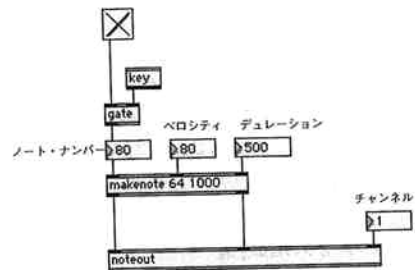


そして、その500msec後に、makenoteオブジェクトは自動的に右アウトレットから0、左アウトレットから60を出力し、それを受けてnoteoutオブジェクトはノート・ナンバー60、ベロシティ0、チャンネル1のノート・オフを送信することになる。

このmakenoteオブジェクトを使うことでキーボードでMIDI音源を演奏するパッチは次のように書き換えることができる。ノート・オフを作り出すための仕組み(keyupオブジェクトから始まる部分)がそっくり削除されているのに注目してほしい。その代わりにmakenoteオブジェクトが確実にノート・オフを作り出してくれるのでユーザーは安心して演奏を楽しむことができるだろう。ただし音の長さはすべて一定(4-4-5の設定では500msec)になる。

■4-4-5 キーボードで演奏するパッチ(3)



## 4-5 スケール変換MIDIフィルター

楽器インターフェースとしてのMIDIコントローラーは、鍵盤楽器やギター、管楽器などの形状をしており、これらになじんだ人間にとっては、ごく自然に演奏し、MIDI機器をコントロールすることができる。しかし、こうした楽器に触れたり学んだりした経験のない人間にとっては、適当に操作を試みても音を外してばかりで音楽の演奏というレベルには至らないかもしれない。音を外すと言っても意味はさまざまだが、一般的なのはスケールに含まれない音高の音を出してしまうことであろう。ここではスケール変換テーブルにより適当に弾いても音を外さないMIDIフィルター・パッチを作ってみよう。

### ① スケールとは？

音楽用語で言うスケール(音階)は、ある特定の音楽の中で使われる音の高さを、1オクターブの中に高さの順に並べたものである。クラシック音楽やそこから展開してきた今日のポップス音楽の多くは、長調や短調といった調性を持っており、そのメロディや和音で頻繁に使われる音を高さの順に並べ、オクターブの違いを無視するとスケールが得られる。例えばハ長調の音楽であればドレミファソラシというメジャー・スケール(長音階)になる。あるいは日本や沖縄、アラブ、ジプシーなどの民謡は一聴ただけで分かる独特なメロディの雰囲気を持っているが、こうした音楽にはそれぞれ特徴的なスケールを持っている。

4-5-1は、キーをCにした場合のメジャー・スケール(Cメジャー・スケール)の構成音だが、これを音名で表現するとC、D、E、F、G、A、B(ドレミファソラシ)となり、MIDIノート・ナンバーで表現すると各音は60、62、64、65、67、69、71になる。仮にデータラメにMIDIコントローラーを演奏しても常に音を外さないよう、このスケール上の音高でMIDI音源を発音させるためには、このスケール上にはない音高の鍵盤を弾いたときにも強制的にスケール上の音高に変換して発音させる必要がある。例えばノート・ナンバー61はC#でありCメジャー・スケール上にはない音だが、これをスケール上の近接した音、つまり60あるいは62に変換する。これから紹介する変換テーブルとはこのようなもので、具体的には60→60、61→60、62→62、63→62……といった1対1の変換を行う。

## ■4-5-1 Cメジャー・スケールの構成音

音名	C	D	E	F	G	A	B
	ド	レ	ミ	ファ	ソ	ラ	シ
ノート・ナンバー	60	62	64	65	67	69	71

## ② 変換テーブルに適したtableオブジェクト

もしこの対応関係がどちらも整数ならば、tableというオブジェクトを使うのが最適だろう。tableは、内部に多くの数値を格納するオブジェクトだ。ただしそれらの数値がごちゃ混ぜにならないように番地を用意し、そこに数値を格納する。数値を入力するときは、番地(x)に数値(y)を入れる方式で行い、取り出すときは番地(x)に入っている数値(y)を呼び出すことになる。この番地(x)のことをアドレスと呼び、格納すべき数値(y)をアドレスの値と呼んでいる。

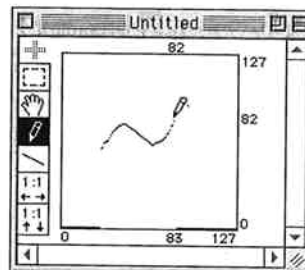
任意のアドレス(x)にアドレスの値(y)を格納するには、第1インレットにx、yというリスト形式のメッセージを与える。アドレスに格納されている値を取り出す際には、アドレスxを第1インレットに与えればよく、第1アウトレットからyが出力される。こうしてオブジェクトに格納されたいくつものx、yの組はパッチとともに保存できるし、外部ファイルとして名前を付けて保存したり、読み込んだりすることもできる。

