

# 非楽器型コントローラを用いた 即興パフォーマンスのための MIDI システム設計

熊谷武洋

MIDI System Design for Musical Improvisation  
by Using Non-Musical Instrument Type Controller

Takehiro KUMAGAI

(Received September 28, 2007)

Key words: MIDI, Digital Music, DTM, Musical Improvisation

## 1. はじめに

本研究の目的は、既存の曲をなぞるのではなく即興的なアプローチから創作行為への積極参加を促すための音楽演奏システムを試作することである。基本コンセプトとして、“操作のための事前練習を不要とする”、“音楽性と即興性の両立を図るために音高と音価を分離して独立2系統による演奏する形態を採る”、“システム側のプログラムによって演奏者の基本入力情報に音楽的情報を付加する”ということを設定した。

制御操作のための事前練習が不要であるためには、従来の楽器形態模倣型以外の形態でありながらも、人間工学的に設計された入力装置でなければならない。このことを前提条件として、まず既存入力装置の検証と選定を行った。そして従来の鍵盤型や弦楽器型のような楽器形態を模倣した入力装置やプログラマブルな汎用的入出力装置は選外とし、非楽器型フィジカル MIDI コントローラを入力装置として選定した。そして選定した入力装置の操作特性を考慮し、暫定的な仕様策定を行った。本システムはあくまで楽器として楽しむ、演奏して楽しいという、体感的要素を前提としている。そのことを踏まえ、単純な操作による入力情報に対し、内部プログラムで入力情報を音楽的に再マッピングしたり、装飾的な情報を加えて出力する機能を付加した。

次に、開発用プログラミング言語として自由度の高い MIDI データのルーティングを可能にする機能を持つ cycling'74社のオーディオ・ビジュアルプログラミング環境 Max/MSP を用いて実装を行った。Max/MSP はラピッドプロトタイプング方式の開発スタイルに適しているため、途中で試作を作成しながら、仕様を逐次変更して、最終的な仕様を策定し、プロトタイプを完成させた。

当該システムの想定利用者として音楽の専門教育を受けていない小中高生や、音楽的素養が平均適度の一般的な人を前提としている。想定利用者による評価やそれらを反映した機能向上が今後の課題として残るが、システム開発における基本コンセプトを具体化することが出来た。

## 2. システムの基本コンセプト

楽曲の基本は音の高低（音高）とその長短（音価）である。当然のことながら、それ以外の音響的要素や表現上の複雑な要素も多く含まれているが、最低限この二つの要素があれば、芸術的であるかどうかは別として曲を奏でることが出来る。そして音高と音価を身体動作で直接的に操作する人、つまり演奏者の意思どおりに制御するための入力装置として完成されているのは、長い歴史を持つ、いわゆる楽器である。弦楽器、管楽器、鍵盤楽器をはじめとする様々な形態の楽器が音響学的な必然性を持ってデザインされている。しかしながら、こうした完成度が逆に音楽的な発展性を抑制しているとも言える。よって本論では、電子楽器とその制御技術が発達した今日、既存楽器を用いた従来型演奏形態の代替手段としてではなく、新しい可能性として電子楽器とその制御技術の特性を活かしたシステムを試案した。

既存楽器の場合は共鳴する本体部分や音を発振する音源部分とそれらを物理的に制御するインターフェイスが一体化しているが、電子楽器の場合は物理的に発振するのではないため、音源部分と制御するインターフェイスは分離している。よって、電子楽器の制御インターフェイスが見かけ上鍵盤や弦の形を模していても、それ自体は物理的な作用を持たず、電気的な信号を送るだけのスイッチである。それにも拘らず多くの電子楽器が従来の楽器の形態を模しているのは、利便性や商業性によるところが大きい。その効果と有用性は高いが、機能的形態としては必ずしも既存楽器を模す必要性はなく、楽器としての物理的制約を考慮しなくてよいのである。しかしながら、そのような既存楽器以外の制御インターフェイスを用いた場合、既成曲を記譜されたとおりに奏でる事は困難になることが考えられる。通常、楽曲というものは特定の形態を持った楽器の物理的な作用を前提して作曲される。その結果、その楽曲の中には、すでに楽曲と楽器との不可分な関係が必然的に生じるからである。

よって、既成曲を記譜されたとおりに奏でるための支援的なものではなく、既存楽器の模倣型でないインターフェイスを用いて、即興的な演奏を行いながら創作行為への積極参加を促すためのシステムという切り口から発想した。これまでも同様なシステムは開発されてきたが、音楽的表現性との両立や入力装置の操作性という点で技術先行的なシステムが少なくなかった。本システムはあくまで楽器として楽しむ、演奏して楽しいという、体感的要素を前提としている。このような基本コンセプトを基に以下のような仕様概要を設定した。

- 制御操作のための事前練習を不要とし、かつ即興性を高めるために音高と音価を分離して演奏する形態を採る
- 音楽性を高めるためにシステム側のプログラムによって演奏者の基本入力情報に音楽的情報を付加する
- 想定利用者として音楽の専門教育を受けていない小中高生や、音楽的素養が平均適度の一般人を前提とする

なお本論では、即興性という言葉の意味をプロのジャズピアニストが引き奏でるような熟達者による音楽的整合性に富んだものや、既存曲をその場でアレンジして演奏する行為のことでなく、思い描いたイメージあるいは音楽的意外性に富んだフレーズを、身体的な感覚で表出するという意味として用いる。

即興性と音楽性を両立させるためには、あらかじめ用意したフレーズをスイッチによってデー

タ駆動で呼び出すような方法が音楽性を高めるという点では有効であるが、即興性に欠ける。そこで、音高と音価を分離することにより、操作者の意図しない意外なフレージングを創出する方法を考案した。

