

Hyper MIDI Lesson はシーケンサの塊

ハイパーMIDIレッスンは「やさしいコンピュータ活用法」音楽之友社(定価3000円)の付録に付いていますが、私のホームページからも無料でダウンロードできます。(http://www.art.hyogo-u.ac.jp/hrsuzuki/Jpaper.html)

USBなどを使ったMIDIインターフェイスに対応する方法は少々ややこしいのですが、MACのシステムホルダーにあるApple MIDI Driverを捨てて代わりにOMS MIDI Manager DriverかFreeMIDI MM Driverを積み込みます。後はPatchBayで設定すればUSBポートもバッチリです。このあたりは後ほど詳しく説明しますが、大切なことはHyperCardのスタックにHyperMIDIというリソースが組み込まれていなければなりません。このリソースの概念はWindowsのユーザーにはなかなか理解し難いのですがMacの一番大きな特長となっています。

さて、前号でも述べましたようにハイパーMIDIレッスンにはクラス全体を対象とした**共有シーケンサ**と6名までの小集団(班)のための**班専用シーケンサ(インスタントシーケンサ)**そして一人ひとりのための**個別シーケンサ**が用意されています。さらに全校18クラスのためのスタックがあります。

単純に計算しても $42 \times 18 = 756$ 人分の個別シーケンサが有るわけで、班用のシーケンサも $7 \times 18 = 126$ 班分がそれぞれ独立しています。デスクオルガンの絵をクリックすると**共有シーケンサ**の演奏が開始されますが、後からピアノの絵をクリックすると全然違うタイミングでピアノ遅れて演奏されます。つまり、このシーケンサは同じMIDIデータを扱いますが、ポートごとに独立しているのです。極端な場合、ある班カードで共有ファイルを演奏しながら別の班カードに切り替えても演奏は途切れません。

ハイパーMIDIレッスンではMIDIデータが8ビットの文字列のようなデータであることからテキストファイルと同じようにコンパクトに扱えると言う点を利用しています。従ってシステム内ではMIDIデータはテキストファイルとして扱っています。このことで多くのシーケンサーが効率よく扱えるようになっていきます。

ハイパーカード

クラスカードはいわば教師のえんま帳です。MIEでは出席簿の機能まで持たせてありましたが、ハイパーMIDIレッスンにはその機能は未搭載です。しかし、もしそ

の機能を付加したければ画面上のどこか空いたスペースに小さなボタンを一つ張り付けて、そのボタンに出席カードの機能を持たせたスタックへのジャンプをスクリプト(記述)すればすむことです。兵庫県の揖保小学校の教師たちはこの小さなボタンをいろいろな所に張り付けて教材楽譜の提示や映像の提示を工夫していました。ハイパーカードの素晴らしいところは、カードとかスタックとか呼ぶユニットの中に記述された命令はプログラム全体のスクリプトを変更することなく単独に働かせることができるということです。CやC++などのプログラム言語は小さなエラーがあっても全体が動かなくなりますが、1987年秋に鮮烈なデビューをしたハイパーカードのハイパートークという言語は小さなオブジェクトの統合体で動きますので素人にも段階に応じたプログラムが作れます。Windows全盛の世の中に見えますがこのようなプログラム環境はまだVisualBasic以外にはWindowsでは実現していません。WindowsといえどもDOS環境であることには変わりはありませんので、あらゆる実行ファイルには.EXEの拡張子がついています。ハイパーカードでつくられたプログラムはデバッグ(エラー修正)も簡単ですし、一度実行してしまうと次回からは機械語レベルで動きますので.EXEの拡張子がついたのと同じでスピードに問題はありません。Macが他のコンピュータより優れている一つの特長がこのユーザーに解放されたプログラム環境です。エクセルやワードなどの既製のアプリケーションを買い続けるしかない今のコンピュータ文化が生まれる前は、自分の欲しいプログラムは自分で作るのが当たり前だったのです。この思想はAppleScriptというキーボードもマウスも使わずにコンピュータを簡単に制御するソフトに今も引き継がれています。Macが今後も生き残れるのはスケルトンボディーや発熱が少ない、などのハード的な面ではなく、このようなユーザー環境が保障されているからなのです。皮肉なことに私もほんの12年前まではMS-DOSでプログラムを自作してきました。その頃Macはプログラムは作る必要のないマシーンであると豪語していました。それが今では逆転して、Macだけが一般ユーザーに自分だけのプログラムを作れる環境を維持しているのです。そのハイパーカードもMIDIのためのスクリプトは予想もしていなかったらしく標準のスクリプトではMIDIファイルの再生演奏しかできませんでしたが、前号で紹介したEarLevel社のHyperMIDIというリソースをスタックに張り付けるだけであらゆるMIDI信号処理ができ

