

Chapter 4

MIDI処理

佐近田展康

Maxは音楽用のプログラミング言語としてその歩みをスタートした。当時コンピューターで音楽を作ったり演奏する場合には、シンセサイザーやサンプラー、音源モジュールといった専用の外部デジタル機器をコンピューターからコントロールすることが主流であり、すでにMIDIはそのコントロールにおいて中心的な役割を果たしている世界標準の規格であった。つまりMSPがリリースされるまでの長い期間、音楽制作の面から見たMax最大の特徴は、MIDIを前提にして、それを自在に取り扱うプログラミング言語という点にあったのである。MSPの登場によってオーディオDSP処理機能が追加されても、長年に渡って培われたMIDI処理機能の充実ぶり、そして、豊富に用意された音楽プログラミング用オブジェクト群がMaxの大きな特徴であることに変わりはない。本章では、MIDIの基礎知識から始めて、MaxでのMIDIの取り扱い方、そしてMaxによって可能になる音楽プログラミングとはどういったものなのかを見ていきたい。

● Chapter 4を読むための基本セッティング

Chapter 4ではMIDIを扱った音楽プログラミングの例を解説していくため、読者の環境で最低限の次のMIDI機器を用意してほしい。

MIDIコントローラー(鍵盤型が望ましい)

MIDI音源モジュール

MIDIインターフェース

一般に市販されている鍵盤型のデジタル・シンセサイザーや電子ピアノなどは、内部に音源部分も持っているため、1台でMIDIコントローラーとMIDI音源モジュールの機能を兼ね備えている。この場合は、別々に用意する必要はない。ただし、鍵盤を押すと内部音源が発音してしまい、コンピューターからのコントロールと区別がつかなくなる

ので、シンセサイザーなどの設定でローカル・コントロールをオフしておく必要がある。

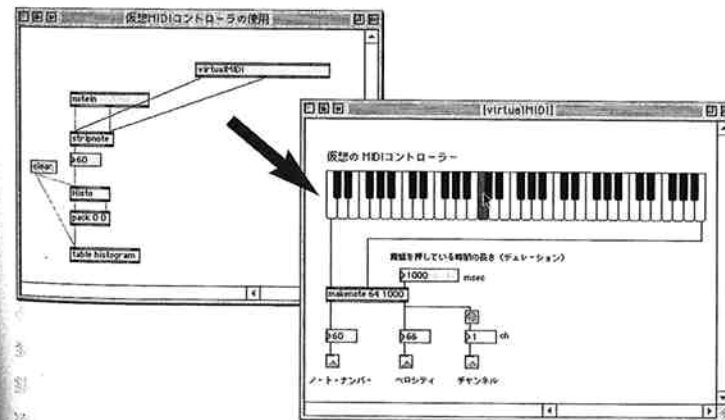
また、MIDI機器の中にはMIDIインターフェースを内蔵しているものもあり、コンピューターのUSBポートなどに直接できるようになっている。この場合にはMIDIインターフェースは必要ない。

次に『Chapter 2 周辺機器とインストール』に従って、コンピューターとMIDI機器の接続、OMSのインストールとセットアップを適切に行って準備してほしい。

なお、こうしたMIDI機器が用意できない場合でも、Chapter 4の大半のサンプル・パッチはサウンドを鳴らしながら実行することができる。

まず、MIDI音源モジュールがない場合は、コンピューターにAPPLEのQuickTimeおよびQuickTime Musical Instrumentsがインストールされていれば、そのソフトウェア音源(QuickTimeミュージック・シンセサイザ)を使ってコンピューター本体だけでサウンドを鳴らすことができる(この設定については362ページで解説している)。あるいは他の市販ソフトウェア音源や音源ボードを使用しても構わない。

MIDIコントローラーがない場合は、MIDI受信用のnoteinオブジェクトの代わりに、次のような仮想のMIDIコントローラー・パッチをサブ・パッチあるいはパッチ・オブジェクトとして作成しておけば、ある程度の代用は可能になるだろう。このパッチの個々のオブジェクトについては後述する。