音高と音価を分離した演奏形態を採ると、発音タイミングを制御することが困難になるため、もはや既成曲を演奏する事は不可能に近い。しかし、このことにより意図しないフレージングが創出される契機が増加する。しかし、これだけでは、音楽的な展開は望めない。したがって、入力装置はバリエーションの展開のタイミングとパターンを、音源発音のきっかけというかたちで操作を行い、プログラムによってバリエーブルな情報が付加され、最終的に発音されるという機能を付加する。具体的にはハーモナイジング、タイミングの量子化、音色パラメータへの同時アクセス等である。このことにより、バリエーションに富むインタラクティブなシステムを構築することが出来る。これまでも演奏者からの入力情報を基に、自動伴奏を行うなどの機能を持った電子楽器が製品として市販されてきたが、その制御操作スタイルと機能はあくまで従来型の演奏方法が前提となっている。本システムはそのような付加情報についても従来の方法によらないスタイルで操作可能とする。

このように入力部分と出力部分が簡便な操作による即興的音楽の生成という機能上の目的によって一体化されているため、入力装置の操作方法は、システムの仕様と直結している。入力装置の型によってシステムの仕様や、完成度が影響されると言える。そのため、システムの設計、実装段階の準備段階として入力装置の選定を行った。前述したように、入力装置の選定はシステム設計と表裏一体であり不可分な関係であり、システム設計と同様に重要な要素であるため、入力装置の選定については可能な限り実物を入手して検討を行った。

### 3. 入力装置

本システムの入力装置としての要件を満たすために以下の項目を選定条件として入力装置の検討を行った。

- ① 音高と音価値の独立2系統入力が可能であること
- ② 連続量と離散量両方の情報を取得できること
- ③ 操作のための事前練習が不要であること
- ④ 人間工学的に設計され、楽器として制御可能な操作性があること
- ⑤ 開発プログラム言語環境から簡便に制御できること

#### 3.1 入力装置の検討と選定

入力装置については、仕様が公開され、開発プログラム言語上で制御できるものであれば、MIDI 規格に準拠している必要はない。よって、MIDI 以外にもゲームコントローラや汎用デバイスについても同様に選定対象とした。そして上記の項目を条件として選定を行った。まず、それぞれの特徴や機能について調査を行い、本論におけるシステムの要件に合致する入力装置の検討と選定を行った。

##### 3.1.1 楽器型 MIDI 対応コントローラ

MIDI とは Musical Instrument Digital Interface の略称であり、電子楽器制御のための通信プロトコルである。この規格はオーバーハイム社（ドイツ）、ローランド社（日本）、シーケ

ンシャルサーキット社（米国）等の複数の国の電子楽器メーカーによって1982年に提唱された。「どの音高をどのような音色でどのくらいの時間鳴らすか」という演奏情報や演奏における付加的な情報に加えて、機器の環境等を定義したステータスといった情報を8ビットで構成している。よって、このMIDI規格に準拠していれば、物理的にはどのような入力装置も開発可能であるが、一般に市販されているものは、その操作性と商業的理由からそのほとんどが既存楽器の模倣形態となっており、最も多いのが鍵盤型である。鍵盤型以外のMIDI対応コントローラについて代表的なものを以下に挙げる。

### ●ウインドシンセサイザー

木管や金管の管楽器と同じような奏法でシンセサイザーや電子音源をMIDI信号でコントロールできる電子楽器である。代表的な機種にコンピュートン社のLyricon、ヤマハ社のWXシリーズ、アカイ社のEWIシリーズがある（図1）。

ウインドシンセサイザーによる演奏情報は、鍵盤型のコントローラでは得ることが困難な滑らかな音量変化や音色・ピッチの変化を出力することができる。実際の管楽器のように本体から音響学的に音を発振するのではなく、マウスピースからの振動や圧力、運指の物理的な情報をMIDI信号化して外部音源に送るため、音色や音域についての制限は少ない。

ただし、鍵盤楽器に比べ管楽器は一般的でないためにコントローラの普及率も低い。

### ●ギターシンセサイザー

ギターのような弦楽器と同じような奏法でシンセサイザーや電子音源をMIDI信号でコントロールできる電子楽器である。ギターの弦振動をセンサーでピックアップし、そこから音高及び音量情報を抽出、MIDI信号に変換する。代表的な機種にヤマハ社のGシリーズ、Roland社のGRシリーズがある（図2）。

上記以外にも様々なコントローラが市販されているが、こうしたMIDI対応コントローラというものは、演奏ジェスチャーを電気的な信号に変換し、MIDI信号によってMIDI音源から出力というスタイルである（図3）。楽器のような共鳴するボディや音を発振する音源部分が必要でないため、もっと斬新なデザインのMIDI対応コントローラも開発可能ではあるが、前述した理由により、一般的なMIDI対応コントローラは、既存の楽器模倣型が主流である。

しかしながら、その入力装置の持つ表現力を最大限に発揮する、あるいは最も適した形で制御や操作をするためには、モデルとなった楽器の奏法に熟達している必要がある。よって全く楽器を扱ったことのない初心者には扱い難い入力装置と言える。加えて、音高と音価を分離し、二系統の信号を出力する仕様には適当ではない。換言すると、既成楽器とは音高と音価をワンアクションで実現することができる優れたインターフェイスデザインが施されていると言えるのである。よって本論における独立2系統入力の入力装置としては適当でないと判断し、選外とした。



図1 ヤマハ社のWXシリーズ

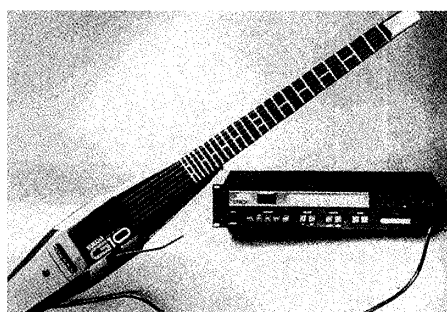


図2 ヤマハ社のGシリーズ



図3 ローランド社の電子ドラム

### 3.1.2 ゲームコントローラ

ゲーム機などで使われる入力装置の一つであり、マウス同様、画面上の入力位置や座標を指定するポインティングデバイスである。十字キーなど方向を指示するための方向キーと、複数のボタン型スイッチによって構成されているものが一般的である。その他にもゲームのプレイスタイルに合わせた様々な形態のコントローラが市販されている。前述した MIDI 対応コントローラとは異なり、MIDI 信号を出力する装置として設計されていないが、人間が直接操作し、老若男女誰でも使えるインターフェイスとしては最も一般的で完成形に近いデバイスであると言える。