るようになったのでこのハイパーMIDIレッスンは誕生したのです。このHyperMIDIについてEarLevel社 <http://www.earlevel.com/> から英語の情報を入手できますが日本語の情報は <http://osaka.cool.ne.jp/hightide/hypermidi/index.html> から入手できます。私はアメリカで80ドルで買いましたがそんなに高いものではありません。最近ではMidiToolz(<http://members.aol.com/MidiToolz/midi/MidiTools.html>) という商品も販売されるようになりハイパーカードでなくてもDirector等のソフトもMIDIコントロールのオーダーメイドが可能になっています。

MIDI ファイルについて

ハイパーMIDIレッスンにはスタンダードMIDI ファイルを記録したり読み込んで再生する機能がありますから、音楽の授業や課外活動に使用するソースつまり教材楽曲を自作することができます。一般に市販されている音楽ソフト(みゅーじ郎やXGWorks等)で教材を自作する位のことでは今どきの音楽教師にとってはそれほど難しいことではないでしょう。市販されていない教材を標準MIDIファイル(以下SMF)で作る醍醐味は一度味わうとくせになるでしょう。自作できなくても市販の教材ファイルを購入すれば良いわけですが、ヤマハの教材ファイルにはファイルフォーマットの違うものが多いので気を付けて下さい。例えばESEQというモードで記録されたフロッピーディスクはSMFとして読めません。これを読めるようにするにはMDRシリーズなどの装置かQX3やEOS(SDX3000)のようなもので一度メモリーに落としてから改めてSMFモードで記録し直す必要があります。これが結構腹が立つのですが著作権の問題らしいので仕方ありません。この時注意が必要なのはHyperMIDIはバージョン2.0の段階ではFormat 1つまりマルチトラックに対応していないと言うことです。セーブする時にはFormat0で保存して下さい。Ver3xからは何でもOKです。ハイパーMIDIレッスンで記録するときは自動的にFormat0になります。演奏に自信のある教師ならハイパーMIDIレッスンにつないだキーボードで演奏してしましましょう。間違えた音や抜けた音の追加は簡単な音楽ソフトでも可能です。



【ファイル読み込み】というボタンをクリックすると、自分の指定のディレクトリーやフォルダーの一覧が示されます。このファイル読み込みという命令は次のようなスクリプトによって実行されています。

```
-- reads a MIDI file into the global variable theSeq
on mouseUp
  global theSeq
  if theSeq is empty then set the name of me to "ファイル読み込み"
  get filename("Midi")
  if it is not empty then
    go to this card
    set cursor to watch
    put empty into theSeq
    put hmMIDIfile("read",it,"msec") into theSeq
    set the name of me to the last word of it
    set the name of bg btn 8 of cd 2 to the last word of it
  -- check for errors
  if char 1 to 5 of theSeq is "Error" then
    answer theSeq with "OK"
    put empty into theSeq
  else
    --checkBufSize
  end if
end if
end mouseUp

-- this routine sees if theSeq will fit in the output buffer,
-- and, if not, create a larger output buffer by reopening
HyperMIDI
on checkBufSize
  global theSeq
  -- see if sequence will fit in buffer
  hmOpenMIDI "!"
  put line 3 of the result into bufSize
  hmWriteMIDI 1,theSeq,"count"
  put the result into requiredSize
  if bufSize < requiredSize then
    -- kill any playing sequence
    if hmUtility("getOutputCount",1) > 0
      then get hmUtility("killOutput",1)
    -- reopen will larger buffer
    hmOpenMIDI 400,requiredSize
    if the result is not empty then answer the result with "OK"
    hmPatcher "connect",cd fld "connections" of cd 1
  end if
end checkBufSize
```

全てのスタック上での使用できるようにする global 関数で宣言されたtheSeqというMIDIデータの入る作業領域が空ならば実行されるhmで始まる以下の命令がすべて意味の分かる英語で書かれたこのようなスクリプトをハイパートークと呼んでいます。BASICの心得のある人やその他のややこしいプログラム言語の経験のある方なら何と分かり易いシンプルな言語であるかと感心するでしょう。因みに前半はファイルネームやMIDIデータの処理で後半がバッファ管理です。