

第3章

拡張BASIC編

1. MSX-Audioと拡張BASIC
2. 拡張BASICを起動しよう
3. 拡張BASICの命令
 - 拡張BASICの機能を設定する命令
 - FM音源の操作に関する命令
 - PCM音源の操作に関する命令
 - ミュージックキーボード関係の命令
 - MK記録に関する命令
4. 命令索引
5. ミュージック・マクロ・ランゲージ (MML)
6. FM音源用音色データ一覧表
7. ROM音声ファイル一覧表

1. MSX-Audioと拡張BASIC

●MSX-Audioとは

このMSXオーディオユニットは、音源ICとしてMSX-Audioを使っています。

MSX-Audioは9チャンネルのFM音源と、1チャンネルのADPCM音源を内蔵しています。

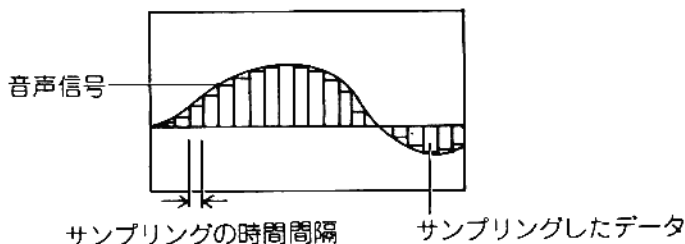
FM音源：

FMとは、ラジオのFM放送などでおなじみの周波数変調のことです。FM音源は周波数変調によって生じる高調波を楽音の合成に利用します。自然楽器の音色から電子音まで、幅広い音の発生が可能です。

ADPCM音源：

ADPCMとは、「適応差分PCM」のことです。PCMは「パルス符号変調」とも呼ばれます。

音源の方式としてのPCMは、入力されたアナログの音声信号を一定の時間間隔で分析（これをサンプリングと呼びます。）し、結果として得られたデジタルのデータをメモリに記憶させます。再生を行うときには、このデータを基にして元の音声信号を復元します。サンプリングを行うとき、また元の音声信号を復元するときの時間間隔を特にサンプリング周波数と呼びます。



●拡張BASIC

MSXの音楽機能は、音源としてPSGを使うことだけを考慮して作られたため、MSX-BASICにも、PSGを操作する命令以外には音源を操作する命令は含まれていません。

そこで、MSX-Audioを操作するために必要な命令をBASICに追加したものが、「拡張BASIC」です。

拡張BASICを理解するには、標準のMSX-BASICをある程度理解しておくことが必要です。またMSX-BASICについての説明書を読んでいないかたは、先にそちらをお読みください。

お手持ちのパソコンにMSX-BASICについての説明書が付属していない場合は、下記の説明書を用意していますので本機をお求めになった販売店にご注文ください。

「MSX2 BASIC使用説明書」……品番DFQF2033Z（別売品）

●ご使用になるパソコンについて

この拡張BASICは、すべてのMSXパソコンおよびMSX2パソコンでご使用になれるように作られていますが、一部の命令については、MSXパソコンとMSX2パソコンでは指定できるパラメータの種類に違いのある場合があります。

2. 拡張BASICを起動しよう

下のようにBASICの初期画面が表示されます。

```
MSX BASIC version x.0  
Copyright 198x by Microsoft  
xxxxx Bytes free  
Ok
```

- ⑧ と入力し、 キーを押します。 は、 キーをあらわします。）

画面に再び、

Ok



と表示されたら、拡張BASICが起動しています。

もし、下のようなメッセージが画面に表示されたら、

```
Syntax error
```

Ok



一度パソコンの電源スイッチを切ってから、FS-CA1がパソコンのスロットに確実に差し込まれているか確かめてください。

2. 拡張BASICを起動しよう

■電源スイッチの切りかた

フロッピーディスクドライブを接続している場合は、

- ① フロッピーディスクドライブのIN USEランプ（アクセスランプ）が点灯していないことを確かめます。
点灯しているときは、ランプが消えるまで待ってください。
- ② フロッピーディスクをフロッピーディスクドライブから取り出します。

上の操作を行ったかた、およびフロッピーディスクドライブを接続していないかたは、次の順序で電源スイッチを切ってください。

- ① パソコンの電源スイッチを切ります。
- ② 他の周辺機器の電源スイッチを切ります。

ご注意

フロッピーディスクドライブのIN USEランプ（アクセスランプ）が点灯しているときにパソコンやフロッピーディスクドライブの電源スイッチを切ったりフロッピーディスクを取り出したりすると、フロッピーディスクの内容が壊れることがあります。

3. 拡張BASICの命令

■拡張BASICの命令の種類

拡張BASICで追加された命令は、大きく分けて次の5種類に分かれます。本書では、この順序で各命令を説明しています。

179 ページに命令の索引を載せていますので、必要に応じてご利用ください。

1. 拡張BASICの機能を設定する命令

AUDIO	BGM	STOPM
PLAY (命令)	PLAY (関数)	SYNTHE

2. FM音源の操作についての命令

VOICE	PITCH	TRANPOSE
VOICE COPY		TEMPER

3. PCM音源の操作についての命令

SET PCM	COPY PCM	PLAY PCM
PCM FREQ	PCM VOL	REC PCM
SAVE PCM	LOAD PCM	CONVA
CONVP		

4. ミュージックキーボードでの演奏についての命令

MK VOICE	MK VOL	MK PCM
MK VEL	INMK	KEY ON/OFF
MK TEMPO		

5. 演奏の記録についての命令

RECMOD	REC MK	PLAY MK
CONT MK	APPEND MK	MK STAT

■この章の読みかた

各命令の説明では、命令の基本的な意味、書式、文例を掲げ、その後、解説を載せています。また、多くの命令にプログラム例とその解説を付け加えています。各命令の書式には、その命令に続けて書くことが必要な（または書くことのできる）数値や記号を、下の例のような形式で示しています。

(例)

```
CALL AUDIO [ (<モード> [, <インスツルメントへのチャンネル数> [, <PLAY文第1文字列へのチャンネル数> [, <PLAY文第2文字列へのチャンネル数> [, …… [, <PLAY文第9文字列へのチャンネル数>] ] …… ] ) ]
```

これらの数値や記号を「パラメータ」と呼びます。

[,] , < , > は入力しないでください。[] 内のパラメータは省略可能です。省略をするときは、[と] が正しく対応するように省略してください。

< > は、そのパラメータの意味を示します。

- ♪ 拡張BASICの命令は、PLAY文（ステートメント）を除いては、すべて命令の前に「CALL」（コール）を付けて使用する必要があります。
- ♪ 「CALL」の代わりに「_」（アンダースコア）を使用することができます。

(例)

```
_AUDIO (0, 9)
```


解説：「MSX-Audio」を初期化する（動作を開始するときそうならない状態にすること）とともにFM音源の9個のチャンネルをどのように使用するかを指定します。AUDIO文により初期化を行うまでは、拡張BASICにより追加された他の命令は使えません。

<モード>は0～3までの数字で指定し、下の表のようにMSX-Audioの動作モードを設定します。

○：可能 ×：不可能

モード：	0	1	2	3
リズム音の使用	×	○	×	○
PCM音源をPLAY文で使用	×	×	○	○

ご注意 モードの指定に4以上の数字を使用しないでください。

リズム音を使用するときはチャンネル7、8、9を使用するので、楽音に使えるのは残りの6チャンネルになります。

FM音源のうち、ミュージックキーボードに割り当てられて使用されるものを、「インストゥルメント」と呼びます。

インストゥルメントにチャンネルを割り当ててAUDIO文を実行すると、FM音源の音色によるミュージックキーボードの演奏が可能になります。

ご注意 BASICで使用している場合には、パソコンのキーボードで演奏することはできません。

<インストゥルメントへのチャンネル数>とPLAY文で使用するチャンネル数の合計は、リズム音を使用するとき（モード1または3）には6以下、リズム音を使わないときには9以下に設定してください。

チャンネルの使用割り当ては、PLAY文ではチャンネル番号の小さい方（1、2、3…）から、インストゥルメントではチャンネル番号の大きい方（9、8、7…）から割り当てます。

パラメータを1つ以上指定したときは、他のパラメータの省略時の値は0となります。

PLAY文へのチャンネル数を0に設定したり、途中のパラメータを省略することはできません。

パラメータなしで、

CALL AUDIO

上のようにして使ったときは、

CALL AUDIO (1, 3, 1, 1, 1)

と同じになります。つまり、

- ・ FM音源のチャンネル1をPLAY文の最初の文字列に割り当てる。
- ・ FM音源のチャンネル2をPLAY文の2番目の文字列に割り当てる。
- ・ FM音源のチャンネル3をPLAY文の3番目の文字列に割り当てる。
- ・ FM音源のチャンネル4～6をインストゥルメントに割り当てる。
- ・ FM音源のチャンネル7～9をリズム音に使用する。
- ・ PLAY文の4番目の文字列でリズム音を操作する。
- ・ PLAY文の5番目から7番目までの文字列はPSG音源の操作に割り当てる。
- ・ PCM音源は、PLAY文では操作しない。

ご注意

PLAY文の1つの文字列に2つ以上のチャンネルを割り当てて使用すると、相互の干渉により、音が小さくなることがありますので、通常はPLAY文へのチャンネル割り当ては、1つの文字列に対して1チャンネルだけにしてください。

BGM (ピージーエム)

[ステートメント]

機能： バックグラウンド処理を行うかどうかを指定します。

書式： CALL BGM (n)

文例： CALL BGM (0) バックグラウンド処理を行わない。

CALL BGM (1) バックグラウンド処理を行う。

解説： nは0または1の値をとり、値によって、

0 : バックグラウンド処理を行いません。

1 : バックグラウンド処理を行います。

ここで言う「バックグラウンド処理」とは、MSXオーディオユニットに拡張BASICの命令を実行させるにあたって、1つの命令の実行が完全に終わってから次の命令を実行させるのではなく、前の命令の実行がまだ終わっていなくても、次の命令の実行が可能なものなら、次々に命令を実行させて行くことです。これによって、一方では音楽を演奏しながらもう一方で次のキー入力を受け入れるプログラムなどが、音楽の演奏を間延びさせたりしないで実行できます。

次にあげる機能は、バックグラウンド処理を行うことができます。

- ・PLAY文による演奏
- ・外部メモリを使用するADPCMの録音／再生
- ・配列変数を記録領域に使用しないMK記録の記録／再生

AUDIO文によって初期化された状態では、バックグラウンド処理が行われますが、nを0にしてBGM命令を実行させることによって、前の命令の実行が終わるまで次の命令の実行を待たせることができます。

参照：AUDIO (94ページ)、STOPM (99ページ)

プログラム例：

```

10 /
20 /
30 /      BGM TEST
40 /
50 /
60 CALL AUDIO
70 CLS
80 LOCATE 0,10
90 PRINT "バックグラウンド"ラウンド"しより"を..."
100 INPUT "する=1,しない=0";K
110 CALL BGM (K)
120 FOR N=1 TO 3
130     CALL VOICE (@N)
140     PLAY#2,"V1504CEG050"
150 NEXT N
160 GOTO 70

```

プログラム例解説：

RUNさせると、バックグラウンド処理をするかどうかの指示を求めますので、最初は **0** キーを押してから **RETURN** キーを押してください。バックグラウンド処理をしないで、3種類の楽器の音色を切り換えながら“ドミソド”の音を演奏します。

一連の動作をしてから、再び指示を求めますので、今度は **1** キーを押してから **RETURN** キーを押してください。同じ動作をバックグラウンド処理をしながら行いますが先ほどとは違い、演奏が始まった直後に指示を求めてきます。また音色も、演奏が始まると同時に、一瞬のうちに3番目の音色に切り換わってしまいます。

このようにプログラムの目的によっては、バックグラウンド処理をしないほうがよい場合もあります。

プログラムを止めるには、**CTRL** + **STOP** を入力してください。

STOPM

(ストップエム)

[ステートメント]

機能： バックグラウンド処理をしている途中のPLAY文の演奏、ADPCM録音／再生、MK記録／再生を停止します。

書式： CALL STOPM [(変数名)]

文例： CALL STOPM

CALL STOPM (A)

解説： バックグラウンド処理をしている途中のPLAY文による音楽の演奏、外部メモリを使用するADPCM録音の録音／再生、MK記録の記録／再生を停止します。

パラメータとして変数名を与えると、MK記録／再生の中止されたアドレスの次のアドレス (CONT MK文により再開されるアドレス) が変数の値となって返されます。

参照： AUDIO (94 ページ), BGM (97 ページ), REC PCM (135 ページ)
PLAY PCM (128 ページ), REC MK (165 ページ)
PLAY MK (128 ページ)

プログラム例：

```

10  '
20  '
30  '          STOPM TEST
40  '
50  '
60  CALL AUDIO
70  CLS
80  LOCATE 5,12
90  PRINT "STOPMを つかう=0,つかわない=1"
100 PLAY#2,"V1204CDEFGAB05C"
110 PLAY#2,"05C04BAGFEDC"

```

```
120 K#=INKEY#
130 IF K#="0" THEN 170
140 IF K#="1" THEN 180
150 GOTO 100
160 '
170 CALL STOPM
180 END
```

プログラム例解説：

RUNさせて演奏を開始させてから、**0** キーを押したときと **1** キーを押したときの演奏の止まりかたの違いを確かめてください。

0 キーを押したときは、プログラムの停止を示す「Ok」の表示が現れると同時に演奏も停止します。

1 キーを押したときは、「Ok」の表示が現れても、演奏が止まるまでには少し時間がかかります。

PLAY

(プレイ)

[ステートメント]

機能： 音楽をミュージックマクロランゲージ (MML) にしたがって演奏する。

書式： PLAY [#<モード>,] <文字列1> [, <文字列2> [, <文字列3> ··· [, <文字列1 3>] ·····]]

文例： PLAY#2, "CD", "EF", "GA"

解説： PLAY文は音楽を演奏する命令で、FM音源 (9)、PCM音源 (1)、従来のPSG音源 (3) の最大13音まで同時発音が可能です。

<文字列>に書かれたミュージックマクロランゲージ (MML) にしたがって演奏します。(181 ページ、ミュージック・マクロ・ランゲージ (MML) 参照)

他の拡張命令と異なりCALL文は必要ありません。

<モード>は0から3までの数字で入力し、PLAY文の音源や動作モードを次のように設定します。

0 や省略されたとき ··· PSGのみが音源になり、1行に書ける文字列の数は最大3つまでになります。従来のBASICの仕様で作成されたPLAY文がそのまま使用できます。