▲仮想MIDIコントローラーを使用する

4-1 MIDIの基礎

MIDI (Musical Instrument Digital Interface) は、1983年に発表されたデジタル楽器のコントロールに関する共通規格である。当時さまざまなメーカーがデジタル楽器の開発と生産販売に力を注いでいたが、MIDIはこうしたメーカー間で結ばれた取り決めで、自社および各社の製品を互いに接続し、コントロールし合うことを目的として生まれたものである。その後、業界団体の組織が結成され、仕様の改訂、新たな規格の追加が行われるなど、楽器のみならずさまざまな音響機器や映像機器、舞台照明機器のコントロールなど広い分野で活用されながら現在に至っている。ここでは、そのMIDIについて、基本的な仕組みと考え方を紹介することにしよう。文中でも強調するように、MIDIでやり取りされるのは機器をコントロールするための命令であり、決してサウンドそのものではない点はくれぐれも念頭において読み進めてほしい。また、MIDIについての詳細資料は『Chapter 9 リファレンス・ガイド』のMIDI詳細資料にまとめているので、そちらも参照しながら読むことをお勧めする。

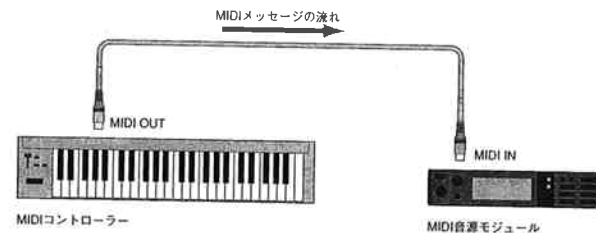
● ハードウェア規格としてのMIDI

MIDIには3つの側面がある。まず機器同士を接続するハードウェア設計規格の取り決めで、この規格に従う機器には通常IN、OUT、THRUの端子が装備され、MIDI OUT端子は他の機器のMIDI IN端子と5ピンのDINコネクタを利用したMIDIケーブルで接続される。メーカーや機種が違っても、これら端子やコネクタ、ケーブルは共通している。このMIDIケーブルを通して31.25kbpsの転送速度で1方向に伝えられるのがMIDIメッセージである。

例えば4-1-1のように接続された2つのMIDI機器があるでしょう。鍵盤型の機器は主にMIDIメッセージを発生させ、外部に送信するためのもので、一般にMIDIコントローラーと呼ばれる。鍵盤型以外に打楽器型、管楽器型、ギター型などさまざまな形がある。他方、ラック型の機器はMIDIメッセージを受けてさまざまな音色のサウンドを発生させるもので、一般にMIDI音源モジュールと呼ばれている。通常よく見かける鍵盤型のデジタル・シンセサイザーや電子ピアノは、内部に音源部分を持ち、1台で演奏が

可能になっているが、これはコントローラーと音源モジュールが一体になり、内部で接続されていると考えればよい。

■ 4-1-1 MIDI機器同士の接続例



● プロトコルとしてのMIDI

MIDIの第2の側面は、プロトコルとしてのMIDIメッセージに関する取り決めである。プロトコルとは簡単に言えば共通語であり、MIDIメッセージの場合もメッセージの形式と意味が定義され各メーカーで共有されている。例えば、MIDIコントローラーのどれか1つの鍵盤を押すと、MIDIコントローラーはノート・オンと呼ばれるMIDIメッセージを発生させ、MIDI OUT端子から送信する。MIDI音源モジュールはMIDI IN端子からこれを受け取り、何らかのサウンドを鳴らす。つまりノート・オン・メッセージとは“音を鳴らせ!”という意味を持った命令なのである。YAMAHAのMIDIコントローラーであろうと、ROLANDのMIDIコントローラーであろうと、送信されるノート・オンは同じであり、受信側のMIDI機器もメーカーに関わらずそれを“音を鳴らせ!”という命令として解釈する。

次に、鍵盤から指を離すとまた別のMIDIメッセージがMIDI音源モジュールに伝えられ鳴っていたサウンドが止まる。このメッセージはノート・オフと呼ばれ“音を止めろ!”という意味を持つ。このノート・オンとノート・オフはまとめてノート・メッセージとも呼ばれている。こうしてピアノやオルガンを弾くようにMIDIコントローラーを操作すると、演奏情報としてのノート・メッセージが伝えられ、MIDI音源モジュールが適切にサウンドを発生させて演奏を行うことができる。

鍵盤型のMIDIコントローラーは、鍵盤以外にボタンやホイールなどを持っており、フット・ペダルなども接続できるようになっているはずである。一般に、これら进行操作するとそれぞれ違った種類のMIDIメッセージが送信され、音源モジュール側の設定次第で、サウンドが変化して演奏に表情を付けることができる。

