル      定価 ( 本体    ¥1,200 / 即売会特価    ¥1,000 )

---

万一、落丁・乱丁の際にはお取り替えいたします。

本製品の内容を無断で複製複写することは、著作者及び発行者の権利を侵害すること  
があります。その場合には、あらかじめ当サークル宛に承諾をお求めください。