1 のとき ····· 将来、MSXオーディオユニットでMIDIアダプタなどが利用できるようになった場合に備えた拡張用のモードです。

2 または3 のとき ··· FM音源、リズム音、PCM音源、PSG音源を使用できます。(2のときと3のときで動作に違いはありません。)

<文字列>と音源との関係は初めから順に、次のようになります。

<FM音源用文字列1>、・・・、<FM音源用文字列n>、
<PCM音源用文字列>、<リズム音源用文字列>、<PSG音源
用文字列1>、<PSG音源用文字列2>、<PSG音源用文字
列3>

nはAUDIO文の設定によりFM音源のチャンネルを割り当てられたPLAY文の総数です。

各文字列はAUDIO文で設定した内容に対応していることが必要です。AUDIO文でリズム音やPCM音源を使用しないモードに設定した場合は、リズム音用文字列やPCM音源用文字列を“ ”（カンマ）と共に省略します。

例として、パラメータを付けずにAUDIO文を実行したときのPLAY文の文字列の並べかたをあげると次のようになります。

<FM音源用文字列1>、<FM音源用文字列2>、<FM音源用文字
列3>、<リズム音用文字列>、<PSG音源用文字列1>、<PSG
音源用文字列2>、<PSG音源用文字列3>

- ご注意**
- ・PLAY文を実行中にエラーが発生したり、想定したとおりの音が出なかったりしたときは、PLAY文を調べるだけでなく、PLAY文とAUDIO文が正しく対応しているかどうかを確認してください。
 - ・PSG音源の音は、FS-CA1の音声出力端子には出力されません。PSG音源も含めた音が必要な場合は、パソコンの音声出力端子から音声出力をとるようにしてください。

参照：AUDIO (94 ページ) , BGM (97 ページ) , VOICE (107 ページ) ,
PLAY [関数] (104 ページ) , ミュージック・マクロ・ランゲージ (MML)
(181 ページ)

【PLAY】

プログラム例：

```
10  '
20  '
30  '      PLAY TEST
40  '
50  '
60  CALL AUDIO (3,3,1,1,1)
70  CALL VOICE (@16,@1,@21)
80  CALL SET PCM (0,0,0,,5)
90  CALL COPY PCM (#3,0)
100 '
110 PLAY#2,"04L8CEER8EGGR8DFFR8ABBR8", "
", " ", "05U15L4R4CR4ER4DR4A", "BSH8R8S8R8B
SH8R8S8R8B8SH8R8S8R8B8SH8R8S8R8"
120 PLAY#2,"CEER8EGGR8DFFR8ABBR8CEER8EG
GR8DFFR8ABBR8", "U1505L2GCAFECDR4", " ", "0
6R4CR4ER4DR4AR4CR4ER4DR4A", "BSH8R8S8R8B
SH8R8S8R8B8SH8R8S8R8B8SH8R8S8R8B8SH8R8S8R8
B8SH8R8S8R8B8SH8R8S8R8B8SH8R8S8R8"
130 PLAY#2,"CEER8EGGR8DFFR8ABBR8CEER8EG
GR8DFFR8ABBR8", "GCAB06CDC1", "U1506L2GCA
B07CDC1", "05R4CR4ER4DR4AR406L2CDC1", "BS
H8R8S8R8B8SH8R8S8R8B8SH8R8S8R8B8SH8R8S8R8B
SH8R8S8R8B8SH8R8S8R8B8SH8R8S8R8"
140 END
```

プログラム例解説：

RUNさせると音楽を演奏します。

FM音源とリズム音、PCM音源を使用しています。

PLAY

(プレイ)

[関数]

機能： PLAY文による演奏中かどうかを調べます。

書式： CALL PLAY (<PLAY文の文字列番号>, <変数名>)

文例： CALL PLAY (0, A) : PRINT A

解説： PLAY文に記入された各文字列が、AUDIO文で割り当てられたチャンネルまたは音源のどれかを使って演奏中かどうかを調べ、演奏中なら-1を、そうでなければ0を、指定された変数の内容にして返します。

PLAY文の文字列番号は、0から、AUDIO文の設定により、PLAY文で使用することができる文字列(パラメータ)の数まで使えます。すなわちこの命令は、AUDIO文で指定したFM音源、PCM音源だけではなく、3チャンネルのPSG音源についても有効です。文字列番号として0が与えられた場合は、どれかの文字列が演奏中であれば-1を、どの文字列も演奏していなければ0を変数の内容にして返します。

参照：AUDIO (94 ページ), PLAY [ステートメント] (101 ページ)

プログラム例：

```

10  '
20  '
30  '      CALL PLAY TEST
40  '
50  '
60  CALL AUDIO
70  CALL VOICE (@0,@5)
80  '
90  CLS

```

【PLAY関数】

```
100 PLAY#2,"U1504C0EFGAB05C"  
110 PLAY#2,"","U1505C04BAGFEDC"  
120 FOR T=1 TO 100  
130   FOR N=1 TO 2  
140     CALL PLAY(N,S)  
150     IF S=-1 THEN GOSUB 200  
160   NEXT N  
170 NEXT T  
180 GOTO 100  
190 '  
200 LOCATE 5,12  
210 PRINT "えんぞうちゅうのチャンネルは ";N  
220 RETURN
```

プログラム例解説：

RUNさせると、音階の演奏を繰り返しながら演奏中のPLAY文の文字列番号を画面に表示します。

このプログラムは **CTRL** + **STOP** を入力するまで止まりません。

S Y N T H E

(シンセ) [ステートメント]

機能： 内蔵のシンセサイザーソフト「MSXミュージックシステム」を起動します。

書式： CALL SYNTH E

文例： CALL SYNTH E

解説： この文の実行は、AUDIO文を実行する前にしてください。
AUDIO文の実行後にこの文を実行させるとエラーになり、画面に
"illegal function call"と表示されます。

参照： AUDIO (94 ページ)

■ FM音源の操作に関する命令

VOICE

(ボイス)

[ステートメント]

機能： FM音源の各チャンネルに音色を直接に設定します。

書式： CALL VOICE ([<チャンネル1用の音色指定>], [<チャンネル2用の音色指定>],, [<チャンネル9用の音色指定>])

音色指定=@+n (nは定数または単純変数)
または、配列変数名

文例： CALL VOICE (@1, @1, @1, . . . , @7, @7, @7)

- ・ FM音源の第1から第3チャンネルに音色番号1の音色を、第7から第9チャンネルに音色番号7の音色を設定します。

CALL VOICE (@3, @4)

- ・ FM音源の第1チャンネルに音色番号3の音色を、第2チャンネルに音色番号4の音色を設定します。

解説： FM音源の9つあるチャンネルのそれぞれに音色を設定します。音色の設定のしかたには2つあります。内蔵のFM音源用音色データを使用する場合は、0～63の音色の番号を単純変数または定数により指定します。(ここでは、配列変数を除いた、数値を値とする変数を単純変数と呼びます。配列変数については、パソコンの説明書またはBASICの説明書を参照してください。)この場合には変数名または定数の前に@記号(アットマーク記号)をつけて次の配列変数名と区別します。プログラムにより音色パラメータを与えて新しい音色を設定する場合には、配列変数に音色のデータを構成する各パラメータの値を入れて、その配列変数名を指定します。

- ♪ 音色データの自作には、FM音源とMSX-Audioについての詳しい知識が必要ですので、本書では説明を省略します。

パラメータを省略したチャンネルの音色は変更されません。(文例参照)。PLAY文において、MMLを使って音色を設定することもできます。インストゥルメントに指定したチャンネルは、MK VOICE文でまとめて設定できます。146 ページのMK VOICE文を参照してください。

ご注意 使用できる音色データは内蔵ソフトの音色データとは異なります。

参照: AUDIO (94 ページ), MK VOICE (146 ページ), ミュージック・マクロ・ランゲージ (MML) (181 ページ), FM音源用音色データ一覧表 (186 ページ)

プログラム例:

```

10 /
20 /
30 /      VOICE TEST
40 /
50 /
60 CALL AUDIO
70 CALL BGM (0)
80 CLS
90  FOR N=0 TO 63
100     CALL VOICE (&N)
110     LOCATE 10,10
120     PRINT "No. ";N
130     PLAY#2,"U15L5R2"
140     PLAY#2,"O5CEG06C"
150     K#=INKEY#
160     IF K# <> "" THEN 210
170     FOR J=1 TO 300

```

[VOICE]

```
180      NEXT J
190 NEXT N
200 '
210 CALL STOPM
220 END
```

プログラム例解説：

RUNさせると、内蔵のFM音源用音色データを音色番号0番（ピアノの1）から63番（無音）まで次々に切り換えながら「ドミソド」の音を演奏します。

途中で SPACE キーを押すと、プログラムの実行を停止します。

PITCH

(ピッチ)

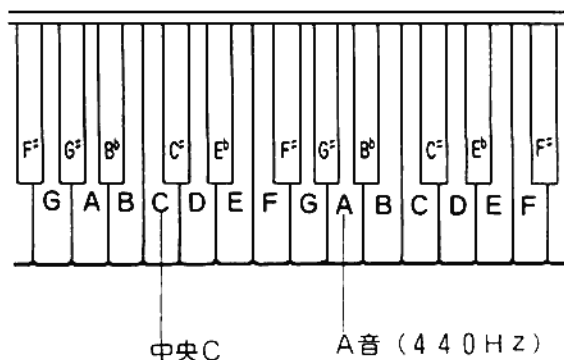
【ステートメント】

機能： FM音源の楽音の音高（ピッチ）を設定します。

書式： CALL PITCH (<ピッチ>)

文例： CALL PITCH (443)

解説： FM音源で発生する楽音の音高を定数または変数で指定します。
 <ピッチ>の範囲は410～459で単位は「Hz」（ヘルツ）です。
 中央Cのすぐ上のA音の周波数で音高をあらわします。



トランスポーズとは別に独立して設定できます。AUDIO文で初期化した直後の値（初期値）は440です。

ピッチ（またはトランスポーズ値）を変えるとリズム音や音程を持たない音を除くFM音の音の高さが変化します。PCM音源やPSG音源には作用しないので注意してください。

♪ トランスポーズについては、113ページのTRANSPOSE文の項を参照してください。

参照：AUDIO（94 ページ）、TRANSPOSE（113 ページ）

[P I T C H]

プログラム例：

```
10 /
20 /
30 /      PITCH TEST
40 /
50 /
60 CALL AUDIO
70 CALL BGM (0)
80 CALL VOICE (@5)
90 /
100 FOR N=440 TO 459 STEP 3
110   CALL PITCH (N)
120   CLS
130   LOCATE 8,10
140   PRINT "PITCH=";N
150   PLAY#2,"05L1A"
160   K#=INKEY#
170   IF K#=CHR#(&H20) THEN 320
180 NEXT N
190 /
200 FOR N=440 TO 410 STEP -3
210   CALL PITCH (N)
220   CLS
230   LOCATE 8,10
240   PRINT "PITCH=";N
250   PLAY#2,"05L1A"
260   K#=INKEY#
270   IF K#=CHR#(&H20) THEN 320
280 NEXT N
290 /
300 GOTO 100
310 /
320 CALL STOPM
330 END
```

プログラム例解説：

RUNさせると、画面に現在のピッチの数値を表示しながら、最初はピッチを3ヘルツづつ上げ、そのたびに“ラ”の音を鳴らします。ピッチが459ヘルツに達すると、今度は440から3ヘルツづつ音を下げいき、410ヘルツに達すると最初に帰って繰り返します。

途中で **SPACE** キーを押すと、その時のピッチの設定のまま、プログラムの実行を停止します。

ピッチの設定を元に戻すには、次の命令を実行させてください。

```
CALL PITCH (440)
```

TRANSPOSE (トランスポーズ) [ステートメント]

機能： ミュージックキーボードに割り当てたFM音源の楽音の音高をセント（半音の1/100）単位で移調（音階全体の音の高さを変更すること）します。

書式： CALL TRANSPOSE (<トランスポーズ値>)

文例： CALL TRANSPOSE (50)

解説： 移調を行うための命令で、「セント」と呼ばれる単位で指示します。これは、半音を100とした移調の単位で、1オクターブ上げるには、トランスポーズ値を+1200にして設定します。主に他の楽器と合奏するときのチューニングに用います。

トランスポーズ値として許される範囲は、+1279.9から-1279.9までですが、実際にはFM音源の音色によってトランスポーズの有効な範囲には制限があります。音高の精度は±2セント程度です。

トランスポーズはピッチとは別に独立して設定できます。AUDIO文による初期化した直後の値（初期値）は0です。ピッチについては、110 ページのPITCH文を参照してください。

参照：AUDIO (94 ページ) , PITCH (110 ページ)

プログラム例：

```
10 /  
20 /  
30 /      TRANSPOSE TEST  
40 /  
50 /  
60 CALL AUDIO (0,9)  
70 CALL TRANSPOSE(0)
```

```

80 CLS
90 LOCATE 7,10
100 PRINT "TRNSPOSE=";T
110 LOCATE 7,12
120 PRINT "UP=1    DOWN=2"
130 K#=INKEY$
140 IF K#="" THEN 130
150 K=VAL(K#)
160 IF K<1 OR K>2 THEN 130
170 ON K GOSUB 190,230
180 GOTO 90
190 '
200 T=T+5:IF T>12799 THEN T=12799
210 CALL TRANSPOSE (T)
220 RETURN
230 '
240 T=T-5:IF T<-12799 THEN T=-12799
250 CALL TRANSPOSE (T)
260 RETURN

```

プログラム例解説：

このプログラムは、ミュージックキーボードで演奏をする前にチューニングするときにお使いいただけます。

RUNさせると、現在設定されているトランスポーズ値を画面に表示します。

このときに接続したミュージックキーボードを押すとトランスポーズ値に応じた音が出ますので、ミュージックキーボードの適当なキーを押しながら、チューニングする相手の楽器の同じ音を鳴らし、音高を比較します。音高を上げたいときは **1** キーを、音高を下げたいときは **2** キーを押してください。指示した方に5セントだけ移調し、画面のトランスポーズ値の表示を書き換えます。チューニングが終わるまでこの操作を繰り返してください。

チューニングが終わったら、**CTRL** キーを押しながら **STOP** キーを押してください。チューニングされたトランスポーズ値の設定のまま、プログラムの実行を中止します。