#### ●ゲームパッド

最も普及しており、一般的な入力装置である。片手でも保持・操作が可能であり、携帯性に富んでいる。方向指示を行うためのキーや、機能実行する複数のボタンによって構成されている。利用シーンの多くが、ゲーム利用であるため、基本的には離散的なデジタル信号を処理するが、アナログ信号にも対応したモデルがある (図4)。

家庭用ゲーム機が普及した今日では、リテラシーも十分に浸透しており、入力装置としては最も実績がある。よって使いこなすための事前練習は全く必要ないといっても過言ではない。実際にゲームパッドを演奏のための入力装置として用いたテクノモーターというソフトがある (テクノモーターについては後述する)。ゲームパッドという性質上、音高と音価値の独立2系統入力については適しているものの、アナログ入力に適さないため、選外とした。

#### ●ゲームマット

指でキーやボタンを押す代わりに足を用いて踊るように押すというコントローラである (図5)。指ではなく、身体動作そのものを入力のための操作とした入力装置は従来にもいくつか存在したが、そのようなコントローラを必然的に用いるようにデザインされたゲームが少なかった。コナミ社が開発したダンスゲームである DDR (ダンスダンスレボリューション) では、そのゲームデザインとコントローラとの間に必然性と娯楽性が生じ、ヒットとなった。なお、当該ゲームは米国において公教育での導入実績がある。ゲームマットは機構的には異なるが信号処理はゲームパッドと同様である。工夫次第で利用可能性が広がる入力装置ではあるが、音高と音価値の独立2系統入力には不向きであるため選外とした。

その他にも、ロータリーコントローラ、パドル、トラックボール、ボディパッドといった内

部の信号処理は同じでも操作感の異なる入力装置として様々な型が市販されている。(図6)

いずれのタイプであっても、WindowsOS ゲームコントロールデバイスとして認識できれば開発プログラミング言語環境から仕様上の差異を吸収できる。よって、家庭用ゲーム機に対応した各種コントローラを入力装置として用いることが出来る。これらのデバイスはC言語などの汎用開発言語によって制御できるだけでなく、Max/MSP や Processing などのような視聴覚情報処理専用開発言語からも直接制御可能である。

このようにゲームコントローラは多くの条件を満たすものの、入力可能な信号の種別に物理的な限界があり、独自仕様のものや仕様が非公開されているものも少なくない。非公開であっても、例えば Wii のコントローラをハッキングしたものや、解析して MAX/MSP など外部入力デバイスとして応用した例もあるが、そもそも入力デバイスとして MIDI 対応ではないので安定動作の保障はない。

コントローラの型に多くのバリエーションがあるため幅広い応用可能性を期待できるが、音高と音価値の独立2系統入力を行うためには、不向きであると判断した。よって本論におけるシステムの入力装置としては、今回は候補に留めた。



図4 ゲームパッド

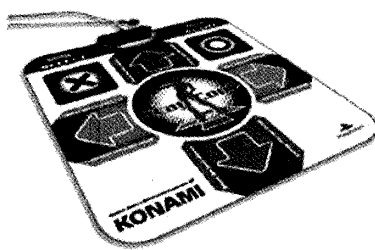


図5 ゲームマット



図6 その他のゲームコントローラー

### 3.1.3 汎用型コントローラ

MIDI やゲームといった特定の目的と機能をもった専用コントローラ以外にも、利用者の意図する目的に合致させるための汎用的なデバイスや、人間の随意的な挙動以外の情報を入出力情報として取得するための様々なセンシング装置がある。そしてそれらの装置によって取得されたアナログ量はデジタル値に変換され、中間的な処理ソフトによって MIDI 化され、さらに本研究で用いているような視聴覚情報処理専用開発言語等によって音楽情報として音楽的に意味のある情報にマッピングすることが可能である。

代表的なものとして加速度や重力を測定する計測装置 I-CubeX や脳波や体温を測定する生態反応センサーである IBVA などがある。これらのデバイスはC言語などの汎用開発言語によって制御できるだけでなく Max/MSP や Processing などのような視聴覚情報処理専用開発言語から直接制御可能である。本論では便宜上コントローラと呼称しているが、厳密にはコントローラとしての使い方もある、という言い方がより正確である。以下に具体例を挙げる。

#### ●Gainer

Gainer とは、2005年4月より活動を開始した IAMAS (情報科学芸術大学院大学+国際情報科学芸術アカデミー) の研究プロジェクトである IAMAS PDP (プログラマブル・デバイス・

プロジェクト) によって開発されたインタラクティブな作品やインターフェイスを試作するためのツールキットである (図7)。電子回路の試作で用いられるブレッドボードと組み合わせることで、物理的な入出力が可能となり、自由に回路を変更することが出来る。2006年6月に公開された Gainer v1.0は、さまざまなメディア・アート作品における利用実績がある。

### ●The Lemur

Lemur とは、フランスの JazzMutant が開発したマルチタッチおよびプログラマブルな液晶コントロールパネルである (図8)。PC 側で開発用専用ソフトによって制御用インターフェイスをプログラミングし、コンパイル後、実行ファイルを通信で Lemur に転送する。Lemur 自体は MIDI 未対応であるが、制御ソフトが MIDI 処理を行なえるので汎用 MIDI 入出力装置として用いることが出来る。

このような様々な汎用入出力装置を使うことによってアイデア次第で様々な試みを行うことが出来る。そして実際に多くのシステムや芸術作品が試作、制作されてきた。

しかしながら、汎用入出力装置を用いた視聴覚システムや芸術作品が表現的に魅力的であるか、あるいは表現と装置の制御上の必然性があるか、という点においてはまだ途上にあるものも少なくない。扇風機の回転量を応用したインタラクティブ・アートや体重計の測定値を用いたダイエット音楽といった類のものも、それ自体はシステム開発や芸術作品として有意義であると考えるが、前述したように本論ではあくまで楽器という範疇を前提としている。つまり、意思を伴う随意的な身体動作や挙動によって入出力制御を行う、具体的には手と指を使って直接音高と音価を操作するということである。汎用入出力装置をベースに新しいコントローラを自作するというアプローチもあるが、それだけで独立した研究対象となりうるので、今回は汎用入出力装置によるコントローラは選外とした。