## ■4-5-2 tableオブジェクトの基本的な使用方法



さらにtableオブジェクトは、アドレス(x)とアドレスの値(y)の対応を2次元の座標上の点として表すことができる。オブジェクト・ボックスをダブル・クリックして開くtableウィンドウは、横軸がアドレス、縦軸がアドレスの値となっており、現在アドレスに格納されている値を視覚的に確認できると同時に、鉛筆ツールなどで直接値を書き込むことで特定のアドレスに値を入力することも可能だ。初期状態として128×128のテーブルが作成され、すべてのアドレスには、最初0という値が格納されている。

## ■4-5-3 tableウィンドウ



## ② スケール変換テーブルの作成

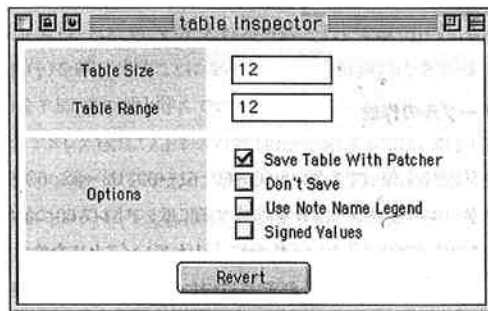
さて、このtableオブジェクトを使ってスケール60→60、61→60、62→62、63→62……といったCメジャー・スケールの変換テーブルを作成するには、アドレス60に値60、アドレス61に値60、アドレス62に値62……という具合に入力していくことになる。しかし、MIDIノート・ナンバーは0~127まであり、それぞれについて鉛筆ツールで書き込むのは面倒である。また、こうして作成された変換テーブルは、Cメジャー・スケールにしか使えず、さまざまなキーで使いたい場合は、DメジャーやF#メジャーなどのスケールごとに別のテーブルを用意しなければならないだろう。

しかし、オクターブの違いを無視すればこの変換はC→C、C#→C、D→D、D#→D……A#→A、B→Bのように12種類に要約できるはずだ。Cといっても鍵盤の左の方の低いCもあれば右端の高いCもあり、MIDIのノート・ナンバーで言うと36も60も72もすべて音名ではCになる。

共通しているのはこれらはみんな12で割った余りが0だということだ。同様に音名がFの音はすべて12で割った余りが5として表現することが可能になる。こうしてCメジャー・スケールの構成音は実際の音高ではなく音名で考えれば12種類になり、それらを12で割った余りで表せば0、2、4、5、7、9、11と簡単に表現できる。したがって変換も0→0、1→0、2→2、3→2、4→4、5→5、6→5、7→7、8→7、9→9、10→9、11→11と簡略化でき、テーブルの大きさは1オクターブの範囲つまり12×12で済むことになる。またキーに関してはいったんすべてをCメジャー・スケール上に変換してから指定したキーにトランスポーズ(移調)する方法をとれば変換テーブルは1つで済みそうだ。

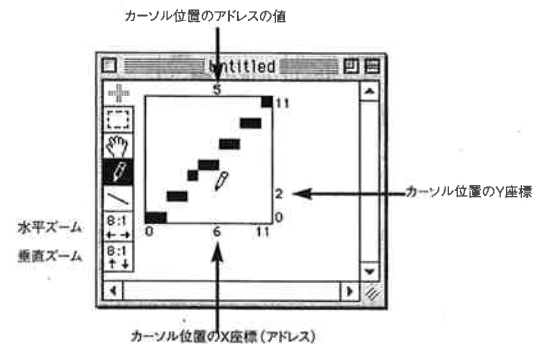
ではその変換テーブルを作成してみよう。まずパッチ・ウィンドウにtableオブジェクトを置くと自動的にtableウィンドウが開くので、その状態でObjectメニューからGet Info...を選び、インスペクターで4-5-4のように設定する。Save Table with Patcherをチェックしたので、tableオブジェクトの設定および内容はパッチとともに保存される。設定が完了すればクローズ・ボックスをクリックしてtableウィンドウに戻る。

#### ■4-5-4 tableオブジェクトのインスペクターの設定



tableオブジェクトの大きさを小さくしたので水平・垂直ズームを最大倍率8:1にして書き込みしやすくする。現在のカーソル位置が編集領域の右と下に表示され、カーソル位置のアドレスに書き込まれた現在の値は編集領域の上に表示されている。0→0、1→0、2→2、3→2……、10→9、11→11のそれぞれを、アドレスとその値として鉛筆ツールで書き込んでいけば、次のような変換テーブルを作ることができる。