トランスポーズ値を元に戻すには、下の命令を実行してください。

```
CALL TRANSPOSE (0)
```

[TRANSPOSE]

ミュージックキーボードを接続していないかたは、次の3行を追加してみてください。

```
173 CALL KEY ON (60)
175 FOR N=1 TO 200 :NEXT N
177 CALL KEY OFF (60)
```

RUNさせてから $\boxed{1}$ または $\boxed{2}$ のキーを押すと、トランスポーズ値に応じた音が鳴ります。

VOICE COPY

(ボイス・コピー) [ステートメント]

機能： FM音源用の音色パラメータデータの転送を行います。

書式： CALL VOICE COPY (<パラメータ1>, <パラメータ2>)

パラメータ1 = @ + 単純変数または定数
 または、配列変数名
 または、「*」(アスタリスク)

パラメータ2 = @ + 単純変数または定数
 または、配列変数名
 または、「*」(アスタリスク)

文例： DIM A (128)
 CALL VOICE COPY (*, A)
 ・音色番号32～63のすべてのデータを配列Aに転送する。

CALL VOICE COPY (@0, PIANO)
 ・音色番号0のデータを配列「PIANO」に転送する。

CALL VOICE COPY (@1, @32)
 ・音色番号1のデータを音色番号32に転送する。

解説： パソコンのメモリに設定した配列とMSXオーディオユニット内蔵のFM音源用音色データの間でのデータの転送を行います。
 パラメータ1で指定された音色データをパラメータ2に転送します。
 @ (アットマーク) と単純変数または定数が指定されたときは、その定数または変数の値で指定される音色番号の音色データが転送の対象になります。

パラメータ1（コピーされる元のデータの指定）では、0～63までのすべての音色番号を指定することができますが、パラメータ2（コピーの転送先の指定）では、0～31までの音色番号は指定できません。これらの音色番号のデータは、起動後も、書き変えのできないROM（ロム）に置かれているからです。

パラメータ1または2に@記号がない変数名を使用した場合は、その変数は配列変数とみなされ、その内容が転送の対象になります。

1つの音色データは32バイトの長さがあります。1つの配列変数には、特に配列の大きさを宣言しなくても88バイトの大きさ（長さ）が割り当てられますから、1つの配列変数名に1つの音色データを転送する場合には、あらかじめ配列の大きさを宣言しておく必要はありません。

*は音色番号32～音色番号63までのすべてのデータの意味で使います。*を使うときは、他方のパラメータは1キロバイト以上の大きさを持つ配列変数名にしてください。（32バイト×32個＝1024バイト）この場合には、使用する配列変数の大きさをあらかじめ宣言しておく必要があります。倍精度実数による配列変数の場合では、添え字1につき8バイトのメモリが確保されますから、配列変数の宣言のときに添え字を128以上にしてください。

音色のデータをフロッピーディスクにセーブするときには、まず配列変数に転送しておいてCOPY文によりその配列変数の内容をフロッピーディスク上にオープンしたファイルに転送します。

（ただし、COPY文はMSX1仕様のパソコンでは使用できません。）

参照：AUDIO（94 ページ）、FM音源用音色データ一覧表（186 ページ）

TEMPER

(テンペラメント) [ステートメント]

機能： FM音源に音律（テンペラメント）を設定します。

書式： CALL TEMPER (<音律番号>)

音律番号：0～21

文例： CALL TEMPER (0)

解説： FM音源の楽音の音高に影響を与えて音階を特定の音律に設定する命令です。

音律とは、1オクターブをどのような比率で12音に分割するかを決めるもので、古典音楽には古典音律が適していると言われています。

AUDIO文で初期化した後の値（初期値）は9番の完全平均律です。

参照：AUDIO (94 ページ)

[TEMPER]

番号	音律	番号	音律
0	ピタゴラス	10	純正律 C メジャー (Aマイナー)
1	ミーントーン	11	純正律 Cisメジャー (Bマイナー)
2	ヴェルクマイスター	12	純正律 D メジャー (Hマイナー)
3	ヴェルクマイスター (修正)	13	純正律 Esメジャー (Cマイナー)
4	ヴェルクマイスター (別)	14	純正律 E メジャー (Cis ♯)
5	キルンベルガー	15	純正律 F メジャー (Dマイナー)
6	キルンベルガー (修正)	16	純正律 Fisメジャー (Es ♯)
7	ヴァロッティ・ヤング	17	純正律 G .メジャー (Eマイナー)
8	ラモー	18	純正律 Gisメジャー (Fマイナー)
9	完全平均律 (初期値)	19	純正律 A メジャー (Fis ♯)
		20	純正律 B メジャー (Gマイナー)
		21	純正律 H メジャー (Gis ♯)

(修正) : 平島達司氏による

プログラム例：

```

10 '
20 '
30 '      TEMPERAMENT TEST
40 '
50 '
60 CALL AUDIO
70 CALL BGM (0)
80 CLS
90 LOCATE 5,12
100 INPUT "おんりつ の は"んご"う は (0~21)";K
110 IF K=99 THEN 200
120 CALL TEMPER (K)
130 PLAY#2,"V15CDEFGAB05C"
140 PLAY#2,"C04BAGFEDC"
150 K$=INKEY$
160 IF K$=CHR$(&H20) THEN 180
170 GOTO 130
180 CALL STOPM
190 GOTO 80
200 CALL STOPM
210 CALL AUDIO
220 END

```

プログラム例解説：

RUNさせると、音律番号の入力を求めてきますので、0から21までの数字を入力して **RETURN** キーを押してください。このときに、99を入力すると音律を元に戻してからプログラムの実行を停止します。0から21までの数字を入力した場合は、指定された音律で“ドレミファソラシド”を繰り返し演奏します。

音律を変えるには **SPACE** キーを押してください。再び音律番号の入力を求めてきます。このとき、ミュージックキーボードを弾くと直前に設定された音律で演奏できます。

■PCM音源の操作に関する命令

拡張BASICは「MSX-Audio」に内蔵されているPCM音源を使って、ADPCM録音／再生またはPCM録音／再生を行うことができます。PCM録音／再生では、音声は1サンプルデータ当たり8ビットのデータに変換して取り扱われるのに対して、ADPCM録音／再生では4ビットのデータに変換して取り扱われます。このため、音質はPCMのほうが優れていますが、同じ時間、音声を再生するのに必要なメモリ容量はADPCMの2倍になります。

ADPCM/PCMの音声データは番号をつけたファイルとして取り扱います。このファイルを音声ファイルと呼びます。MSXオーディオユニットでは、18種類のADPCM音声データが内蔵のROMに収められています。これらの音声データを特にROM音声ファイルと呼びます。ROM音声ファイルの内容については、この章の終わりにあるROM音声ファイル一覧表をご覧ください。

MSXオーディオユニットには、音声ファイルを収めたROMの他に256キロビットのRAMが内蔵されています。これらのメモリはMSX-AudioがCPUとは無関係に直接管理するため、外部メモリと呼ばれます。外部メモリ内に置かれた音声ファイルの録音／再生はローカルモードの録音／再生と呼ばれ、MSX-AudioがCPUの介入なしに単独で行えます。

SET PCM

 (セット・ピーシーエム) [ステートメント]

機能： ADPCM/PCMの音声ファイルを初期化します。

書式： CALL SET PCM (<音声ファイル番号>, <デバイス番号>, <モード>, <パラメータ1>, <パラメータ2> [, <サンプリング周波数>])

文例： CALL SET PCM (0, 0, 0, , 32)

解説： <音声ファイル番号>で指定された音声ファイルの初期設定をします。
<音声ファイル番号>は0から15までの数字で指定し、次のPCM関係の命令で参照される番号を実際のデータと結び付ける役割をします。

CONVA
CONVP
COPY PCM
LOAD PCM
MK PCM
PLAY PCM
REC PCM
SAVE PCM

<デバイス番号>は次ページの表に示すデバイス（装置：ここではMSX-Audioが利用することのできるいくつかのメモリのこと）を音声ファイルの格納場所に指定します。

指定されたデバイスの性質により、その他のパラメータの内容が次ページの表に示すように変わります。

デバイス番号	デバイス名	モード	パラメータ1	パラメータ2
0	外部RAM	0/1	—	長さ
5	VRAM	0/1	アドレス	長さ

—: 省略すること

アドレスと長さの単位は256バイト

<モード>はPCMのモードを指定し、0のときはADPCMモードに、1のときはPCMモードに設定されます。

アドレスは、パソコンのVRAM（ビラム：画像表示用のメモリ）のうちの使用されていない画面領域に音声ファイルを格納するときに、音声ファイルを収める場所の先頭の番地を入力します。

ご注意 MSX1パソコンではデバイスとしてVRAMを指定できません。

VRAM128キロバイトのパソコンでは、アドレスの範囲は1から（512-長さ）まで指定できます。（ただし、ディスプレイページに音声ファイルを格納すると画像が乱れます。また、音声も正常に再生されないことがあります。）

<サンプリング周波数>は音声をデータに変換するときの間隔を設定するパラメータです。単位は「Hz」（ヘルツ）で、範囲は1800～16000Hzです。AUDIO文で初期化した直後の値（初期値）は8000Hzです。サンプリング周波数が高いほど音質は良くなりますが、その分ファイルは大きくなります。ファイルの大きさを最大にしたときの記録可能時間は次の通りです。

サンプリング周波数を8000Hzとして、

PCM音声ファイルの場合…………… $256\text{Kbit} \div (8\text{KHz} \times 8\text{bit}) = 4\text{秒}$

ADPCM音声ファイルの場合… $256\text{Kbit} \div (8\text{KHz} \times 4\text{bit}) = 8\text{秒}$

AUDIO文で初期化された直後には、音声ファイル番号0が外部RAM 256キロビット分に割り当てられ、他の音声ファイルは長さが0になっています。つまり次のSET PCM文が実行されたのと同じ状態になっています。

```
CALL SET PCM (0, 0, 0, , 128)
```

参照：AUDIO (94 ページ)

プログラム例：

```
10 /
20 /
30 /      SET PCM TEST
40 /
50 /
60 CALL AUDIO
70 /
80 FOR N=0 TO 17
90 READ L,T
100 CALL SET PCM (0,0,0,,L)
110 CALL COPY PCM (#N,0)
120 /
130 CLS
140 LOCATE 5,12
150 PRINT "ファイル は" んこ"う =" ;N
160 /
170 CALL PLAY PCM (0)
```

```

180
190 K#=INKEY#
200 IF K#=CHR#(&H20) THEN 290
210 IF K#=CHR#(&H1B) THEN 320
220 '
230 FOR J=1 TO T
240 NEXT J
250 '
260 '
270 GOTO 170
280 '
290 CALL STOPM
300 NEXT N
310 '
320 END
330 '
340 DATA 35,0,33,1500,20,1300,
5,200,29,1200,43,1500,26,1700,
11,1000,4,110
350 DATA 1,500,6,250,4,80,2,80
,5,600,9,1000,9,1000,6,900,7,9
00

```

プログラム例解説：

RUNさせると、外部ROMに内蔵されているADPCM音源用ROM音声ファイルを0番から順に読み出しては SPACE キーを押すまで再生を繰り返します。

SPACE キーを押すと、次のファイルを読み出し、再生します。

途中で ESC キーを押すと、プログラムの実行を停止します。

90行のREAD文により、340行以降のDATA文からデータを2つつ読み取っては変数L（音声ファイルの長さ）と変数T（再生を繰り返す間隔）に代入します。

変数Lは1000行のSET PCM文で<長さ>として使用し、変数Tは230行のFOR～NEXT文で時間調整のためのループの回数を決めるのに使用しています。

COPY PCM

 (コピー・ピーシーエム) [ステートメント]

機能: ADPCM/PCMデータを転送します。

書式: CALL COPY PCM (<ファイル番号1>, <ファイル番号2> [, <オフセット1>] [, <長さ>] [, <オフセット2>])

文例: CALL COPY PCM (1, 2)
ファイル1の全体をファイル2に転送します。

解説: 2つのADPCMまたはPCM音声ファイルの間でデータの全体のコピーや一部分だけのコピーを行うことができます。

<ファイル番号1>の先頭から後ろに<オフセット1>の位置から、<長さ>のデータをコピーし、<ファイル番号2>の先頭から後ろに<オフセット2>の位置を先頭として転送します。

オフセットと長さの単位は256バイトを1とします。

ファイルがADPCMのファイルかそれともPCMのファイルかというタイプのチェックを行いません。また、転送後の転送先のファイルの長さやタイプを変えないため、転送先のファイルをあらかじめ転送されるデータのタイプと長さに合わせて設定しておく必要があります。

パラメータは2つの音声ファイル番号以外は省略可能です。省略時の値は<オフセット1>・<オフセット2>については0、<長さ>については<ファイル番号1>の最後までになります。

[COPY PCM]

<ファイル番号の1>として「#」の付いた数字を使用すると、内蔵のADPCM音源用ROM音声ファイルからのADPCMデータのコピーを実行します。ROM音声ファイルはCOPY PCM文により一度他の音声ファイルに転送してから再生します。

ROM音声ファイルの内容については、188 ページのROM音声ファイル一覧表を参照してください。

ご注意 内蔵のROM音声ファイルに対して他のファイルからデータを転送することはできません。

参照：AUDIO (94 ページ) , SET PCM (122 ページ) , PLAY PCM (128 ページ) , REC PCM (135 ページ)

PLAY PCM

(プレイ・ピーシーエム) [ステートメント]

機能: ADPCM/PCM音声ファイルを再生します。

書式: CALL PLAY PCM (<音声ファイル番号> [, <rep> [, <オフセット> [, <長さ> [, <サンプリング周波数>]]]])

文例: CALL PLAY PCM (0)

解説: <音声ファイル番号>で指定されたADPCMまたはPCM音声ファイルの<オフセット>の位置から<長さ>だけのデータを<サンプリング周波数>で再生します。

<音声ファイル番号>は0～15の数字で入力して、再生する音声ファイルを指定します。

<rep>は0または1で指定します。1を指定したときにはリピートモードになり、いつまでも再生を繰り返します。

このモードでは、END文の実行(バックグラウンド処理を行っている場合。バックグラウンド処理をしていない場合は、END文は実行されません。)または **CTRL** + **STOP** の入力によりプログラムの実行を中止しても、音声の再生は止まりません。ダイレクトモードでCALL STOPM命令またはCALL AUDIO命令を入力すれば再生を中止させることができます。バックグラウンド処理を行っているプログラムでは、END文の前にCALL STOPM文を実行させることにより再生を中止させることができます。

