

MIDI音源利用のための汎用入出力インタフェース

5Q-1

森川 治、竹内 晴彦
製品科学研究所

1. はじめに

人間どうしの対話では対話内容(言語情報)だけでなく、身振り手振り、つぶやき、間合い等の非言語情報も有効に利用してスムーズな対話を実現していると考えられる。

これらの非言語情報の多くは、たとえ単独では明確な情報を伝えないものであっても、同時に提示された言語情報を補強する役割を担っていると考えられる。つまり、人間は多くの情報チャンネルから複数の協調し合った情報が与えられることに慣れており、これらを総合評価して情報授受を行なう能力に優れていると考えられる。

しかし、現在のマン・マシン・インタフェースにおいては、単一の情報チャンネルからあいまい性の低い情報を授受するといった手法が主流になっており、これらの非言語情報を積極的に対話に利用している例はあまり無い。このことは、言語情報に非言語情報を付加して提示することにより、人間の心理的負荷が減少する可能性を示唆している。

本報告では機械側からの音による情報提示に着目し、これに関連した種々の実験を行なうのに必要な環境を整えつつあるので、それについて報告する。

2. 機械の発する音情報

機械が発する音情報には、合成音声等による言語的な情報と警告用のブザー等のような非言語的な情報がある。また非言語的な情報にはブザーのように積極的にアプリケーションプログラムが発する音だけでなく、ディスクアクセスの音のような処理に伴う副次的な音情報も含まれる。つまり、ディスクアクセスの音はユーザに、「コンピュータは今仕事をしている」という非言語情報を発していることになる。しかし、このような非言語情報を積極的にマン・マシン・インタフェースに活用するには、次のような問題を解決しなければならない。

1、現在普及している端末・パソコンでは音を出力するハードウェアが貧弱である。

2、たとえ端末・パソコンに音出力の機構があったとしても、それらの機能は機種間で互換性がなく、互換性のある部分はマン・マシン・インタフェースとして利用するには能力不足である。

3、音によるメッセージには時間軸に関する制御を必要とするので、音情報の制御方法や記述方法が複雑になる。

4、音を出しながら他の処理をするといった並列処理機構が基本的に必要である。

特に第4の問題点は、安価なパソコン等ではシングルタスクのOSが多いので考慮する必要がある。

3. MIDI音源による音情報

Using MIDI in Man Machine System

Osamu MORIKAWA & Haruhiko TAKEUCHI

Industrial Products Research Institute, MITI.

問題点1、2を解決する為に我々は、音出力のハードウェアとしてMIDI規格に合致した音源を使用することにした。採用理由は、MIDI規格は電子楽器に使用される規格なので、FM音源やPCM音源等多くの音源がMIDIデータで操作することが可能な点、しかもMIDI規格に準拠した音源が安価に入手可能な点、さらに音源を端末・パソコン外部に持たせることにより、使用機種に制限が少なくなる点である。

逆に欠点としては、使用する音源によって、表現できる内容が異なり、マン・マシン・インタフェースに利用できる内容に差が出てしまう点がある。しかし、MIDI規格に準拠した音源であればメーカを問わず接続できるので、この欠点は現時点ではあまり問題が無いと思われる。それよりも、MIDI音源を利用することにより、出力できる音の種類が飛躍的に増大するメリットの方が大きいと考える。

4. 拡張MIDI

MIDI音源を採用したが、MIDIデータには時間軸に関する制御情報が含まれていないため問題点3の解決にはならない。そこで、MIDIデータにタイミング情報を付加した「拡張MIDIデータ」を定義して使うことにした。「拡張MIDI」は、ローランド社のMPU-401が採用している方式を採用した。具体的にはMIDIデータ間にタイミングデータを挿入したものである。時間分解能は可変であるが標準で5msecである。例えば、チャンネル0のドの音(音高3C番)を21の強さで鳴らし、14クロック(標準で70msec)後に、レの音(音高3E番)を2Aの強さで鳴らす場合、音出力を表わす2つのMIDIデータの間にOEを挿入した

90, 3C, 21	MIDI
OE	timing
90, 3E, 2A	MIDI

の7バイトで表現する。なお、タイミングデータとして使用できる値は0-EF(最大239クロック、標準で1.195秒)であるが、それ以上の時間間隔の場合は時間間隔240クロックを表わすF8コードを必要に応じて先頭に追加挿入することにより表現する。

5. 拡張MIDIインターフェース

拡張MIDIデータを受け取りタイミングを取りながらMIDIデータを出力する、あるいはMIDIデータを受け取りタイミング情報を付加して拡張MIDIデータに加工するインターフェースを個々の端末・パソコン上に作成する必要がある。この作成には第4の問題点を考慮する必要がある。我々は、NEC社のパソコンPC98シリーズ上のシングルタスクOS(MS-DOS)の上で動作する、拡張MIDIデータ用の汎用入出力インタフェースを作成した。使用したハードウェアはローランド社のMPU-PC98である。

本インターフェースがサポートするサービス内容は表

1の通りである。アプリケーションプログラムは、PC 98シリーズのBIOSやMS-DOSのシステムコールと同様な手順（機能番号をAXレジスタにセットし、int 1Chを実行）で、これらのサービスを受けることができる。