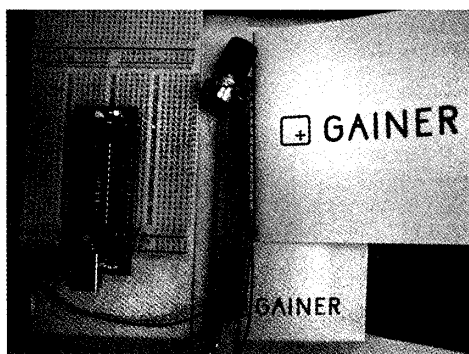


図7 Gainer 一式



図8 Lemur

#### 3.1.4 非楽器型 MIDI 対応コントローラ

MIDI 対応であるが、楽器のように演奏を行うのではなく、本来的な用途としては音量調性、ピッチ調整、音色パラメータなどの演奏情報以外の付随的な情報を入力するためのコントローラである (図9)。

一般的にはフィジカルコントローラと称してリボンコントローラ、フェーダー、ノブ、アサインャブルホイールといった回転運動や上下運動をする操作のボタンやスイッチなどが備えられている。本システムにおいては、音楽的な調性と簡便性から、入力装置から取得した入力情

報にフィルタをかけて、得られたデータをマッピングすることにより音楽的に意味のあるものに変換することを想定している。よって、操作感覚はアナログ感覚であっても取得できる値は離散的にしたい。そのような目的にはこのような非楽器型 MIDI 対応コントローラは合致している。

### ●ピッチベンドホイール

鍵盤楽器などの場合は、鍵盤に割り当てられる周波数が固定のために、弦楽器のように音程間で連続したニュアンスのある音を出すことができないが、ピッチベンドホイールを用いることによって、連続的に変化する情報を出力し、微妙な音程を音源に送信することが可能となる。

このようにピッチベンドホイールは本来的な用途としては演奏のための直接情報を入力する手段ではないが、それをあえて手段として検討した。しかし、ピッチベンドホイールはバネなどによる抵抗があり、支持する力が持続的にないと、中央の定点に戻るよう調整されている(図10)。意図した音高や音価を持続させたりするような操作には不向きである。よってピッチベンドホイールは選外とした。

### ●アサインابلホイール

本来的な用途としては音源側のフィルター、エンベロープ、レゾナンスといった音色に関わるパラメータを連像的に変化させるためのコントローラである。ピッチベンドホイールが演奏情報に付随する情報を入力するのに対し、アサインابلホイールは音源側の音色情報に付随する情報を入力するコントローラである(図10)。アサインابلホイール同様、連続量を入力する操作に適した設計であるが、バネの力によって中央の定点に戻るよう調整されていないため、任意の位置でホイールの角度を固定させることができる。

最大180度の回転角度に対して、7bit の情報が割り当てられているため、128段階の情報を取得できるが、プログラムによってこれをフィルタリングすることが可能である。例えば、180度の回転で、2オクターブ分の演奏情報として値取得し、それをCメジャースケールのノートナンバーとしてアサインし、アサインابلホイールで演奏といったような応用が考えられる。よってアサインابلホイールを候補として選定した。



図9 フィジカルコントローラ

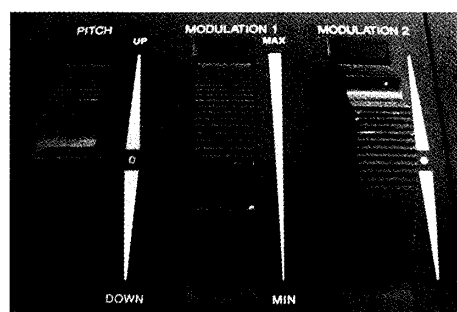


図10 ピッチベンドホイール(左)、アサインابلホイール(中央、右)

### ●ジョイスティック

ピッチベンドホイールとアサインابلホイールの両方の機能を一体化したコントローラである。それぞれの機能を縦軸と横軸に置き換えて、一本のスティックで操作を行う。操作感はホイール型と比較するとあまり優れていないが、同時に2系統の入力を行うことが出来、加えて設置面積が少ないという利点があるため、コンパクト型のキーボードに用いられることが多い。



KORG 社や ROLAND 社が積極的に採用している (図11, 図12)。

しかしながら、常にスティックに指をかけて力を持続的にかけておかねばセンターポジションに戻ってしまうため、ピッチベンドホイールと同様の理由により、操作性の観点から選外とした。

### ●ノブ

入力装置としては連続量を取得できるが、フィルタリングによって離散量にも対応できる。また、駆動部分が一つの軸を中心に固定で回転するため多くのノブを狭い面積で併置させることが可能である (図13)。加えてパラメータの設定値を目視で確認することが出来、状態の視認性に優れている。

ノブの場合は、その動作の特性により、一本の指で動かす事は困難であり、意図したとおりに動かすには二本以上の指を使うことになる。しかし、換言すれば、非常に些細で繊細なコントロールが出来る、ということでもある。そして値の範囲に対して大きい移動範囲を取れるため、直感的に操作できる。実際にレコーディングスタジオのコンソール卓や DJ のミキサーで最も用いられている入力装置である。エモーショナルな楽器演奏のための操作子という目的にも合致すると判断し、ノブを候補として選定した。

### ●フェーダ

利点および欠点は、ノブとレードオフの関係にあると言ってよい。入力装置としては連続量を取得できるが、フィルタリングによって離散量にも対応できる、パラメータの設定値を目視で確認することが出来るため状態の視認性に優れているという点は同様である (図14)。ただしノブと異なり一本の指で制御できる反面、設置面積がノブよりは多く必要である。しかしながら、隣接するパラメータとの関連性を把握しやすいという利点があり、100mm 前後を基準として、そのフェーダの長さによって操作精度を変えることが出来る [1]。

フェーダもノブと同様にしてレコーディングスタジオのコンソール卓で最も用いられている入力装置であるが、それだけでなく実際に本システム同様にフェーダを演奏のための入力装置として用いたフェーダーボードという機器も市販されている (フェーダーボードについては後述する)