<オフセット>と<長さ>は、256バイト単位で指定します。

<サンプリング周波数>は、外部RAM（デバイス番号0）に格納した音声ファイルをADPCM再生するとき（ローカルモード）に限り1800～49716Hzまでの範囲で指定できます。その他のデバイスの場合は1800～16000Hzの範囲で指定します。

各パラメータは音声ファイル番号以外は省略可能で、省略したときには次のように設定されます。

<rep>	= 0
<オフセット>	= 0
<長さ>	= ファイルの終わりまで
<サンプリング周波数>	= SET PCM文で設定された値

途中のパラメータを省略する場合には、省略したパラメータの位置に「,」（カンマ）を入れることが必要です。

参照：AUDIO (94 ページ) , SET PCM (122 ページ) , COPY PCM (126 ページ) , REC PCM (135 ページ)

プログラム例：

```
10 '
20 '
30 '      PLAY PCM TEST
40 '
50 '
60 CALL AUDIO
70 '
80 CALL SET PCM (0,0,0,,4)
90 CALL COPY PCM (#8,0)
100 '
110 F=6500
120 FOR N=0 TO 18
```

```
130 CALL PLAY PCM (0,,,F)
140 FOR J=1 TO 110
150 NEXT J
160 F=F+500
170 NEXT N
180 '
190 END
```

プログラム例解説：

RUNさせるとウイスキーを注ぐ音が聞こえます。

8番のROM音声ファイルを再生するのに、再生を繰り返すたびにサンプリング周波数を上げることによって音の高さを次第に高くしています。元の音声ファイルより本物らしく聞こえるでしょうか？

PCM FREQ

(ピーシーエム・フリークエンシー) [ステートメント]

機能: ADPCM音声ファイルのローカルモード再生実行中にサンプリング周波数を変更します。

書式: CALL PCM FREQ (<サンプリング周波数>)

文例: CALL PCM FREQ (24000)

解説: 外部メモリ内でADPCM音声ファイルを再生中にサンプリング周波数を変更するための命令です。<サンプリング周波数>の値の範囲は18000~49716で、単位はHzです。

SET PCM命令のパラメータとしての<サンプリング周波数>(123 ページ参照)に比べて、値の範囲が広いことと再生中に変更できることに注目してください。

参照: AUDIO (94 ページ) , SET PCM (122 ページ) , REC PCM (135 ページ) , PLAY PCM (128 ページ)

プログラム例:

```
10 '  
20 '  
30 '          PCM FREQ TEST  
40 '  
50 '  
60 CALL AUDIO  
70 '  
80 Q*(1)="はし"め:" : Q*(2)="あわりの"  
90  FOR N=1 TO 2  
100  CLS  
110  LOCATE 0,12
```

```

120 PRINT Q$(N); " しゅうはすう は (1800
~49716)" :PRINT:INPUT K(N)
130 IF K(N)<1800 OR K(N)>49716!
THEN 100
140 NEXT N
150 IF K(2)>K(1) THEN S=100 ELSE S
=-100
160 L=ABS(K(2)-K(1))
170 '
180 CALL SET PCM (0,0,0,,33)
190 CALL COPY PCM (#1,0)
200 CALL PLAY PCM (0)
210 FOR J=1 TO L STEP S
220 CALL PCM FREQ (K(1)+J-1)
230 NEXT J
240 K$=INKEY$
250 IF K$=CHR$(&H20) THEN 90
260 GOTO 240

```

プログラム例解説：

RUNさせるとスタート時の周波数を聞いてきますから、数字を入力してから **RETURN** キーを押してください。変化を聞き取りやすくするためには1800～3000ぐらいの小さな数字が良いでしょう。次に終了時の周波数を聞いてきますから同じ様に数字を入力してください。今度は8000以上の数字をお勧めします。 **RETURN** キーを押すと二フトリがおかしな鳴きかたをします。鳴き終わってから **SPACE** キーを押すと再びスタート時の周波数を聞いてきます。プログラムの実行を中止したいときは、 **CTRL** + **STOP** を押してください。

PCM VOL

(ピーシーエム・ボリューム)

[ステートメント]

機能： ADPCM/PCM再生の音量を設定します。

書式： CALL PCM VOL (<ボリューム値>)

文例： CALL PCM VOL (63)

解説： ボリューム値の範囲は0～63でボリューム値が8変化するごとに6dBの割合で変化します。AUDIO文で初期化したときの値(初期値)はADPCMでは55、PCMでは32になっています。

参照： AUDIO (94 ページ) , SET PCM (122 ページ) ,
PLAY PCM (128 ページ)

プログラム例：

```
10 '  
20 '  
30 '          PCM VOL TEST  
40 '  
50 '  
60 CALL AUDIO  
70 CALL SET PCM (0,0,0,,1)  
80 CALL COPY PCM (#9,0)  
90 '  
100 FOR N=40 TO 63  
110     CALL PCM VOL (N)  
120     CALL PLAY PCM (0)  
130     '  
140     FOR J=1 TO 200  
150     NEXT J  
160 '
```

```
170 NEXT N
180 '
190 FOR N=63 TO 40 STEP -1
200     CALL PCM VOL (N)
210     CALL PLAY PCM (0)
220     '
230     FOR J=1 TO 200
240         NEXT J
250     '
260 NEXT N
270 '
280 END
```

プログラム例解説：

テレビの音量を最大にしてからRUNさせてください。

耳を澄まして聞いていると誰かがこちらに歩いて来て、また遠ざかります。

200行のPCM VOL文で再生するたびに音量を変化させています。

REC PCM

(レコード・ピーシーエム) [ステートメント]

機能： 音声をADPCM/PCM音声ファイルに録音します。

書式： CALL REC PCM (<音声ファイル番号> [, <SYNC>]
[, <オフセット>] [, <長さ>] [, <サンプリング周波数>])

文例： CALL REC PCM (0)

解説： <音声ファイル番号>で指定されたADPCM/PCM音声ファイルの
<オフセット>の位置から<長さ>だけ<サンプリング周波数>で録音
します。

<音声ファイル番号>は0～15の数字で入力し、録音する音声ファイル
を指定します。

<SYNC>は0または1で指定し、0が指定されるとシンクロスタート
モードになり、マイク入力端子に音声信号が入ってくるまで待ってから録
音を始めます。

<オフセット>と<長さ>は、256バイトを単位とします。

各パラメータは音声ファイル番号以外は省略可能です。

省略時の値(初期値)は次のようになります。

<SYNC>	= 0	(シンクロスタート機能あり)
<オフセット>	= 0	(ファイルの先頭から)
<長さ>	=	ファイルの最後まで
<サンプリング周波数>	= SET	PCM文で設定された値

参照：AUDIO (94 ページ) , SET PCM (122 ページ) , PLAY PCM
(128 ページ)

プログラム例：

```

10 '
20 '
30 '      REC PCM TEST
40 '
50 '
60 CALL AUDIO (3,0)
70 CALL SET PCM (0,0,0,,32)
80 CALL BGM (0)
90 '
100 CLS
110 LOCATE 5,10
120 INPUT "マイクにゆりよく を と"うぞ" !" ;K$
130 '
140 CALL REC PCM (0)
150 CALL MK PCM (0)
160 '
170 CLS
180 LOCATE 5,10
190 INPUT "えんぞう を と"うぞ" !" ;K$
200 '
210 GOTO 100

```

プログラム例解説：

このプログラムは、マイクから入力した音声をADPCM音声ファイルに録音し、録音が終わるとその音声ファイルをミュージックキーボードに割り当てて、押されたキーの音の高さで再生されるようにします。これでお手持ちのミュージックキーボードがサンプリング・キーボードに変身します。

このプログラムを実行する前には、FS-CA1にミュージックキーボードとマイクを接続する必要があります。

正しい接続のしかたは19 ページに載せてありますので参照してください。

RUNさせると、画面にマイク入力を求める表示が現れます。
用意ができたなら、**RETURN** キーを押して、録音を始めてください。
録音時間は約2秒です。

録音が終わったら、画面に演奏を求める表示が現れますから、ミュージックキーボードのキーを押してみてください。いま録音した音声は再生されます。

新しい音声を録音したくなったら、再び **RETURN** キーを押して最初の表示に戻ります。

高い音のキーを押したときと、低い音のキーを押したときとは、再生される音の高さだけでなく、再生時間が違うことに注目してください。これは拡張BASICが、押されたキーの高さに応じた音でADPCM音声ファイルを再生するために、再生時のサンプリング周波数を変えていることを示しています。

シンクロスタートモードでは、音声が入力されてから実際に録音が始まるまでにわずかに時間がかかります。このため、録音したい音声の最初の部分が録音されないことがあります。

また、最初に入力される音があまり小さいと録音を開始しないことがあります。このような場合には、140行を下のように修正してください。

```
140 CALL REC PCM (0, 1)  
           _____ 修正した部分
```

この修正を行った場合は、**RETURN** キーを押すと、すぐに録音が始まります。

録音時間を調整したいときは、70行の()の中の32という数字を1から128の範囲で変更してください。

プログラムの実行を停止したいときは、**CTRL** + **STOP** を押してください。

SAVE PCM

(セーブ・ピーシーエム) [ステートメント]

機能: ADPCM/PCM音声ファイルをディスクにセーブする。

書式: CALL SAVE PCM (<ファイル名>, <音声ファイル番号>)

文例: CALL SAVE PCM ("DEMO2.DAT", 2)

- ・ Aドライブの中のフロッピーディスクに2番の音声ファイルの内容を「DEMO2.DAT」というファイル名でセーブします。

CALL SAVE PCM ("B:SAMPLE11", 0)

- ・ Bドライブの中のフロッピーディスクに0番の音声ファイルの内容を「SAMPLE11」というファイル名でセーブします。

A\$="A:サウンド8.CA1"

CALL SAVE PCM (A\$, 1)

- ・ Aドライブの中のフロッピーディスクに1番の音声ファイルの内容を「サウンド8.CA1」というファイル名でセーブします。

解説: <音声ファイル番号>で指定された音声ファイルを<ファイル名>で指定されたファイル名でフロッピーディスクにセーブします。
 <ファイル名>はDISK BASICの説明書の中のファイルスペックについての説明にしたがって作られた文字列である必要があります。
 ファイルスペックの書式は次の通りです。
 [ドライブ名+" ":""] +最大8文字までのファイル名+ [". " +最大3文字までの拡張子]

[SAVE PCM]

正しいファイルスペックを内容とするものであれば、<ファイル名>に文字変数を使用することもできます。

参照： AUDIO (94 ページ) , SET PCM (122 ページ) ,
REC PCM (135 ページ) , LOAD PCM (141 ページ)

プログラム例：

```
10  '  
20  '  
30  '      SAVE PCM TEST  
40  '  
50  '  
60  CALL AUDIO  
70  CALL BGM (0)  
80  CALL SET PCM (0,0,0,,48)  
90  '  
100 CLS  
110 LOCATE 5,10  
120 PRINT "マイク にゅりょく スタート !"   
130 CALL REC PCM (0)  
140 '  
150 CLS  
160 LOCATE 5,10  
170 INPUT "ファイルめいは";F#  
180 CALL SAVE PCM (F#,0)  
190 '  
200 FILES  
210 PRINT F#;"の セーフ" おわり !"  
220 END
```

プログラム例解説：

このプログラム例の実行には、マイクおよびフロッピーディスクドライブとフォーマット済みのフロッピーディスクが必要です。

プログラム例を入力したあと、

- ① フロッピーディスクのライトプロテクトタブが書き込み可能な位置にあることを確認します。
- ② ドライブAのフロッピーディスクドライブにフロッピーディスクを入れます。

RUNさせると画面にマイク入力を求めるメッセージが表示されますから、マイクのスイッチを入れて音声を入力してください。

約3秒間で録音が終わる、画面にファイル名の入力を求めるメッセージが表示されます。ファイル名を入力し、**RETURN**キーを押してください。録音した音声を入力したファイル名でフロッピーディスクにセーブします。

ご注意 画面にセーブの完了を示すメッセージが表示されるまで、フロッピーディスクを取り出したり、電源スイッチを切ったりしないでください。

LOAD PCM

(ロード・ピーシーエム) [ステートメント]

機能： ADPCM/PCM音声ファイルをフロッピーディスクからロードする。

書式： CALL LOAD PCM (<ファイル名>, <音声ファイル番号>)

文例： CALL LOAD PCM ("DEMO.DAT", 1)

CALL LOAD PCM ("B:SAMPLE11", 0)

AS="A:サウンド8. CA1"

CALL LOAD PCM (AS, 2)

解説： <音声ファイル番号>で指定された音声ファイルに<ファイル名>で指定されたフロッピーディスク上のファイルからADPCM/PCMデータをロードします。

SET PCM文で設定した音声ファイルの長さよりフロッピーディスク上のファイルが長いときは、設定された音声ファイルの長さの分だけデータをロードします。音声ファイルのデータ型式 (ADPCM/PCM) とサンプリング周波数は、ロードしたデータに合わせて再設定されます。

<ファイル名>はDISK BASICの説明書の中のファイルスペックについての説明にしたがって作られた文字列である必要があります。ファイルスペックの書式は次の通りです。

[ドライブ名+":"] +最大8文字までのファイル名+ [". " +最大3文字までの拡張子]

正しいファイルスペックを内容とするものであれば、<ファイル名>に文字変数を使用することもできます。

高速化のためにBASICのフリーエリアを転送時のバッファとして使
用します。したがってフリーエリアが少ないときにはロード時間が長
くなります。

参照： AUDIO (94 ページ) , SET PCM (122 ページ) , PLAY PCM
(128 ページ) , MK PCM (150 ページ) , SAVE PCM (138 ページ)

プログラム例：

```
10 '
20 '
30 '      LOAD PCM TEST
40 '
50 '
60 CALL AUDIO
70 CALL BGM (0)
80 CALL SET PCM (0,0,0,,48)
90 '
100 FILES
110 PRINT:INPUT "ファイル名は";F$
120 '
130 CALL LOAD PCM (F$,0)
140 '
150 CLS
160 LOCATE 5,10
170 PRINT F$;" の ロート" おわり!"
180 '
190 CALL PLAY PCM (0)
200 '
210 END
```


プログラム例解説：

このプログラムは、フロッピーディスクドライブを接続しているかたで SAVE PCM命令のプログラム例を使って音声ファイルのセーブを行ったかた以外は実行できません。