これにより音出力の時間軸制御は本インターフェースが行なうことになり、シングルタスクのOSでありながら、音を出しながら他の処理を同時に行なうといった事が可能になる。またインターフェースをアプリケーションプログラムと分離したことにより、音出力として本方式を採用したプログラムの他機種への移植が可能になる。

6. 使用例

いろいろな応用が考えられるが、ここでは入力促進時に音出力を発する例を考える。ここでは選択子の個数を和音で、デフォルトとなる選択子を音の強弱で、選択内容のおおまかな分野を音色とリズムにより表示する。これにより、言語情報で表現すると例えば「文書印刷に関する入力促進であり、そこには選択子が3つあり、1番の選択子がデフォルトとなっていて、それほど処理の本質に影響のない選択である」や「エラーが発生したため、現在ある文書を保存して終了するか保存しないかを判断する入力促進であり、保存を選択するのがデフォルトとなっていて、重大な判断を要求している」といった情報が一瞬に伝えられる。C言語のプログラムでは、

```
m_put(m_msg1);
printf("処理を継続しますか (y/n) <");
scanf("%d",&ans);
```

のように記述できる。この例では m_msg1 という文字列定数に拡張MIDIで記述された音データがあるとする。

もちろん、これ以外にも表現方法は考えられる。また、どの内容をどの音色、音程、リズム、音強に対応するかによってシステムの使用感に大きく影響する。最適な対応付けは研究課題である。

上の例では入力促進の音を毎回発するように述べたが、場合によってはユーザの入力が一定時間、例えば3秒間無かった場合にだけ音により入力促進をしたほうが良い場合もある。この様な場合でも、基本的には上のプログラム手法で十分であり、m_msg1内の拡張MIDIデータの先頭のタイミングデータを3秒間に相当する「F8、F8、78」という3バイトにし、入力が行なわれた直後に現在発生中の音を中止し、バッファ内の拡張MIDIデータをキャンセルする関数

```
m_stop();
```

を実行することにより実現できる。なお、関数m_put()、m_stop()は表1のサービスを組み合わせることにより容易に実現する。

7. 終わりに

使いやすいマン・マシン・インタフェースを実現するには、人間の特性に合った情報授受方式を採用する必要がある。そのためには、言語情報だけでなく、非言語情報もマン・マシン・インタフェースに積極的に活用してゆくことが必要と考える。多くのアプリケーションプログラムが音情報を有効に使用して、人間にとって使いやすいインタフェースを実現する事を望む。それと同時に、コンピュータが扱うのに適した音情報の標準入出力機能の標準化の研究にも期待したい。

参考文献

- [1] ローランド、MIDI Processing Unit MPU-401 テクニカル・リファレンス・マニュアル、1985
- [2] 青柳、小池：ICOTone-/dev/midi0、情処学33全大、5N-6(1986)
- [3] 森川 治：タイミング情報利用可能なOSの作成、情処学37全大、4H-2(1988)
- [4] 森川 治：MIDIボード用BIOSの作成、インターフェース、vol.15-11、pp267-280(1989)
- [5] 竹内、森川：計算機インタフェースにおける音情報の利用、情処学40全大(1990)

表1、サービス内容

割り込みベクタ番号： 1Ch
 レジスタの使用規則（原則）：
 AX=機能番号/戻り値；
 DL=trks; dl=05h --- trk0,2 指定
 DL=データ；
 DH=trk; dh=3 --- trk 3 指定
 DH=MPUコマンド
 DS;BS=buf; CX=len;バッファのアドレスと長さ

----- 機能番号 ----- 内容 -----

80番 (MPU基本コマンド関係)

midi_ver	=80FFh	； システム常駐確認 &
		； バージョン取得
midi_init	=8000h	； システム初期化
midi_play_start	=8001h	； 演奏開始
midi_play_stop	=8002h	； 演奏停止
midi_play_cont	=8003h	； 演奏再開
midi_punch_init	=8004h	； パンチイン準備
midi_punch_start	=8005h	； パンチイン開始
midi_rec_start	=8006h	； 録音開始
midi_rec_stop	=8007h	； 録音終了
midi_stat	=8008h	； 演奏・録音状態の確認
midi_mpu_ver	=8009h	； ハードのバージョン取得

81番 (バッファ関係)

midi_getc	=8100h	； 1バイト入力
midi_getcr	=8101h	；
midi_read	=8102h	； ブロック入力
midi_readr	=8103h	；
midi_putc	=8108h	； 1バイト出力
midi_putcr	=8109h	；
midi_write	=810Ah	； ブロック出力
midi_writer	=810Bh	；

★末尾がrのサービスはバッファに余裕が無い場合には可能なだけデータ授受を行い処理を終了する。

midi_set_buf	=8110h	； バッファの設定
midi_set_clr_buf	=8111h	； バッファの設定・初期化
midi_clr_buf	=8112h	； バッファ内容の初期化
midi_stat_buf	=8113h	； バッファの状態取得

82番 (MPUコマンド送信関係)

midi_cmd	=8200h	； MPU-401 へのコマンド送信
midi_dat	=8201h	； MPU-401 へのデータ送信