このようにフェーダは、汎用的でありながら、演奏のための入力装置としての潜在的可能性を秘めていると言える。よってフェーダを候補として選定した。

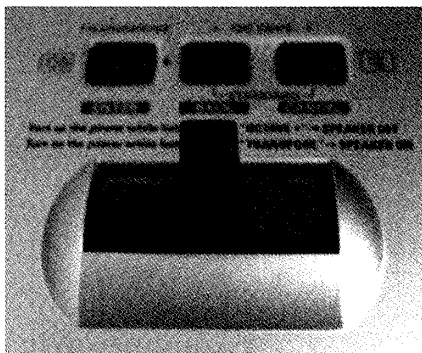


図11 ローランド社のスティック

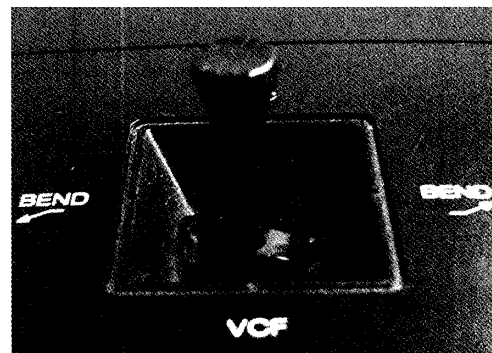


図12 コルグ社のスティック



図13 ノブ

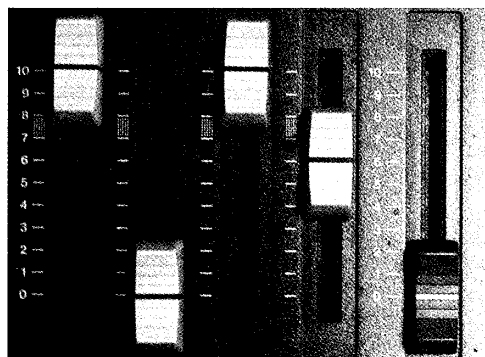


図14 フェーダ

### 3.2 入力装置の選定結果

本システムの入力装置としての要件を満たすための項目を満たす入力装置の候補として、アサインブルホイール、ノブ、フェーダを挙げた。これらの入力装置は、音高と音価値の二系統入力が可能であり、連続量離散量アナログデジタル両方の情報を取得できる。また、本来汎用入力装置として設計されているので、楽器として制御可能な操作性があり、操作のための事前練習が不要である。そしてMIDI対応であるため、開発プログラム言語環境から簡便に制御することができる。よって検討の結果、フェーダとノブを相補的に運用することで高い相乗効果を期待できると考え、結果的にこれらの入力装置を選定した。

## 4. システムの仕様

次に基本コンセプトをより具体的に要求定義し、暫定的な仕様化を行った。

実際には、プログラムを逐次試行させながら仕様を定めた。このような感性指向型のソフトウェア開発の場合、ドキュメントとして定義できない不確定要素が多いため、ウォーターフォール方式ではなく、ラピッドタイピング方式が適している。よって、開発用プログラム言語も、そのような開発スタイルに合致したMax/MSPを用いた。

### 4.1 暫定仕様策定

#### ●入力装置の割り当て

音高と音価を分離させるために、音高についてはフェーダ、音価についてはノブに割り当てる(図15)。フレーズシンセサイザーや、プログラマブルアルペジエーターを参考にし、音高については2オクターブの音域とする。

両手を音高と音価に分離するため、鍵盤楽器のように同時性を持って和音を奏でることは出来ないが、フェーダは指一本で制御できるため複数のフェーダとノブを用いてポリニク演奏も可能にする(図16)。しかし、操作が煩雑になるため、今回の試作においては、操作子は1組だけとする。操作子による入力は音高と音価のみであるため、発音タイミングはシステム側のプログラムで自動的に発生させることとする。なお、このような点はMIDIのような電子楽器ならではの長所である[2]。

#### ●ゲートタイミング

入力された演奏情報がリズムミクになるように発音のタイミングを指定したBPMをもとに

全音符、二分音符、三分、四分、八分、十六分、二十四分、三十二分と8段階に変化させる。

### ●スケールマッピング

どのように操作しても音楽的に整った響きにするために、プリセットとして西洋音楽におけるCメジャースケールにマッピングを行う。必要に応じて半音を省いてペンタトニックにしたり、その他のスケールに随意に変更可能にもする。

### ●リズム支援機能

操作者のリズム感を安定させ最低限の音楽性を維持することができるようにするためメトロノームのようにメインテンポに合わせて自動的にリズムを奏でる機能を付加する。

### ●ハーモナイザー

入力に対して出力される情報の表現力を高めるために、プログラムによってハーモナイズさせる機能を実装する。あまり複雑な和音やハーモナイジングは逆効果であるため、ハーモニーの生成規則はユニゾンのものや根音に完全5度、長3音を加えた長3和音のような耳あたりがよく単純なものにする。なお、アルペジオ機能は発音タイミングを随時可変させるため聞きづらく、効果的でないと判断し、今回は実装しないものとする。

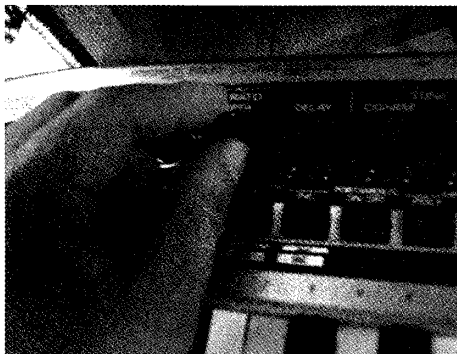


図15 ノブを二本の指で操作する



図16 複数のスライダーを指で同時に操作する

## 4.2 関連・参考事例

次に関連事例や参考事例として本研究におけるシステムのコンセプトに比較的近似している機器を以下に挙げる。

### 4.2.1 テクノモーター

テクノモーターは、ゲームパッドを使って家庭用ゲーム機を楽器として遊ぶためのセガ社サターン用ソフトウェアである(図17)。ゲームパッド上の各ボタンに音階や機能を割り当て、鍵盤のようにボタンを押すことにより演奏を行う(図18)。