音声ファイルをセーブしたフロッピーディスクを、ドライブAに入れてください。

RUNさせるとフロッピーディスクにセーブされているすべてのファイルのファイル名を表示したあと、ファイル名の入力を求めてきますので、ロードする音声ファイルのファイル名を入力して **RETURN** キーを押してください。

音声ファイルをロードしたあと、再生を行います。

ご注意

SAVE PCM命令のプログラム例を使ってセーブした音声ファイル以外のファイルはロードできません。

CONVA

(コンバート・エイ)

[ステートメント]

機能： PCM形式のデータをADPCM形式のデータに変換します。

書式： CALL CONVA (<元のファイルの番号>, <変換したファイルの番号>)

文例： CALL CONVA (1, 2)

1番の音声ファイルに収められているPCMデータ全体をADPCMデータに変換して2番の音声ファイルに収めます。

解説： PCM形式のデータをADPCM形式のデータに変換します。
<元のファイル番号>と<変換したファイルの番号>は、異なっている必要があります。

データの長さは1/2になりますが、同じサンプリング周波数で再生したときの再生時間は変わりません。

この命令を実行すると、元のファイルのデータの形式がPCMであることを確認し、異なっていると「Illegal function call」と画面に表示してプログラムの実行を中止します。

変換したデータを収めるファイルは外部RAMまたはVRAMのいずれか(内容の変更可能なデバイス)を使用していなければなりません。また、ファイルの形式と長さは変換後のデータに合わせて再設定されます。同時に、サンプリング周波数は元のファイルに合わせて再設定されます。

参照： AUDIO (94 ページ), SET PCM (122 ページ), CONVP (145 ページ)

CONVP

(コンバート・ピー)

[ステートメント]

機能： ADPCM形式のデータをPCM形式のデータに変換します。

書式： CALL CONVP (<元のファイルの番号>, <変換したファイルの番号>)

文例： CALL CONVP (1, 2)

1番の音声ファイルに収められているADPCMデータ全体をPCMデータに変換して<変換したファイルの番号>で指定される音声ファイルに格納します。

解説： ADPCM形式のデータをPCM形式に変換します。

<元のファイルの番号>と<変換したファイルの番号>は、異なる必要があります。

変換したデータの長さは元のデータの長さの2倍になりますが、同じサンプリング周波数で再生したときの再生時間の長さは変わりません。この命令を実行すると、元のファイルの形式がADPCMであることを確認し、異なっているときは画面に「Illegal function call」と表示してプログラムの実行を中止します。

変換したデータを収めるファイルは外部RAMまたはVRAMのいずれか(内容の変更可能なデバイス)を使用していなければなりません。また、ファイルの形式と長さは変換後のデータに合わせて再設定されます。同時に、サンプリング周波数は元のファイルに合わせて再設定されます。

参照： **AUDIO** (94 ページ), **SET PCM** (122 ページ), **CONVA** (144 ページ)

■ミュージックキーボード関係の命令

AUDIO文でミュージックキーボードに割り当てられたFM音源のチャンネルは、以後、ミュージックキーボードを弾くことによって直接に操作することができます。拡張BASICでは、このようなミュージックキーボードとFM音源の組み合わせをインストゥルメント（楽器）と呼びます。

インストゥルメントの操作はパソコンのCPUに関係なくバックグラウンド（背景）で処理されるので、MSXオーディオシステムをミュージックキーボードにより演奏する楽器としての使い方をBASICでのプログラムやコマンドの実行と独立して行うことができます。

MK VOICE

(エムケー・ボイス) [ステートメント]

機能： インストゥルメントの音色を設定します。

書式： CALL MK VOICE (<パラメータ>)

<パラメータ>： @+単純変数
または 配列変数名

文例： CALL MK VOICE (@2)

解説： インストゥルメントの音色の種類を設定します。

@（アットマーク）記号とともに単純変数がパラメータとして与えられたときは、内蔵のFM音源用音色データの音色番号を指定します。

音色番号は0～63の範囲で指定できます。

内蔵のFM音源用音色データについては、186 ページを参照してください。

@記号がない場合には、配列変数がパラメータとして与えられたとみなし、その配列の内容が音色のデータになります。

♪ 音色データの自作には、FM音源とMSX-Audioについての詳しい知識が必要ですので、本書では説明を省略します。

参照： AUDIO (94 ページ) , VOICE (107 ページ) , FM音源用音色データ一覧表 (186 ページ)

プログラム例：

```
10 '
20 '
30 '      MK VOICE TEST
40 '
50 '
60 CALL AUDIO (0,9)
70 SCREEN 1,,0
80 CLS
90 LOCATE 5,10
100 INPUT "ねいろ は (1から62まで)" ; K
110 IF K<1 OR K>62 THEN 80
120 CALL MK VOICE (@K)
130 GOTO 80
```

プログラム例解説：

RUNさせるとミュージックキーボードに設定する音色の音色番号の入力を求めてきますので、186 ページのFM音源用音色データ一覧表を参照し、選んだ番号の数字キーを押してから **RETURN** キーを押してください。画面から入力した文字が消えたら、ミュージックキーボードを弾いてみてください。いま入力した番号の音色で演奏できます。演奏中に音色を切り換えたくなったら、新しい音色の音色番号を入力してください。**RETURN** キーを押すと音色が切り換わります。プログラムを停止させたいときは、**CTRL** + **STOP** を押してください。

MK VOL

(エムケー・ボリューム)

[ステートメント]

機能： インストゥルメントの音量を設定します。

書式： CALL MK VOL (<ボリューム値>)

文例： CALL MK VOL (40)

解説： インストゥルメントの音量を設定します。<ボリューム値>の範囲は0～63で、AUDIO文による初期化を行ったあとの値(初期値)は55です。

ミュージックキーボードのキーを押している途中でこの命令を実行すると、鳴っている音の音量が途中から変化します。

参照： AUDIO (94 ページ)

プログラム例：

```

10 '
20 '
30 '      MK VOL TEST
40 '
50 '
60 CALL AUDIO (0,9)
70 V=55
80 '
90 CLS
100 LOCATE 7,12
110 PRINT "おんりょう =" ; V
120 '
130 K#=INKEY$
140 IF K#=CHR$(&H1E) THEN 180
150 IF K#=CHR$(&H1F) THEN 220

```

```
160 GOTO 130
170 '
180 V=V+1
190 IF V>63 THEN V=63
200 GOTO 260
210 '
220 V=V-1
230 IF V<0 THEN V=0
240 GOTO 260
250 '
260 CALL MK VOL (V)
270 GOTO 90
```

プログラム例解説

RUNさせると、現在設定されている音量の値を画面に表示します。
カーソルキー \blacktriangleleft を押すと設定されている音量が大きくなり、カーソル
キー \blacktriangleright を押すと音量が小さくなります。
ミュージックキーボードのキーを押して確かめてください。

CTRL + **STOP** を入力すると、最後に表示された
音量の値に設定されたまま、プログラムの実行を停止します。
音量の設定を初めの状態に戻すには、次の命令を実行してく
ださい。

```
CALL MK VOL (55)
```

MK PCM

(エムケー・ピーシーエム)

[ステートメント]

機能： インストゥルメントとして演奏するADPCMの音のファイル番号を指定します。

書式： CALL MK PCM (<音声ファイル番号>)
音声ファイル番号：0～15

CALL MK PCM (OFF)
音声ファイルの指定を解除。

文例： CALL MK PCM (1)

解説： インストゥルメントでADPCMを使って演奏する音声ファイル番号を指定または解除します。

指定する音声ファイルは、あらかじめ外部RAM（デバイス番号0）にADPCMで録音されている必要があります。

ご注意 PCMタイプの音声ファイルは指定できません。

参照： AUDIO (94 ページ) , SET PCM (122 ページ) , COPY PC (126 ページ) , REC PCM (135 ページ)

プログラム例：

```

10 '
20 '
30 '      MK PCM TEST
40 '
50 '
60 CALL AUDIO (0,0)
70 DIM A(17)
80 FOR N=0 TO 17
90     READ A(N)
100 NEXT N
110 CLS
120 LOCATE 5,12
130 INPUT "ADPCMの ぬいろ は ";K
140 IF K>=0 AND K<=17 THEN 170

```



```

150 GOTO 110
160 '
170 S=A(K)
180 CALL SET PCM (0,0,0,,S)
190 CALL COPY PCM (#K,0)
200 CALL MK PCM (0)
210 GOTO 110
220 '
230 DATA 35,33,20,5,29,43
240 DATA 26,11,4,1,6,4
250 DATA 2,5,9,9,6,7

```

プログラム例解説：

RUNさせるとミュージックキーボードに設定する音色の音色番号を聞いてきますので、188 ページのROM音声ファイル一覧表を参照し、番号の数字を入力して **RETURN** キーを押してください。ミュージックキーボードのキーを押すと指定した音声ファイルの音が押したキーの音の高さで鳴ります。

ファイルの長さが長い音声ファイルを指定した場合、キーを押してから音が鳴るまで少し時間がかかります。また押したキーの音の高さによって、音の鳴っている時間が変わります。

60行のCALL AUDIO文に注目してください。パラメータを(0,0)に設定しています。

これは、インストゥルメントへのFM音源の割り当てを0に設定することにより、ミュージックキーボードのキーを押してもFM音源が鳴らないようにするための設定です。この設定をしなくてAUDIO文のパラメータが初期値に設定されたまま演奏すると、PCM音源の音といっしょにFM音源の音も鳴ってしまいます。

CTRL+**STOP** の入力により、プログラムの実行は停止されますが、ミュージックキーボードには最後に割り当てられたROM音声ファイルの音がそのまま残ります。元のFM音源の音色に戻すには次の命令を実行してください。

```
CALL AUDIO
```

MK VEL

(エムケー・ベロシティ) [ステートメント]

機能： インストゥルメントにベロシティを設定します。

書式： CALL MK VEL (<ベロシティ値>)

ベロシティ値： 0～15

文例： CALL MK VEL (15)

解説： インストゥルメントにベロシティを与えて初期化します。ベロシティとはキーボードのキーの押し下げられる速さ（タッチ速度）のことで、キーを押す力の強さをあらわします。これによって、FM音の音量とともに音質も変化します。

MSXオーディオシステム用のミュージックキーボードでは一定のベロシティしか発生しませんので、この命令によってその値を変化させます。初期値は8です。

この命令はインストゥルメントを初期化しますので、演奏中にこの命令を実行すると音は一度途切れます。

参照： AUDIO (94 ページ)

プログラム例：

```

10 '
20 '
30 '      MK VEL TEST
40 '
50 '
60 CALL AUDIO (0,9)
70 CALL MK VOICE (@14)
80 '
90 CLS

```

```
100 LOCATE 5,12
110 INPUT " ` "ロシティ=" ;U
120 IF U<0 OR U>15 THEN 80
130 '
140 CALL MK VEL (U)
150 GOTO 80
```

プログラム例解説：

RUNさせると設定するベロシティの値を聞いてきますので、0から15の範囲で数字を入力して **RETURN** キーを押してください。入力した数字が画面から消えたら、ミュージックキーボードのキーを押してみてください。

プログラムの実行を止めるには、**CTRL**+**STOP** を入力してください。最後に設定されたベロシティ値のまま停止します。

ベロシティ値を初めの状態に戻すには、次の命令を実行してください。

```
CALL MK VEL (8)
```

I N M K

(インフォーム・エムケー)

[関数]

機能： ミュージックキーボードの変化を知らせます。

書式： CALL I N M K [([<変数 1 >] [, [<変数 2 >] [, <変数 3 >]))]

文例： CALL I N M K

CALL I N M K (A)

CALL I N M K (A, B, C)

CALL I N M K (A, B)

CALL I N M K (, , C)

解説： ミュージックキーボードの演奏による変化を知らせます。

ミュージックキーボードのキーを押すと、そのたびに変化についての情報がMSXオーディオユニットの中のキーバッファと呼ばれる部分に貯えられていきます。

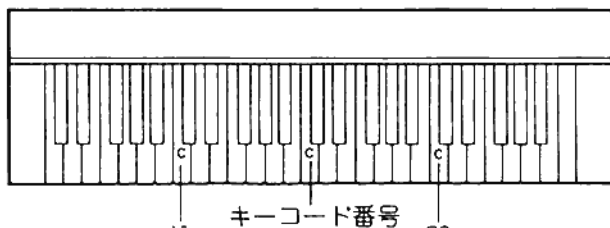
変数のいずれかを付けて実行したときは、キーバッファから1つの情報を取り出し、

変数 1 にはキーコード番号

変数 2 にはキーのON/OFF (0 : ON, 1 : OFF)

変数 3 にはキーコード番号に対応するADPCMの周波数を値として入れます。

キーコード番号は0～127の範囲で、中央C（ミュージックキーボードの中央に最も近い“ド”のキー）は60に対応します。



キーバッファが空のときは、各変数の値を0にします。

パラメータなしで実行した場合には、キーバッファを空にします。

キーバッファは、変化の回数にして32回分の情報を貯えられるだけの大きさがあります。キーバッファに溜まった情報は、先に入ったものから順に取り出されて音を発生するためのデータとして使用されます。キーバッファがいっぱいのときに33回目の情報が送られてくると、キーバッファを空にして、「Device I/O error」と画面に表示し、プログラムの実行を中止します。

参照： AUDIO (94 ページ)

プログラム例：

```
10 '  
20 '  
30 '          INMK TEST  
40 '  
50 '  
60 CALL AUDIO (0,9)  
70 CLS  
80 WIDTH 30  
90 LOCATE 0,0  
100 DIM C$(12)  
110 FOR N=1 TO 12  
120     READ C$(N)  
130 NEXT N  
140 '  
150 CALL INMK (A,B)  
160 IF B=1 THEN 190  
170 GOTO 150  
180 '  
190 IF A<48 THEN S$="-2"  
200 IF A>47 AND A<60 THEN S$="-1"  
210 IF A>59 AND A<72 THEN S$=" "  
220 IF A>71 THEN S$="+1"
```

```
230 H=(A MOD 12 )+1
240 C#=C$(H)
250 K#=S#+C#
260 PRINT K#;CHR$( &H20 );
270 GOTO 150
280 '
290 DATA "C ", "C#", "D ", "D#"
300 DATA "E ", "F ", "F#", "G "
310 DATA "G#", "A ", "A#", "B "
```