#### ■4-5-5 スケール変換テーブルの作成

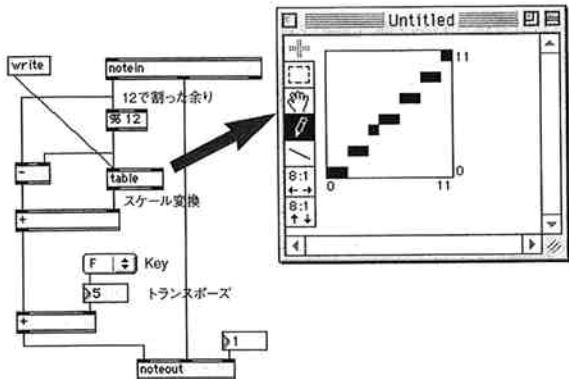


#### ● スケール変換パッチの作成

このtableオブジェクトを利用して作成したのが4-5-6のパッチだ。まず、noteinオブジェクトによってMIDIコントローラーから受信されたノート・メッセージのうちノート・ナンバーは% 12に渡される。“% 数字”はモジュロ演算子と呼ばれ、除算した余りを出力するオブジェクトであり、ここではアーギュメントとして書かれた12で割った余りを出力する。この結果すべてのノート・ナンバーは0~11までのいずれかの値になる。つまり音高をいったん音名(CやF)に変えたことになる。これを受けたtableは強制的にCメジャー・スケールへの変換を実行する。例えばノート・ナンバーが75だとすると% 12は3を出力し、それをtableはアドレスとして解釈し、そのアドレスの値である2を出力する。つまりD#→Dの変換を実行したということになる。

一方、こうして変換された音名は-と+によって実際にMIDI出力すべき音高つまりノート・ナンバーに戻される。こうしてノート・ナンバー75はCメジャー・スケール上の74に変換される。最後の+は一律にCメジャー・スケールに変換されたノート・ナンバー全体を平行移動させて、指定されたキーにトランスポーズ(移調)する。例えば5を加算するとFメジャー・スケールにキー・トランスポーズしたことになる。なお、5がFだと言われても分かりづらいので、umenuオブジェクトによりキーの音名を選べるようにしている。

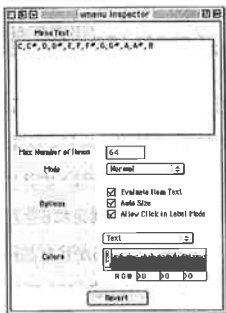
■4-5-6 スケール変換パッチ



umenuオブジェクトについてはインスペクターで次のようにメニュー項目の設定を行うと、ポップアップ・メニューの項目としてC、C#、D、D#、E、F、F#、G、G#、A、A#、Bが表示される。各項目には内部で先頭から0、1、2……という具合に順に項目番号が割り振られており、ユーザーがどれかの項目を選択すると、その項目番号が左アウトレットから出力されるので、ちょうどその数値をキートランスポーズの加算に使っている。

こうしてデタラメにMIDIコントローラーの鍵盤を操作しても、MIDI音源から鳴るサウンドは指定したキーのメジャー・スケール上の音になり、音を外すことはなくなるだろう。

■4-5-7 umenuオブジェクトのインスペクター

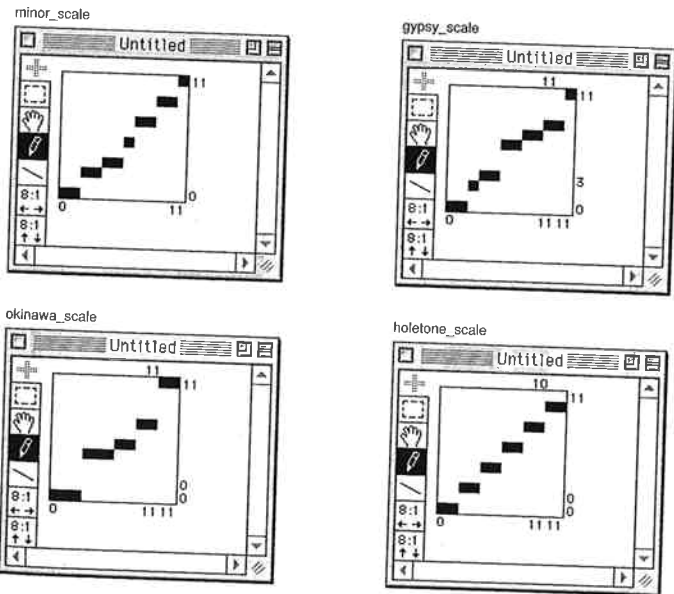


2) テーブル内容をセパレート・ファイルで保存する

ここで作成したメジャー・スケール変換テーブルの内容はパッチとともに保存されるが、メジャー・スケール以外にもいくつかのスケール変換テーブルを用意し、その都度取り替えたいとする。いくつか実現方法はあるが、ここでは外部ファイルとしてtableオブジェクトの内容をも別途保存し必要に応じて読み込む方法で実現してみよう。tableウィンドウが開いている状態でFileメニューからSave as...を選ぶと、現在のtableの内容を外部ファイルとしてパッチとは別に保存することができる。あるいはtableオブジェクトの左インレットにwriteというメッセージを送っても同様に保存することができる。ここではmajor\_scaleとでも名前を付けて保存しておこう。tableファイルを保存する場所はMaxサーチ・パスの範囲内にする。