機能ボタン：

X=ド、Y=レ、Z=ミ、A=ソ、B=ラ、C=シ、L=ピッチベンド下げ、R=ピッチベンド上げ

方向キー：

左右方向キーと、機能ボタン同時押し＝半音（bと#）

上下方向キーと、機能ボタン同時押し＝オクターブの変更

例）“ドレミファソラシド”の場合、X,Y,Z,Z+,A,B,C,C+とボタンを押す

しかし、事前の練習がなければ意図した演奏は困難である。よって、テクノモーターには、簡易にメロディ演奏を可能とするために音階だけ先に登録し、ボタンを押す度に登録した順に音が鳴るようにできるメロディモードやコード、アルペジオといった半自動演奏のような支援機能が加えられている。

このように試みとしては興味深い音楽的表現としては従来のであり、本システムとはコンセプトが異なる。しかし、ゲームパッドというコントローラを演奏という目的に対して積極的に用いた好例であるため、参考例として挙げた。

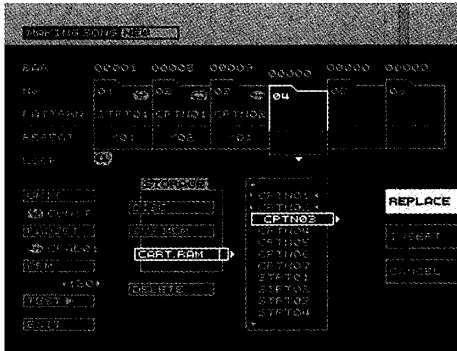


図17 テクノモータプレイ画面



図18 セガサターン用ゲームパッド

#### 4.2.2 FADERBOARD（フェーダーボード）

ベスタクス社のフェーダーボードは、本システム同様に楽器演奏の操作をフェーダのスライディング操作に置換した楽器である（図19、図20）。内蔵音源や入力された音源を全て音楽的に調整されたスケールにマッピングし、1度ずつ音階としてフェーダにアサインすることが可能である。これを各フェーダ毎に行うことが出来るので、フェーダの数が最大同時発音数である。オクターブの切替スイッチを用いて、3オクターブまでの音域で演奏でき、演奏に必要なリズム調整・強弱・ミュート等の表現は、フェーダのカーブ調整でコントロールすることができる。フェーダへのアサインは、MAJOR、MINOR、ARABIC、GYPSY、BLUE NOTEの計5種類のスケールから選択することができ、楽曲に合わせたスケールで演奏が可能である。音価についてはフェーダを上げている時間が発音時間と設定されている。つまり、フェーダを上げる瞬間に発音され、その音の持続を制御するには、ノートオフ信号の代わりに、もう一度フェーダを下げて音量をゼロに戻す仕様になっている。この操作感は大変興味深く、本システムのコンセプトに最も近いが、一つの制御子で音高と音価を制御するという点で従来型の楽器に区分されるといえる。

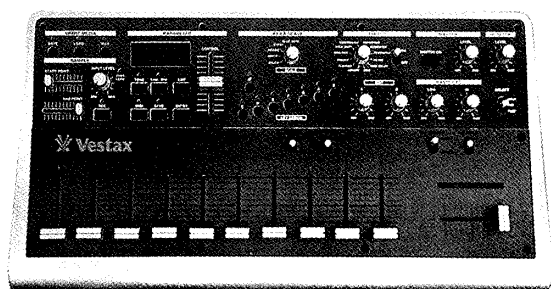


図19 フェーダーボード本体

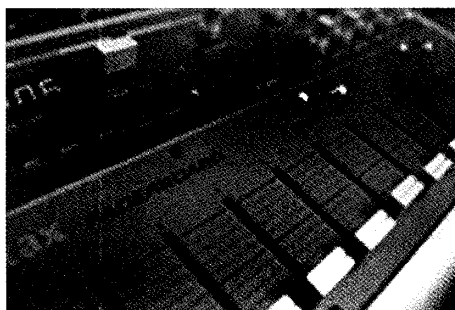


図20 フェーダーボードのフェーダ

## 5. システムの仕様

### 5.1 開発環境

本論におけるシステムを開発する上で用いた開発用機材および環境は以下のとおりである。

#### ●開発プログラム言語

演奏の制御を行うにあたって、自由度の高い MIDI データのルーティングを可能にする機能を持つ cycling'74 社のオーディオ・ビジュアルプログラミング環境のソフトウェア Max/MSP を用いた。Max/MSP では、外部の MIDI コントローラから入力される MIDI 情報を、オーディオ MIDI インターフェースより受け取り、演奏処理にふさわしいように加工を施し、最終的に音声を発音させる諸音源に出力を行う。

#### ●MIDI 対応フィジカルコントローラ

フェーダ、ノブによる入力装置として、EDIROL 社の MIDI コントローラ PCR-A30 を用いた(図21)。なお、鍵盤部は使用していない。

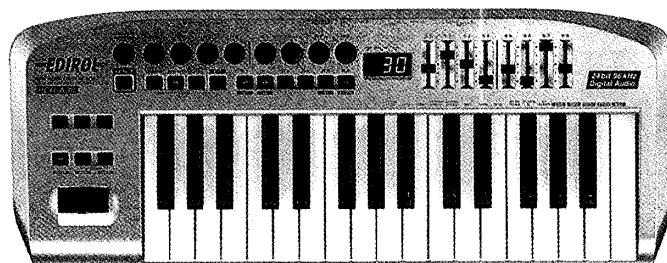


図21 PCR-A30

#### ●ソフトウェア音源

発音を行う音源として、ソフトウェア音源である Steinberg 社の Hypersonic2 を用いた(図22)。GM として動作が軽く、音色ライブラリが豊富である。