しかし、このように演奏ができると言っても、間違っはいけないのは、MIDIケーブルを通して送られているのが決してサウンドそのものではなく、あくまでも鍵盤を押したり、フット・ペダルを踏み込んだりといった演奏情報であり、言い換えれば“音を鳴らせ”“ボリュームを上げる”といった機器に対する命令に過ぎないという点だ。実際にどのようなサウンドが鳴るかは受信側の機器(この例ではMIDI音源モジュール)の設定で決まる。

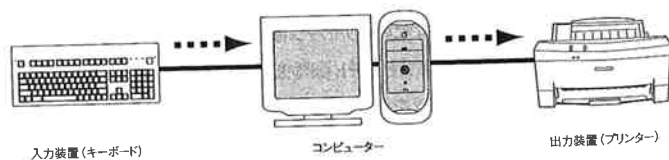
なお、MIDIの第3の側面は、MIDIメッセージをデータとして記録し保存するためのファイル・フォーマットに関する取り決めであるが、これについては後述しよう。

● コンピューターとMIDI

MIDIが生まれた最初の目的は、演奏者の手によってさまざまなデジタル楽器をリアルタイムに演奏(コントロール)するためであったが、MIDIをここまで広く普及させた最大の理由は、コントロールの主役が演奏者の手からコンピューターに移り、MIDIメッセージの記録、編集、生成、保存といった新しい可能性が広がったことにある。

コンピューターにはさまざまな種類の入力装置や出力装置が接続される。入力装置としては、キーボードやマウス、デジタル・カメラ、スキャナーといったものがすぐに思い浮かぶ。また出力装置としてはモニターやプリンターなどが代表的だろう。

■ 4-1-2 コンピューターと入出力装置



入力装置(キーボード)

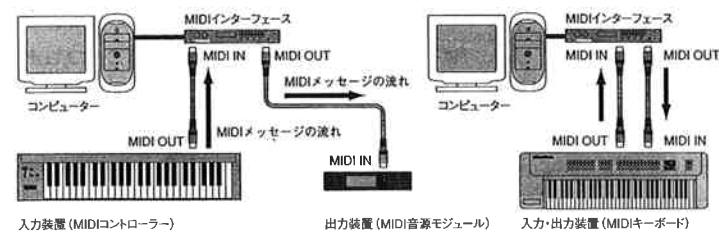
コンピューター

出力装置(プリンター)

MIDI機器がコンピューターと接続される場合も、この入力装置、出力装置として考えることができる。先ほどのMIDIコントローラーは入力装置、MIDI音源モジュールは出力装置になるだろう。音源を内蔵した鍵盤型のキーボードは入力装置であると同時に出力装置にもなる。ただしコンピューターとMIDI機器を接続する際には通常MIDIインターフェースという外部機器を経由しなければならない。

05/3/1

■ 4-1-3 入出力装置としてのMIDI機器



入力装置(MIDIコントローラー)

出力装置(MIDI音源モジュール)

入力・出力装置(MIDIキーボード)

ところで、4-1-2のようにキーボードとプリンターがコンピューターに接続されている状態で、例えばコンピューター・キーボードでAという文字をタイプしたとしよう。もしこれによってタイプライターのようにプリンターが反応し、すぐにAという文字を印刷したらどうなるだろうか？ 文字の打ち間違いを取り消すこともできないし、文章全体のレイアウトを自由に変えることもできないだろう。また1度打ち込んだ文章を保存しておいて、あとで一部分だけ修正するといったこともできず、結局、修正がある場合は文章全部をもう1度打ち直さなくてはならない。これではタイプライターと同じで、わざわざコンピューターを使用する意味はない。

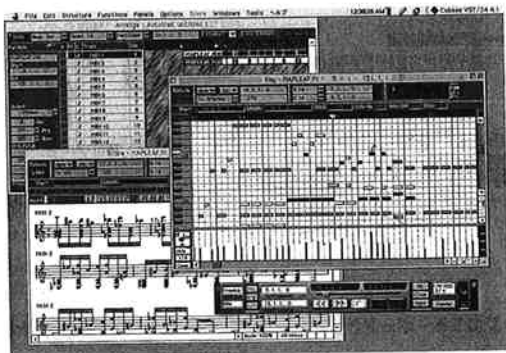
ワープロソフトと呼ばれているものは、キーボードで打ち込んだ文字データをすぐには印刷しないで、記録し、編集するためのものである。何度も気の済むまで書き換えて、レイアウトを再構成し、最終的に満足のいく文書にしてから印刷を行うことができる。またこのデータを保存し、いつでも再編集することができる。