プログラム例解説：

RUNさせてミュージックキーボードのキーを押してみてください。
押したキーが画面に記号で表示されます。アルファベットは1オクターブ中のキーの位置を示し、アルファベットの前の正負記号と数字はオクターブの範囲を示します。中央Cを含む1オクターブは、正負記号、数字共に表示されません。

プログラムを止めるには、 **CTRL** + **STOP** を入力してください。

KEY ON/OFF

(キー・オン/オフ) [ステートメント]

機能: インストゥルメントにキーのオンまたはオフの情報を送ります。

書式: CALL KEY ON (<キーコード番号> [, <ベロシティ>])

CALL KEY OFF (<キーコード番号>)

文例: CALL KEY ON (60, 3) : CALL KEY OFF
(59)

解説: この命令を実行すると、キーが押された、または放されたという情報を、実際のキーの状態に関係なくインストゥルメントに送ります。

<キーコード番号>は0~127の範囲で、中央Cは60に対応します。

<ベロシティ>は0~15の範囲で、省略時は8に設定されます。

これにより、ミュージックキーボードを使って演奏したときのMSXオーディオシステムの動作をプログラムによって模倣することができます。ただし、この命令で作られたキーの状態に関する情報は、後述のMK記録の対象にはなりません。

参照: AUDIO (94 ページ)

プログラム例：

```
10  '  
20  '  
30  '      KEY ON/OFF TEST  
40  '  
50  '  
60  CALL AUDIO (0,9)  
70  FOR N=0 TO 127  
80    CLS  
90    LOCATE 7,10  
100   PRINT "キーコード"="";N  
110   CALL KEY ON (N)  
120   FOR J=1 TO 50  
130   NEXT J  
140   CALL KEY OFF (K)  
150 NEXT N  
160 '  
170 END
```

プログラム例解説：

RUNさせるとキーコード番号0から127までの音を連続して鳴らします。ミュージックキーボードではこれだけのキーの数は有りません。

MK TEMPO (エムケー・テンポ) [ステートメント]

機能： ミュージックキーボード演奏の記録／再生とメトロノーム機能の速度を設定します。

書式： CALL MK TEMPO ([<テンポ値>] [, <パーカッションマップ>])

文例： CALL MK TEMPO (60)

CALL MK TEMPO (60, 1)

CALL MK TEMPO (, 0)

解説： タイマーの周期をコントロールしてミュージックキーボード演奏の記録／再生機能（MK記録／再生）やメトロノーム機能の動作速度を設定します。

<テンポ値>は25から360の範囲で設定します。AUDIO文の実行による初期値は120です。

メトロノーム機能は、リズム音を設定されたテンポに合わせてメトロノームのように繰り返し鳴らす機能です。

<パーカッションマップ>は0から31までの範囲で設定し、メトロノーム機能で使用するリズム音を次ページの図のように指定します。

メトロノーム機能の利用には、あらかじめAUDIO文のパラメータ設定によりリズム音の使用を可能にしておく必要があります。（95 ページ参照）

パーカッションマップの初期値は0でメトロノーム機能は停止していません。<パーカッションマップ>を1以上に設定してMK TEMPO文を実行すると、プログラムの終了にもバックグラウンド機能の設定にもかかわらず、<パーカッションマップ>を0に設定してMK TEMPO文を実行するまでメトロノーム機能が働き続けます。

メトロノーム機能によるリズム音は、後述のMK記録の対象にはなりません。

この命令により、次の命令の実行の速度が影響を受けます。

MK PLAY
MK REC
MK APPEND

パーカッションマップ一覧表

○印： リズム音発生

×印： 無音

値	HH	SY	TT	SD	BD	値	HH	SY	TT	SD	BD
0	×	×	×	×	×	16	×	×	×	×	○
1	○	×	×	×	×	17	○	×	×	×	○
2	×	○	×	×	×	18	×	○	×	×	○
3	○	○	×	×	×	19	○	○	×	×	○
4	×	×	○	×	×	20	×	×	○	×	○
5	○	×	○	×	×	21	○	×	○	×	○
6	×	○	○	×	×	22	×	○	○	×	○
7	○	○	○	×	×	23	○	○	○	×	○
8	×	×	×	○	×	24	×	×	×	○	○
9	○	×	×	○	×	25	○	×	×	○	○
10	×	○	×	○	×	26	×	○	×	○	○
11	○	○	×	○	×	27	○	○	×	○	○
12	×	×	○	○	×	28	×	×	○	○	○
13	○	×	○	○	×	29	○	×	○	○	○
14	×	○	○	○	×	30	×	○	○	○	○
15	○	○	○	○	×	31	○	○	○	○	○

[MK TEMPO]

- ・HH・・・ハイハットシンバル音
- ・SY・・・トップシンバル音
- ・TT・・・タムタム音
- ・SD・・・スネアドラム音
- ・BD・・・バスドラム音

参照： **AUDIO** (94 ページ) , **PLAY MK** (168 ページ) , **REC MK**
(165 ページ) , **APPEND MK** (174 ページ)

プログラム例：

```
10  '
20  '
30  '      MK TEMPO TEST
40  '
50  '
60  CALL AUDIO (1,6)
70  '
80  CLS
90  LOCATE 5,10
100 INPUT "テンポ*(25~360)=";T
110 IF T<25 OR T>360 THEN 80
120 LOCATE 5,12
130 INPUT "n*-カッション(0~31)=";P
140 IF P<0 OR P>31 THEN 80
150 '
160 CALL MK TEMPO (T,P)
170 '
180 K#=INKEY#
190 IF K#<>CHR#(&H20) THEN 180
200 CALL MK TEMPO (120,0)
210 END
```

プログラム例解説：

RUNさせるとテンポ値とパーカッションマップの入力を求めてきますので、数字キーを押して入力してから **RETURN** キーを押してください。メトロノーム機能が働いて、入力したテンポ値にしたがってリズム音が鳴り始めます。**SPACE** キーを押すとメトロノーム機能を止めてからプログラムが止まります。

CTRL + **STOP** を入力すると、プログラムは止まりますがリズム音は鳴り続けます。このとき、STOPM文を実行させてもリズム音は止まりません。あらためてMK TEMPO文またはAUDIO文を実行させてください。

60行のAUDIO文のパラメータの設定と、210行のEND文の前に200行でMK TEMPO文を<パーカッションマップ>を0に設定して実行させていることに注目してください。

■MK記録に関する命令

ミュージックキーボードによるインストゥルメントの演奏の記録に関する命令です。記録はミュージックキーボードから行われ、再生はインストゥルメントで行います。

CPUメモリのアドレスを直接指定することにより記録領域として使用する場合には、バックグラウンド（背景）で行うことができます。

RECMOD

(レコード・モード)

[ステートメント]

機能： MK記録の記録モードを設定します。

書式： CALL RECMOD (<記録モード>)

文例： CALL RECMOD (2)

解説： MK記録の記録と再生を同時にバックグラウンドで行うときに便利な命令です。初期値は1です。

<記録モード>は0から3で指定し、次のような意味を持ちます。

- 0： ミューティング (演奏はできますが、記録/再生はしません。)
- 1： ミュージックキーボードによる演奏を記録します。
- 2： MK記録の再生を別の領域に再び記録します。
- 3： MK演奏と再生を両方とも記録します。

参照： **AUDIO** (94 ページ) , **BGM** (97 ページ) , **REC MK** (165 ページ)
PLAY MK (168 ページ)

REC MK

(レコード・エムケー)

[ステートメント]

機能： インストルメントの演奏の記録を行います。

書式： CALL REC MK (<配列名>)

CALL REC MK (<開始アドレス>, <終了アドレス>)

文例： ミュージックキーボード (MK) の演奏を記録します。記録領域としては配列、またはメインメモリの中のユーザーリザーブ領域のアドレスを指定して使用することができます。

配列を記録領域として指定したときはバックグラウンドでの記録はできません。配列を指定してREC MK文を実行したときは、BGM文の設定にかかわらず、フォアグラウンドで処理されます。

<開始アドレス>と<終了アドレス>をパラメータとして指定したとき、記録はBGM文の指定によりバックグラウンドで行えますが、2つのアドレスの間の領域はシステムが使用することのない領域である必要があります。あらかじめBASICのCLEAR文を使用してユーザーリザーブ領域を設定し、使用してください。

MK記録には、新しく演奏を記録する機能の他にも、再生中の演奏の記録をあらためて別の領域に記録したり、古い演奏の記録を再生しながら新しくミュージックキーボードを演奏して両方の音をまとめて別の領域に記録する機能があります。これらの機能はREC MK文の実行の前に、RECMOD文により設定します。

記録を中止するには、BGM文の設定により2通りの方法があります。

CTRL + STOP を入力する。バックグラウンド
処理をしていない
とき

STOPM文を実行する。 バックグラウンド
処理中

STOPM文で停止したMK記録はCONT MK文により再開でき
ます。

記録されたデータをフロッピーディスクにセーブするには、

配列に記録したデータ COPY文によりフロッピ
ーディスク上のファイルに
転送

アドレスを指定して記録したデータ . . BSAVE文でセーブ

記録した方法により、上記のどちらかを行います。

参照： AUDIO (94 ページ) , BGM (97 ページ) , STOPM (99 ページ)
RECMOD (164 ページ) , PLAY MK (168 ページ)

プログラム例：

```

10 '
20 '
30 '   REC MK TEST
40 '
50 '
60 CLEAR 300, &HA000
70 CALL AUDIO (0,9)
80 CALL MK VOICE (@16)
90 CALL BGM (0)
100 '
110 CLS
120 LOCATE 8,10
130 PRINT "えんぞう スタート!"
140 '
150 CALL REC MK (&HA000, &HA2FF)

```



```
160 '
170 CLS
180 LOCATE 5,10
190 PRINT "えんぞうストップ* . セーフ"ちゅう."
200 BSAVE "CAS:RMTEST", &HA000, &HA2FF
210 '
220 CLS
230 END
```

プログラム解説:

このプログラムは、演奏のデータをカセットテープにセーブするように作ってあります。フロッピーディスクをお使いのかたは、200行のCAS:を削除するか、またはお使いになるドライブ名に書き換えてください。

記録が終わりしだいセーブしますので、あらかじめカセットテープ（データレコーダ）またはフロッピーディスク（ドライブ）を記録可能な状態にしておいてください。

RUNさせると演奏を始めるよう求めてきますので、ミュージックキーボードを弾いてください。画面の表示が変わったら、今の演奏のデータがカセットテープ（またはフロッピーディスク）にセーブされます。

PLAY MK

(プレイ・エムケー)

[ステートメント]

機能： MK記録の再生を行います。

書式： CALL PLAY MK (<配列名>)

CALL PLAY MK (<開始アドレス>, <終了アドレス>)

CALL PLAY MK

文例： CALL PLAY MK (A)

(配列Aは事前にDIM文で宣言され、REC MK文により記録されていること)

解説： インstrumentの演奏を再生します。

パラメータが配列名のときはバックグラウンド処理はできません。

配列をパラメータとして指定したときは、BGM文の設定にかかわらず、バックグラウンド処理をせずに再生します。

パラメータがアドレスのときは、BGM文の設定により、バックグラウンド処理を行うことができます。

バックグラウンド処理を行う場合、記録と再生は同時に行うことができ、記録する対象をRECMOD文の指定によって切り換えることができます。

パラメータがない場合は、最後に記録したものを再生します。

再生を中止するには、

CTRL + STOP を入力する・・・バックグラウンド処理をしていないとき

[PLAY MK]

STOPM文を実行する バックグラウンド処理中

STOPM文により中止したMK再生は、後述のCONT MK文により再開できます。

参照: AUDIO (94 ページ), BGM (97 ページ), RECMOD (164 ページ)
REC MK (165 ページ), STOPM (99 ページ)

プログラム例:

```
10  '  
20  '  
30  '      PLAY MK TEST  
40  '  
50  '  
60  CLEAR 300, &HA000  
70  CALL AUDIO (0, 9)  
80  CALL MK VOICE (@16)  
90  CALL BGM (0)  
100 '  
110 CLS  
120 LOCATE 10, 10  
130 PRINT "ロート"ちゅう."  
140 '  
150 BLOAD "CAS:RMTEST"  
160 '  
170 CLS  
180 LOCATE 8, 10  
190 PRINT "さいせいスタート!"  
200 '  
210 CALL PLAY MK (&HA000, &HA2FF)  
220 '  
230 CLS  
240 END
```

プログラム例解説：

このプログラムは、REC MK文のプログラム例を使ってセーブした演奏の記録を再生するようになっています。まだREC MK文のプログラム例を入力していないかたは、先にそちらのプログラムを入力して動かしてからこのプログラムを動かすようにしてください。

このプログラムは、カセットテープにセーブされた演奏の記録を自動的にロードするようになっていますので、フロッピーディスクをお使いのかたは150行のCAS:を削除またはお使いのドライブ名に書き換えてください。また、このプログラムを動かす前には、カセットテープ（データレコーダ）またはフロッピーディスク（ドライブ）をロード可能な状態にしておいてください。

RUNさせるとREC MK文のプログラム例を使ってセーブした演奏の記録をロードして再生します。

CONT MK

(コンティニュー・エムケー) [ステートメント]

機能： STOPM文により中止したMK記録の記録／再生を再開します。

書式： CALL CONT MK

文例： CALL CONT MK

解説： STOPM文により中止したMK記録の記録／再生を再開します。

参照： AUDIO (94 ページ) , STOPM (99 ページ) , REC MK (165 ページ) , PLAY MK (168 ページ)

プログラム例：

```

10  '
20  '
30  '      CONT MK TEST
40  '
50  '
60  CLEAR300,&HA000
70  CALL AUDIO (0,9)
80  CALL MK VOICE (@16)
90  CALL REC MK (&HA000,&HCFFF)
100 A$="きろくちゅう。"
110 CLS
120 LOCATE 5,10
130 PRINT "1:つづ"き!", "2:さいせい"
140 LOCATE 5,12
150 PRINT "3:たいき ", "0:おわり!"
160 LOCATE 5,14
170 PRINT A$
180 K$=INKEY$
190 IF K$="1" THEN 360

```

```

200 IF K$="2" THEN 320
210 IF K$="3" THEN 280
220 IF K$="0" THEN 250
230 GOTO 180
240 '
250 CLS
260 END
270 '
280 CALL STOPM
290 A$="たいきちゅう。"
300 GOTO 120
310 '
320 CALL PLAY MK
330 A$="さいせいちゅう。"
340 GOTO 120
350 '
360 CALL CONT MK
370 A$="そっこうちゅう。"
380 GOTO 120