こうして、メジャー・スケール以外に次のようにいくつかのスケール変換テーブルを作成し、その都度外部ファイルとして別名で保存する。

■4-5-8 いくつかのスケール変換テーブル

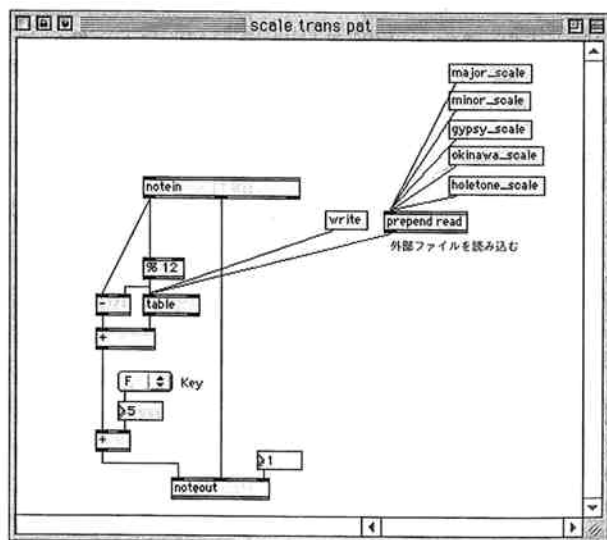


## 4-5-9 tableファイルを選択する

こうしてmajor\_scale、minor\_scale、gypsy\_scale、okinawa\_scale、holetone\_scaleという5つのtableファイルが外部ファイルとして保存された。これらを自由にパッチから読み込んで使用するために4-5-6のパッチを次のように拡張してみた。tableファイルを読み込むためにはtableオブジェクトにreadメッセージを与えればよいのだが、いちいちダイアログからファイルを選択するのではなく、readに続けてファイル名を与えれば自動的に読み込んでくれる。tableファイルの保存場所を“Maxサーチ・パスの範囲”と指定したのは、自動的に読み込むためにMaxが検索可能な範囲にファイルを置くという意味である。prependオブジェクトはアーギュメントとして書かれたもの(ここではread)を、受け取ったメッセージの前に付けて出力する。したがって、gypsy\_scaleを受け取れば、read gypsy\_scaleというメッセージ出力することになる。

これでtableオブジェクトは自動的にMaxサーチ・パスの範囲を検索し、gypsy\_scaleという名前のファイルを読み込む。

### ■4-5-9 スケール変換パッチの完成

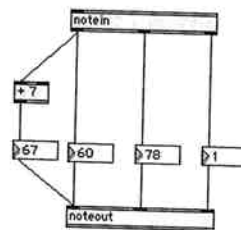


## 4-5-10 MIDIハーモナイザーへの拡張

4-5-9のパッチにほんの少し手を加えるとMIDIハーモナイザーが出来上がる。ハーモナイザーは、入力された音に対して音程を変えて出力する結果、原音にハーモニーを付加するものであるが、同様にMIDIハーモナイザーも入力されたノート・メッセージにノート・ナンバーの異なる別のノート・メッセージを付加する。

4-5-10のパッチでは、noteinオブジェクトはノート・メッセージを受信し、ノート・ナンバー60、ベロシティ78、チャンネル1としてパッチに取り込んでいる。それらはそのままnoteoutオブジェクトに渡されノート・メッセージとして送信されるのだが、同時にノート・ナンバーに7を加算したのもnoteoutオブジェクトに渡されている。Maxオブジェクトの共通した性質として、第1インレットにメッセージを受け取ると、それをきっかけにして処理を実行する。このためnoteoutオブジェクトはノート・ナンバー67、ベロシティ78、チャンネル1のノート・メッセージも送信することになり、合わせて2個のノート・オンを送信する。

### ■4-5-10 簡単なMIDIハーモナイザー



この方法を先ほどのパッチに応用すると、4-5-11のようになる。追加したgateを開くと、設定した音程差でもう1つのノート・メッセージが付加される。ただし、このノート・ナンバーもスケール変換テーブルを通すことで、指定したスケール上の音になり、音楽的に調和のとれたハーモニーが得られる。加算する値が4の場合、音楽的には3度上のハーモニーを付加していることになり、7の場合は5度上のハーモニーを付加していることになる。