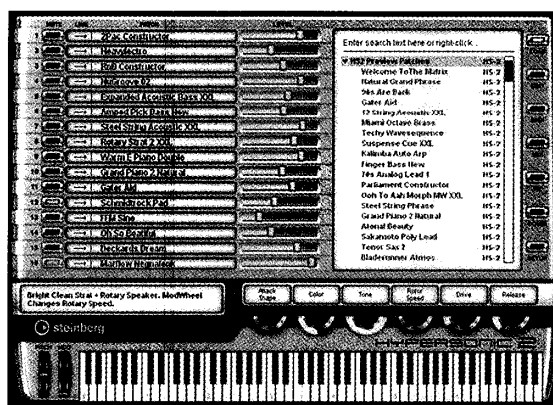


図22 Hypersonic2メイン画面

## 5.2 音高と音価の独立2系統制御の処理

PCR-A30のフェーダやノブは、様々な種類を有する MIDI メッセージの中でも、MIDI 機器のさまざまなパラメータを連続的にコントロールするコントロール・チェンジ・メッセージという種の多目的な MIDI メッセージを出力する。コントロール・チェンジ・メッセージは2個のデータ・バイトを持ち、フェーダやノブを個々に識別するための固有値としてのコントロール番号(0~127)と、フェーダやノブが個々に指し示す値を表すコントロール値(0~127)の組み合わせで構成されている。

まず、Max/MSP 側で、フェーダやノブの情報をこれらのコントロール・チェンジ・メッセージとして取得する。Max/MSP 内にコントロール・チェンジ・メッセージを取得するために、コントロール・チェンジ・メッセージの取得のための専用オブジェクトである ctlin オブジェクトを使用した。ctlin オブジェクトでは、コントロール・チェンジ・メッセージのコントロール番号とコントロール値の2つの値を独立して取得が可能であり、フェーダならびにノブの固有値であるコントロール番号を Hypersonic2側の操作したい部位のパラメータの固有値に、コントロール値もパラメータの値に変換し直し、これらの値をともに Hypersonic2側に送信することで、PCR-A30のフェーダ、ノブを用いて Hypersonic2を操作することが可能となる [3]。

PCR-A30のフェーダ、ノブから Hypersonic2の操作が可能になった次段階として、フェーダ、ノブから音高、音価、発音のタイミングなどをコントロールする機構を考える。なお、この過程においては、非常に多様な試行錯誤を要した。ここでは、その試行錯誤の過程を含めて述べる。

### 5.2.1 発音タイミング手動化による音高と音価値の1系統制御

まず、はじめに、どのように音を発生させるかという、発音を行うメカニズムに関して考案する必要があった。ピアノなどに代表される鍵盤では打鍵と同時に発音が行われるように、個々の鍵盤それぞれから発音を行うことが出来る。そこで単純に鍵盤同様に複数のフェーダとノブにコントローラ上に並んでいる順で左から右へ、コントロール番号にドレミファソラシと音高を当てはめ、コントロール値に音量のパラメータを当てはめて、個々の操作子から発音し操作を行う処理過程を設計した。

実際に使用した際、音高の操作としては鍵盤とそれほど変わらなかったが、操作感覚として

は、打鍵で操作出来る鍵盤に対して、音量の増減で発音と消音を制御しなければならない点で鍵盤よりも煩雑な操作を求められた。また音楽的意外性を誘発する契機が実装されていないため、即興演奏を行うシステムという観点からも不向きであるという結論に至った。この結果を受けて個々の操作子から発音の操作を行うシステムに関しては、この場合不向きであると判断し、当初の基本コンセプトの方向性を確認した。

### 5.2.2 発音タイミング自動化による音高と音価値の独立2系統制御

次に基本コンセプトである音高と音価値の独立2系統制御による処理過程を設計した。音高と音価を独立に2系統に分離すると、発音タイミングを操作することができない。そのため、発音タイミングを操作子で行わず自動で処理するシステムについて考察した。これは、Max/MSP 内で独自に信号を生成して、この信号を発音の命令に使用するものである。信号は連続的に発生させることが可能であり、信号を発生させるタイミングに関して、専用オブジェクトを用いることで、最短で1ms 単位で信号生成が可能である。また音楽使用向けに BPM(テンポ)のタイミングに合わせて信号生成させることも可能である[4]。今回は、BPM のタイミングに合わせて信号生成が可能な tempo という専用オブジェクトを発音の制御に用いた(図23)。

フェーダ、ノブから音高、音価、発音のタイミングなどをコントロールして演奏を行う機構を考案するにあたって、各パラメータの制御をどのように行うか、どのような変化域を設けてコントロールすると効果的であるかという部分から設計する必要があった。また、どのようなパラメータがフェーダでの操作に相応しいか、ノブでの操作に相応しいかということについて人間工学的な観点から設計した。

音高に関しては、低い音から高い音に変化するという性質上、「上下する」ものとして捉えることが体感的に理解しやすいと考えられる。その結果、音高の操作子としては回して操作するノブの形態の操作子よりも、縦軸を上下させて操作するフェーダの形態の操作子にアサインし、音価は、ノブにアサインした。このように、操作する対象のパラメータと操作子の関係性についてアフォードさせることにより対象のパラメータを最も体感的にイメージしながら操作出来るように設計した。