MIDIとコンピューターの関係についても同じことが言える。MIDIコントローラーを使ってある曲を演奏したとしよう。その演奏情報がMIDIメッセージとしてコンピューターに入力され、記録される。これには演奏ミスがあるかもしれないし、メロディを変更し

たいと思うかもしれない。あとから別パートの演奏を追加したいと思うかもしれない。

このようにMIDI機器の中にコンピューターが加わることでMIDIメッセージを記録し、何度も気の済むまでMIDIメッセージを編集することが可能になったのである。これを可能にする音楽のワープロ・ソフトに当たるものが、シーケンス・ソフトと呼ばれる音楽ソフトウェアである。

■4-1-4 シーケンス・ソフト例(STEINBERG Cubase VST)



ワープロ・ソフトがコンピューターのウィンドウを白い紙に見立て、そこに文字を書くイメージで文書を作成するのに対し、シーケンス・ソフトは、ウィンドウを楽譜に見立て、そこに音符を書き込んで行く感覚で曲を作成していく。つまり横軸が時間の経過、縦軸が音の高さといった具合だ。STEINBERG Cubase VST、EMAGIC Logic Audio、MARK OF THE UNICORN Digital Performerなど多数の有名なシーケンス・ソフトが存在し、それぞれに独自のMIDI処理機能を充実させているが、この基本的な考え方はみな同じである。

● スタンダードMIDIファイルと同期

こうして作成された音楽データは、時間情報とMIDIメッセージをまとめて記録したものであり、MIDIシーケンス・データと呼ばれる。これをファイルとして保存する場合、上

に挙げたいいくつかのシーケンス・ソフトでは、それぞれ独自のフォーマットを持っており、互いに互換性がない。これではCubase VSTで作成した音楽はCubase VSTでなければ再現できないことになり、制作やデータの配布に不都合が生じる。MIDIではこのファイル保存フォーマットにおいても共通の取り決めがあり、この規格に従って保存されたファイルをスタンダードMIDIファイル(SMF)と呼ぶ。通常、シーケンス・ソフトは独自のファイル保存フォーマットの他に、このスタンダードMIDIファイルの読み書きができるようになっている。

一方、コンピューターのソフトウェア上でMIDIシーケンス・データを編集し、音楽を作り上げていく作業が一般的になると、必ず同期の問題が浮上する。つまり、録音機器とシーケンス・ソフトを同期させマルチトラック・レコーディングを行ったり、ビデオ・デッキなどの映像機器と同期させる必要が出てくる。例えばシーケンス・ソフトでドラムのパートを作成しMIDI音源モジュールで鳴らせながらテープに録音する。次にこのドラムに合わせてピアノのパートを多重録音しようとするれば、シーケンス・ソフトはテープの走行に合わせてピアノ・パートを再生しなければならないだろう。

同期とは、異なる機器がタイミングを共有しながら動作することで、このためMIDIでは、タイミング・クロックやMIDIタイムコード(MTC)という同期信号に関する規格を定めた。さらに録音機器や映像機器の操作自体もMIDIメッセージによりコントロールするMIDIマシンコントロール(MMC)という規格も追加されている。

● General MIDI (GM)

これまで説明したようにMIDIメッセージは“音を鳴らせ!”“音を止める!”といった機器に対する命令であって、肝心のどんな音色のサウンドが鳴るのかは受信側のMIDI機器の設定に任されている。これはMIDIシーケンス・データを一般に配布するような場合に大きな問題となる。ピアノ曲として作曲したものが、ハードなエレキギターの音色で再生されてしまっただけでは作者の音楽イメージは十分には伝わらないだろう。

MIDIにはもともとプログラム・チェンジというMIDIメッセージがあり、受信側の音色プログラムを変更することができるが、プログラム番号の1番が何の音色になるかまでは定義されていなかった。そこでMIDIではGeneral MIDI (GM)という規格を追加し、プログラム番号と音色との対応関係を定めている。これによりMIDIシーケンス・データ