```

プログラム例解説：

RUNさせると画面に4つの数字とそのキーを押したときの動作が、少し下に現在の動作（「きろくちゅう」）が表示されます。

このとき、すでに演奏の記録が始まっていますので、すぐにミュージックキーボードで何か弾いてください。少し演奏したら **3** キーを押してください。

画面に「たいきちゅう」と表示されたら、**1** キーを押してから、先ほどの演奏と区別しやすくするために演奏する音の音域を1オクターブ変えるなどの工夫をして演奏を再開してください。演奏が終わったら再び **3** キーを押してください。

次に **2** キーを押してください。画面に「さいせいちゅう」と表示され、最初の演奏といまの演奏が続けて再生されます。

再生の途中で **3** キーを押して「たいきちゅう」にしてください。

再び **1** キーを押すと、今度は残りの記録が再生されます。

[3] キーでSTOPM文を、**[1]** キーでCONT MK文を実行させています。CONT MK文が直前のSTOPM文を実行したときの状態を再開することをお確かめください。

プログラムの実行を終えるときは、どの動作をしているときでも必ず、

[3] キーを押して画面に「たいきちゅう」と表示させてから、**[0]** キーを押すようにしてください。

APPEND MK

(アベンド・エムケー) [ステートメント]

機能： MK記録の追加記録を行います。

書式： CALL APPEND MK (<配列名>)

CALL APPEND MK (<開始アドレス>, <終了アドレス>)

文例： CALL APPEND MK (A)

(配列AはすでにDIM文で宣言され、REC MK文により一部分記録されていること。)

解説： 記録領域の中の終了マークをさがし、その場所からMK記録を続けます。

参照： AUDIO (94 ページ), REC MK (165 ページ), CONT MK (171 ページ)

プログラム例：

```

10 '
20 '
30 '      APPEND MK TEST
40 '
50 '
60 CLEAR 300, &HA000
70 CALL AUDIO (0, 9)
80 CALL MK VOICE (@16)
90 '
100 CLS
110 LOCATE 10, 10
120 PRINT "ちゅうし： 5"
```



```
130 LOCATE 10,12
140 PRINT "つづ"き: R"
150 '
160 CALL REC MK (&HA000,&HCFFF)
170 '
180 K$=INKEY$
190 IF K$="S" THEN GOSUB 230
200 IF K$="R" THEN GOSUB 270
210 GOTO 160
220 '
230 CALL STOPM
240 CALL PLAY MK
250 RETURN
260 '
270 CALL APPEND MK (&HA000,&HCFFF)
280 RETURN
```

プログラム例解説:

RUNさせると記録が始まります。ミュージックキーボードで何か演奏をしてから、**[S]** キーを押してください。いまの演奏が再生されます。再生が終わったら**[R]** キーを押してください。再び記録が始まりますから、違った音域でもう一度演奏してください。演奏が終わったら**[S]** キーを押してください。最初の演奏と次の演奏が続けて再生されます。プログラムの実行を止めるには、**[CTRL]+[STOP]** を入力してください。

MK STAT

(エムケー・ステイタス)

[関数]

機能： MK記録/再生の状態を知らせます。

書式： CALL MK STAT (<変数名>)

文例： CALL MK STAT (A) : PRINT A

解説： 現在のMK記録/再生の状態を調べ、10進数にして変数に代入する関数です。

(変数の値) = 128 × (MKにFM音源の音色がセットされている。)
 + 16 × (ADPCMの音色がセットされている。)
 + 8 × (MK再生中)
 + 4 × (MK記録中)
 + 2 × (RECMODが2または3に設定されている)
 + 1 × (RECMODが1または3に設定されている)

例えば、

137 = 128 + 8 + 1 : (FM音源の音色がセットされ、RECMODが1に設定されていて、MK再生中)

23 = 16 + 4 + 2 + 1 : (ADPCM音声ファイルがMKの音色としてセットされ、RECMODが3に設定されていて、MK記録中)

参照： AUDIO (94 ページ) , RECMOD (164 ページ) , REC MK (165 ページ) , PLAY MK (168 ページ)

プログラム例：

```
10  '
20  '
30  '      MK STAT TEST
40  '

```

```
50 '
60 CLEAR 300,&HA000
70 CALL AUDIO (0,9)
80 CALL BGM (1)
90 '
100 CLS
110 PRINT "きろくちゅう。"
120 CALL REC MK (&HA000,&HA200)
130 CALL MK STAT (A)
140 IF A<>133 THEN 190
150 GOTO 130
160 '
170 CLS
180 PRINT "さいせいちゅう。"
190 CALL PLAY MK (&HA000,&HA200)
200 CALL MK STAT (A)
210 IF A<>137 THEN 200
220 GOTO 240
230 CLS
240 PRINT "さいせいしゅうりょう!"
250 END
```

プログラム例解説：

RUNさせると画面に「きろくちゅう」と表示されますので、ミュージックキーボードを演奏してください。指定された記録領域を使い果たして記録が終了すると、画面に「さいせいちゅう」と表示が出て再生が始まります。再生が終わると、画面に「さいせい しゅうりょう」と表示してプログラムの実行を終了します。

バックグラウンド処理中でも、130行と200行のMK STAT文でそれぞれ記録と再生の状態を繰り返し調べ、変数Aの数値により終了したことを確かめてから次の命令の実行にかかっていることに注目してください。

■その他の命令

MSXオーディオユニットの拡張BASICには、この章で説明したものの以外にも次のような命令があります。これらの命令はMSXオーディオユニットのハードウェア（ICや回路など）を機械語により直接操作するためのものですので、一般のユーザーが使われる必要はありません。

APEEK MSX-Audioのメモリ内容を参照します。

書式： CALL APEEK (<アドレス>, <変数名>)

APOKE MSX-Audioのメモリ内容を変更します。

書式： CALL APOKE (<アドレス>, <データ>)

AUDREG MSX-Audioのレジスタに値を書き込みます。

書式： CALL AUDREG (<レジスタ番号>, <値>)

4. 命令の索引

命 令	機 能	参照頁
APEEK	システムメモリの参照	178
APOKE	システムメモリの変更	178
APPEND MK	MK記録の追加	174
AUDIO	MSXオーディオユニットの初期化	94
AUDREG	MSX-Audioのレジスタに値を書き込む	178
BGM	バックグラウンド処理の設定または解除	97
CONT MK	STOPM文中で中止したMK記録/再生の再開	171
CONVA	PCM形式のデータをADPCM形式のデータに変換	144
CONVP	ADPCM形式のデータをPCM形式のデータに変換	145
COPY PCM	ADPCM/PCMデータの転送	126
INMK	ミュージックキーボードの変化を報告	154
KEY ON/OFF	インストゥルメントにキー・オン/キー・オフを与える	157
LOAD PCM	ADPCM/PCM音声ファイルのディスクからのロード	141
MK PCM	インストゥルメントにADPCM音声ファイルを割り当てる	150
MK STAT	MK記録システムの状態を報告	176
MK TEMPO	MK記録/再生とメトロノーム機能の速度の設定	159
MK VEL	インストゥルメントのペロシティを設定	152
MK VOICE	インストゥルメントの音色を設定	146
MK VOL	インストゥルメントの音量を設定	148

命 令	機 能	参照頁
PCM FREQ	ADPCMのローカルモードでの再生中にサンプリング周波数を設定	131
PCM VOL	ADPCM/PCM再生の音量を設定	133
PITCH	FM音源の楽音の音高を設定	110
PLAY (命令)	音楽をMMLにしたがって演奏する	101
PLAY (関数)	PLAY文による音楽の演奏中かどうかを報告	104
PLAY MK	MK記録を再生	168
REC MK	インストルメントの演奏を記録	165
RECMOD	MK記録の記録モードを設定	164
REC PCM	ADPCM/PCMにより音声を録音	135
SAVE PCM	ADPCM/PCM音声ファイルをディスクにセーブ	138
SET PCM	ADPCM/PCMの音声ファイルを初期設定	122
STOPM	バックグラウンドで実行中のPLAY文による演奏、ADPCM、MK記録再生を停止させる	99
SYNTHE	内蔵ソフトを起動する	106
TEMPER	FM音源の音律を設定	118
TRANSPOSE	FM音源の楽音を移調	113
VOICE	FM音源のチャンネルに音色を設定	107
VOICE COPY	音色データを転送	116

5. ミュージック・マクロ・ランゲージ(MML)

ミュージック・マクロ・ランゲージ (MML) とは、BASICのPLAY文により音楽を演奏させるときに、発生する音の高さや長さ、音量などを指定するために使用する文字や記号のことで、楽譜で言えば音符や音楽記号にあたるものです。

実際に使用するときには、PLAY命令の後ろに「”」（ダブルクォーテーション）で囲んで記述します。同じチャンネルの音について指定する複数のMMLは、1組みの「”」の中にまとめて記入することができますので、この1組みの「”」で囲まれたMMLを特に「PLAY文用の文字列」と呼びます。

PLAY文用の文字列の実行に際しては、先頭のMMLから順に実行されます。複数の音源または複数のチャンネルを同時に操作する場合は、1つのPLAY文の中で命令の後ろに必要な個数のPLAY文用の文字列を「,」（カンマ）で区切って記述します。

例: PLAY#2, "V1506L1CDF", "V1506L1EFA",
"V1506L1GA>C"

ただし、プログラムの1行に書き込める文字数は命令や空白を含めて255文字以内に制限されていますので、各PLAY文用の文字列に書き込むMMLの個数は、この制限を越えないようにする必要があります。

この場合は、複数のPLAY文を使って演奏するようにしてください。

拡張BASICのMMLは、FM音源やPCM音源などの操作のため、標準のBASICのMMLにくらべて種類が増えています。

次頁に拡張BASICで使用するMMLの仕様を表にして掲げます。

●FM音源、PCM音源、PSG音源用MMLの仕様

文字	意味	値のとり範囲	初期値
Mn	エンベロープ周期の設定	$1 \leq n \leq 65535$	M255
Sn	エンベロープ形状の設定	$0 \leq n \leq 15$	S0
Vn	音量の設定	$0 \leq n \leq 15$	V8
Ln	長さの設定	$1 \leq n \leq 64$	L4
Qn	音の長さの割合	$1 \leq n \leq 8$	Q8
On	オクターブの設定	$1 \leq n \leq 8$	O8
>	オクターブを1つ上げる	-	-
<	オクターブを1つ下げる	-	-
Tn	テンポの設定	$32 \leq n \leq 255$	T120
Nn	nで指定された高さの音を発生する	$0 \leq n \leq 96$	-
Rn	休符の設定	$1 \leq n \leq 64$	R4
A~G	音程の発生	-	-
+, #	音を半音上げる	-	-
-	音を半音下げる	-	-
.	音符や休符の長さを1.5倍にする	-	-
(ピリオド)			
=x;	パラメータnを変数xで設定する	*	-
&	タイ。前後の音をつなぐ	-	-
{ } n	連符。n分音符を {} 中の音程の個数で等分にした音を発生する	$1 \leq n \leq 64$	Lnで設定された値
@n	n番の音色に切り替える	$0 \leq n \leq 63$	-
@Vn	音量を細かく設定する	$0 \leq n \leq 127$	-

5. ミュージック・マクロ・ランゲージ (MML)

文字	意味	値のとり範囲	初期値
@Wn	nで指定された長さだけ状態を継続する	$1 \leq n \leq 64$	L nで設定された値

* : 値のとり範囲は直前のMMLの種類によって決定されますが、値が32767を越えることは許されません。

●リズム音用MMLの仕様

リズム音の場合、1つのMMLで同時にいくつかの音を発生するため楽音用とは異なる記述形式をとります。まず鳴らしたい楽器を表す文字を並べてその後ろに長さを表す数字を付け加えます。

文字	意味	値のとり範囲	初期値
B	バスドラム音を発生	—	—
S	スネアドラム音を発生	—	—
M	タムタム音を発生	—	—
C	シンバル音を発生	—	—
H	ハイハット音を発生	—	—
!	直前の楽器の音量をアクセントボリュームにする	—	—
n	直前までに書かれた楽音を発生し、n分音符分待つ	$1 \leq n \leq 64$	—
Vn	アクセントの付いていない楽音の音量を設定する	$0 \leq n \leq 15$	8
@An	アクセントの付いている楽音の音量を設定する	$0 \leq n \leq 15$	—

5. ミュージック・マクロ・ランゲージ (MML)

Tn, @Vn, Rn, =x;, . については、FM音源用と同じです。

例: "BSH8H8S!H8H8"

- ・バスドラム、スネアドラム、ハイハットシンバルを鳴らし、8分音符分待ちます。
- ・ハイハットを鳴らし、8分音符分待ちます。
- ・スネアをアクセント付きでハイハットといっしょに鳴らし、8分音符分待ちます。
- ・ハイハットを鳴らし、8分音符分待ちます。

●MMLと各音源との対応

文字	FM音源	PCM音源	PSG音源
Mn	*1	*1	○
Sn	*1	*1	○
Vn	○	○	○
Ln	○	○	○
Qn	○	○	*1
On	○	○	○
>	○	○	○
<	○	○	○
Tn	○	○	○
Nn	○	○	○
Rn	○	○	○
A~G	○	○	○

5. ミュージック・マクロ・ランゲージ (MML)

文字	FM音源	PCM音源	PSG音源
+, #	○	○	○
-	○	○	○
.	○	○	○
=x;	○	○	○
&	○	○	○
{ } n	○	○	* 3
@n	○	○	* 1
@Vn	○	○	* 1
@Wn	○	○	* 2

* 1 : 無視されます。

* 2 : R nと同じ効果 (n分休符) を持ちます。

* 3 : P S G音源に対しては使用できません。使用するとエラーになります。

6. FM音源用音色データ一覧表

音色番号0～31のデータは常にROM内に置かれているため変更できませんが、32～63のデータはVOICE COPY文により変更することができます。一度電源スイッチを切ってから再び電源スイッチを入れると、変更したデータは元のデータに戻ります。

略号は音色の名前としてデータの先頭に書き込まれているものです。

(この表に示した音色名は参考のために付けたもので、音色によっては実際の楽器の音色と異なるものがあります。)

音色番号	音色名	略号
0	ピアノ 1	Piano 1
1	ピアノ 2	Piano 2
2	バイオリン	Violin
3	フルート 1	Flute
4	クラリネット	Clarinet
5	オーボエ	Oboe
6	トランペット	Trumpet
7	パイプオルガン 1	PipeOrgn
8	シロフォン	Xylophon
9	オルガン	Organ
10	ギター	Guitar
11	サンツール 1	Santool
12	電子ピアノ 1	Elecpian
13	クラビコード 1	Clavicod
14	ハーブシコード 1	Harpsicd
15	ハーブシコード 2	Harpscd2
16	ビブラフォン	Vibraphn
17	琴 1	Koto
18	太鼓	Taiko
19	エンジン 1	Engine
20	UFO	UFO
21	シンセサイザベル	SynBell

音色番号	音色名	略号
22	チャイム	Chime
23	シンセ・ベース	SynBass
24	シンセサイザー	Synthsiz
25	シンセ・ドラム	SynPercu
26	シンセ・リズム	SynRhyth
27	ハーモ・ドラム	HarmDrum
28	カウベル	Cowbell
29	ハイハット	CiseHiht
30	スネア・ドラム	SnareDrm
31	バス・ドラム	BassDrum
32	ピアノ 3	Piano 3
33	電子ピアノ 2	Elecpia2
34	サンツール 2	Santool2
35	ブラス	Brass
36	フルート 2	Flute 2
37	クラビコード 2	Clavicd2
38	クラビコード 3	Clavicd3
39	琴 2	Koto 2
40	パイプオルガン 2	PipeOrg2
41	P o h d s P L A	PohdsPLA
42	R o h d s P R A	RohdsPRA
43	チャーチオルガン L	Orch L

6. FM音源用音色データ一覧表

音色番号	音色名	略号
44	チャーチオルガンR	Orch R
45	シンセ・バイオリン	Synviol
46	シンセ・オルガン	SynOrgan
47	シンセ・ブラス	SynBrass
48	チューバ	Tube
49	三味線	Shamisen
50	マジカル	Magical
51	フワフ	Huwawa
52	ワンダーフラット	WnderFlt
53	ハードロック	Hardrock

音色番号	音色名	略号
54	マシーン	Machine
55	マシーン V	MachineV
56	コミック	Comic
57	SE-コミック	SE-Comic
58	SE-レーザー	SE-Laser
59	SE-ノイズ	SE-Noise
60	SE-星 1	SE-Star
61	SE-星 2	SE-Star2
62	エンジン 2	Engine 2
63	無音	Silence

7. ROM音声ファイル一覧表

ADPCM音源用の音声データとして次の表のものがROM内に用意されています。これらのデータはCOPY PCM文により一度他の音声ファイルに転送してから再生します。

ファイル番号	内容	長さ (256バイト単位)
0	カッコー	35
1	ニワトリ	33
2	猫	20
3	犬	5
4	馬	29
5	ライオン	43
6	人の笑い声	26
7	ドアの閉まる音	11
8	ウイスキーを注ぐ音	4
9	靴音	1
10	行進	6
11	玩具	4
12	拍手 (ハンド・クラップ)	2
13	テニス	5
14	ゴルフのスイング	9
15	ゴルフのカップイン	9
16	刀を振る音 1	6
17	刀を振る音 2	7

- ♪ ROM音声ファイルに収められている音声は、どれも1回分だけの音声ですので、音声によっては繰り返して再生しないとそれらしく聞こえないものがあります。

第4章

資料編

1. 用語集

■ F S - C A 1 特有の用語

■ コンピュータ用語

■ 音楽用語

2. 仕様

3. 故障かな！？と思われたときは

4. お手入れのしかた

5. アフターサービス

1. 用語について

■MSXオーディオユニット FS-CA1特有の用語

MSX-Audio MSXパソコンのために開発された新しい音源IC。9チャンネルの2オペレータFM音源と1チャンネルの4ビットFM音源を持ちます。FS-CA1ではCPUの管理するメモリとは別に、256キロビットまでの外部メモリと呼ぶ専用のメモリを独自に管理するため、ADPCMの録音/再生などをCPUの負担なしに行うことができます。

MSXミュージックシステム MSX-Audioを使って、誰でも気軽に音楽の演奏を楽しむように作られたFMシンセサイザーソフト。MSXオーディオユニットに内蔵されているため、「内蔵ソフト」とも呼びます。

MSX-Audioの拡張BASIC MSX-Audioの高度な機能を自由に使いこなしたい人のために作られた拡張BASIC命令。ADPCM音源を使った「サンプリング録音」が可能です。

FM音源のチャンネル 内蔵のFM音源は最大9種類の異なった音を同時に発生することができます。この機能を有効に使用するため、9つの異なったチャンネルがあるものとみなして、それぞれのチャンネルごとに異なった操作をすることができます。

内蔵ソフト用音色データ MSXミュージックシステムで使用するために用意された65種類のFM音源用の音色データ。MSXミュージックシステムの各演奏パートごとに別々に割り当てることができます。

1. 用語について

FM音源用音色データ	拡張BASICで使用するために用意された63種類のFM音源用の音色データ。内蔵ソフト用音色データとは内容が異なります。
ROM音声ファイル	拡張BASICでADPCM音源を使用する際に利用できるように用意された18種類のADPCM音声データ。
外部メモリ	MSX-Audioによって直接に管理されるメモリ。FS-CA1には256キロビットのRAMが外部メモリとして内蔵されています。
バックグラウンド処理	MSX-Audioに、CPUが行っている動作とは別の動作を並行して行わせること。
インストゥルメント	拡張BASICでは、ミュージックキーボードに割り当てられたFM音源のチャンネルをひとつの楽器として扱い、インストゥルメントと呼びます。
演奏データの記録／再生	ミュージックキーボードのキーの操作を読み取ってデータに変換したものを演奏データと呼びます。内蔵ソフト、拡張BASICともに演奏データをメモリに記録し、インストゥルメントを使って再生する機能があります。記録された演奏データはフロッピーディスクやカセットテープに保存したり呼び出したりすることができます。

■コンピュータ関係の用語

- CPU
(シーピーユー)
- 中央処理装置。コンピュータにとって、自動車であればエンジンにあたる大事な装置。メモリからデータを読み取って処理（加工）し、送り出すのが仕事です。パソコンの場合、1個のIC（複雑な回路を小さなパッケージの中に詰め込んだもの）のかたちで内部に収められています。
- メモリ
- 記憶装置。CPUが処理するデータやプログラムをたくわえておく装置。読み出し専用のROMと読み出しも書き込みもできるRAMに分けられます。パソコンの場合、多数のICのかたちで内部に収められています。MSX-AudioはCPUのメモリとは別に、自分専用のメモリを持っており、これを特に外部メモリと呼びます。
- ROM
(ロム)
- 読み出し専用の記憶装置。内部にはあらかじめプログラムやデータが収められており、内容を消したり書き換えたりできないようになっています。電源を切っても内容はそのまま残ります。
- RAM
(ラム)
- 書き込みが可能な記憶装置。内部に書き込まれたデータやプログラムは、電源を切ると消えてしまいます。
- アドレス
- CPUがたくさんデータやプログラムを収めるメモリの中から特定のデータや命令だけを読み出せるようにするため、メモリの中は無数の番地に分けられています。この番地のことをアドレスと呼びます。パソコンの場合、アドレスは1バイト単位です。MSX-Audioでは256バイト単位でアドレスを指定することがあります。

1. 用語について

バイト	操作するうえでのデータや命令の最少限度の大きさを示す単位。1バイトは8ビット。
ビット	コンピュータにとって情報の基本になる単位。0か1かどちらかの情報をあらわします。
BASIC (ベーシック)	初心者がコンピュータのプログラムを作成しやすい、ようにするために作られた入門用の言語。最も理解しやすいプログラミング言語のひとつです。
FM音源	FMはフリークエンシー・モジュレーションの略で周波数変調方式により音声信号を発生する回路のこと。PSG音源にくらべて周波数帯域が広くとれるため、より広い音域とより正確な波形の音を発生できます。
ADPCM	PCMはパルス・コード・モジュレーションの略でADPCMは適応差分PCMのこと。サンプリング周波数の1サイクルごとに音声信号を1つのコード(符号)を基にして発生する音源。マイクなどを通じて入力された音声信号をコード化するサンプリング機能があります。

■音楽関係の用語

- シンセサイザー ミュージックシンセサイザーのこと。主に楽器の音色を合成する装置です。
- チューニング 調律。または各種の楽器のピッチ（音高）を合わせること。A音が440Hzになるようにするのが標準ですが、合奏の場合は最も調律しにくい楽器のA音に合わせるのが普通です。
- トランスポーズ 移調。
- サスティン 余韻。音響用語としては音が持続している状態のことです。
- ルート 根音。コードの一番下の音で、コード名の最初の文字により指示されています。
（例： G7のG音）
- トニック・コード 主和音。主音（音階の第一音）の上にてできる和音で、その調の中心になる機能を持っています。
- アルペジオ奏法 コードの各音を一齐にではなく順に演奏する奏法。ハープの弦をかき鳴らしたときのように各音の間に音の切れ目を付けない奏法と、1音1音を区切って演奏する分散和音的奏法があります。
- パーカッション 打楽器。

2. 仕様

項目		内容
接続可能パソコン		MSX1 (RAM32KB以上) MSX2パソコン ※1)
内蔵メモリ		ROM 128KB SRAM 4KB 外部RAM 256Kビット
音源部	音源用LSI	Y-8950
	FM音源	9チャンネル
	同時発音数	FM音源 9音 PCM音源 1音
音声取り込み	方式	PCM、ADPCM
	取り込み可能時間	約8秒 (ADPCM、サンプリング 周波数8KHz時)
内蔵ソフトウェア	FMシンセサイザーソフト	内蔵音色データ 65音 内蔵リズムパターン 19パターン 自動伴奏機能あり
	拡張BASIC	内蔵FM音源用音色データ 63音 内蔵ADPCM音源用ROM音声 ファイル 18音 音声サンプリング機能あり
接続端子	音声出力端子	2個 (モノラル・同一信号出力)
	ミュージック キーボード端子	1個 (20ピンコネクタ用)
	マイク入力端子	1個 (M6プラグ用)
オーディオ出力	出力レベル	-15dBm [内蔵ソフト FL 440Hz (O3のA) 出力時]
	出力インピーダンス	約1K Ω
マイク入力 (使用可能マイク)	マイク感度	-70~-76dB (0dB=1V/ μ bar)

※1) ただし、スロットの位置や形状によっては、実用的には使用できないものも
あります。

例： FS-5500F1/F2、FS-5000F2、CF-3300、CF-3000など。

2. 仕様

項 目		内 容
寸法 / 重量	寸 法	巾 : 354mm 奥行き : 78mm 高さ : 176mm
	重 量	570g
消費電流	+ 5V	約130mA
	+12V	12mA
	-12V	8mA
使用条件	温 度	5~35℃
	湿 度	20~80% (ただし結露しないこと)

3. 故障かな！？と思われたときは

症 状	原 因	処 置
内蔵ソフトが起動しない。	内蔵ソフト切換えスイッチが「切」になっている。	電源スイッチを切ってから、内蔵ソフト切換えスイッチを「入」にし、電源スイッチを入れる。
	パソコンのRAM容量が不足している。	メインRAM32キロバイト未満のMSXパソコンでは、RAMを拡張してください。
音が出ない。	ケーブル接続不完全	音声ケーブルの接続を確認する。 ・パソコンからテレビへ ・FS-CA1からテレビまたはアンプへ ・アンプからスピーカへ
	テレビまたはアンプの音量が小さ過ぎる。	テレビまたはアンプの音量を大きくする。
拡張BASICの命令が使えない。	AUDIO文を実行していない。	AUDIO文を実行する。
PSGの音が出ない。	パソコンの音声出力端子をテレビまたはアンプに接続していない。	パソコンの音声出力端子をテレビまたはアンプに接続する。 (FS-CA1の音声出力端子には、PSGの音は出力されません。)

3. 故障かな！？と思われたときは

症 状	原 因	処 置
ミュージックキーボードを弾いても音が出ない。	ケーブル接続不完全	キーボードからのケーブルの接続を確認する。
	AUDIO文を実行していない。	AUDIO文を実行する。
キーボードの左側のキーだけ音が出ない。	内蔵ソフトがENSEMモードになっている。	エディット画面でKEY-MODEをNORMALかSPLITに設定する。 {ENSEMモードではベース部は自動伴奏のため、マニュアルで演奏できません。}
マイク入力ができない。	ケーブル接続不完全	マイクからのケーブルの接続を確認する。
	マイクの電池が切れている。	電池を内蔵しているマイクの場合、電池を交換してみる。
	内蔵ソフトを使用中	拡張BASICでプログラムを作成する。 {内蔵ソフトでは、マイクは使用できません。}
パソコンからのキー入力ができない。	PAUSEキーを押した。	もう一度PAUSEキーを押す。
パソコンのキーを押しても音が出ない。	BASIC使用時	BASIC使用時には、パソコンのキーでは演奏できません。

3. 故障かな！？と思われたときは

症 状	原 因	処 置
カセットテープのロード・セーブができない。	ケーブル接続不完全	オーディオカセットケーブルの接続を確認する。
	テープレコーダの調整不良	<ul style="list-style-type: none"> ●音量、音質のつまみを調整する。 ●電池を電源にしているテープレコーダの場合、電池を交換してみる。 ●テープレコーダにフェイズ（位相）切換えスイッチが付いている場合は、このスイッチを切り換えてみる。
	設置場所が不適当	テープレコーダをテレビよりできるだけ離して使用する（30cm以上）
	カセットテープの不良	カセットテープを交換してみる。
フロッピーディスクのロード・セーブができない。	接続不完全	パソコンとフロッピーディスクドライブの接続を確認する。
	フォーマットしていないフロッピーディスクを使用	フロッピーディスクをフォーマットする。

3. 故障かな！？と思われたときは

症 状	原 因	処 置
フロッピーディスクのロード・セーブができない。 (つづき)	フロッピーディスクのライトプロテクトタブが書き込み禁止の位置にある。 (セーブのみできない場合)	ライトプロテクトタブを書き込み可能な位置に動かす。
	フロッピーディスクが違ふ。 ファイル名が違ふ。 (ロードのみできない場合)	FILE S命令を使って、フロッピーディスクに呼び出したいファイルがセーブされているかどうか確認する。
	フロッピーディスクの不良	フロッピーディスクを交換してみる。
エラーメッセージが表示された。	操作の誤り。 接続不完全 e t c .	内蔵ソフトまたはBASIC、DISK BASICのエラーメッセージ一覧表を参照する。