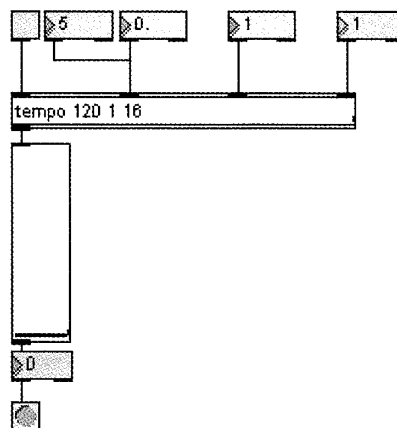


図23 発音の命令のための信号を生成する処理

### 5.3 処理過程

#### 5.3.1 全体構成

本システムは、大きく以下の3つの処理過程によって構成されている。システム全体を通して流れていく情報はMIDI信号である(図24)。

- ① 操作情報を入力装置から取得する処理
- ② 取得した情報に二次加工を加える処理
- ③ 二次加工を加えた情報を音源に渡す処理

こうした処理過程をより詳細に定義・仕様化し、Max/MSPのプログラムとして実装を行った(図25)。次に各処理過程の機能と内容について述べる。

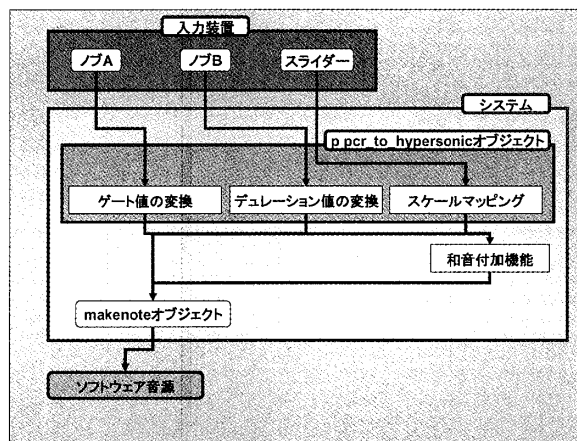


図24 処理過程の概要図

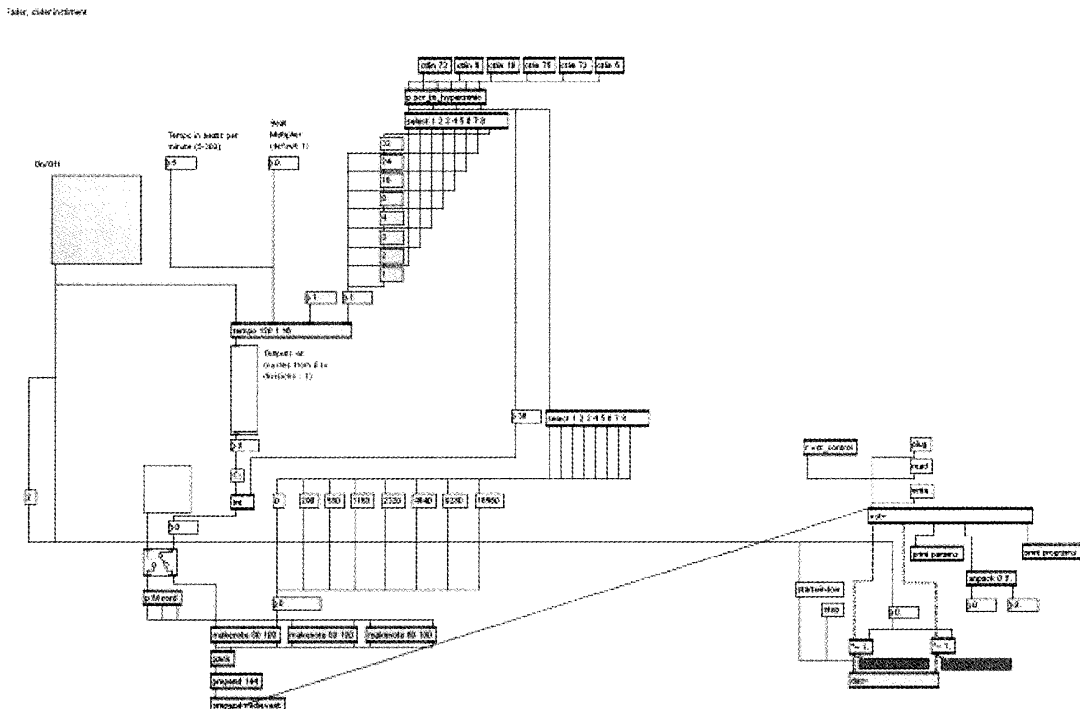


図25 MAX/MSPによるプログラム全体図



### 5.3.2 入力装置からのパラメータ処理過程

入力装置からのパラメータ処理過程について述べる。PCR-A30側からの MIDI 信号であるコントロール・チェンジ・メッセージを Max/MSP 側の ctlin オブジェクトから p pcr\_to\_hypersonic というオブジェクトに入力し、入力されたコントロール・チェンジ・メッセージを Max/MSP や hypersonic で扱う値に変換を行った。

ctlin オブジェクト内に記されている番号がコントロール・チェンジ・メッセージにおけるフェーダやノブを個々に識別するための固有値としてのコントロール番号を示しており、ctlin オブジェクト内にコントロール番号を入力することで特定のフェーダならびにノブを指定することが出来る。

それぞれの ctlin オブジェクトには指定されたフェーダやノブの個々のコントロール値である、フェーダやノブが個々に指し示す値 (0~127) が入力され、ctlin オブジェクトを経由して p pcr\_to\_hypersonic オブジェクトに格納される。このオブジェクト内で出力先別にコントロール値の変換を加えた後、各パラメータに出力されている (図26)。

この処理によりフェーダ、ノブを用いてゲート、デュレーション、ノートナンバー (音高) のパラメータ値の操作を可能にしている

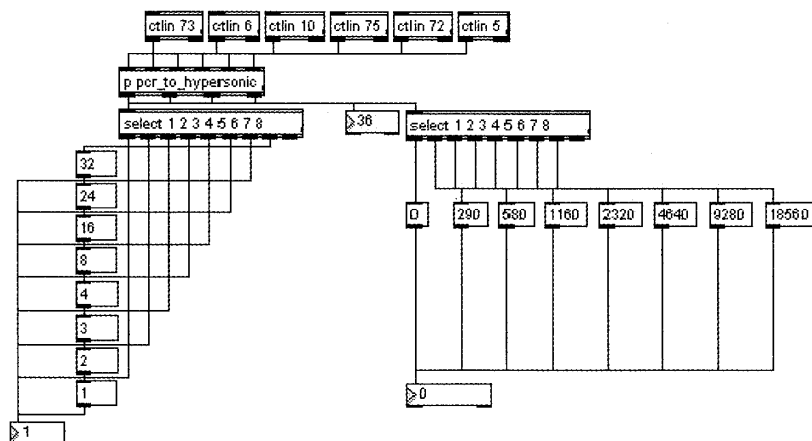


図26 MIDI メッセージの変換を行う処理

また、p pcr\_to\_hypersonic オブジェクトについて、Max/MSP ではオブジェクト名の先頭に p と記名し、一文字分スペースを空けて任意の名称を記すことでユーザーオリジナルのオブジェクトを作成が可能である。概念的にはこれはルーティングされた複数のオブジェクト群を1つのオブジェクトのように使用することが出来る機能であり、オブジェクトのルーティングを行うウインドウ内のスペースを有効的に使用することが出来ることからパッチ全体の見通しをよくするという利便性がある。

この p pcr\_to\_hypersonic オブジェクトでも、先に述べた通り、入力されたコントロール・チェンジ・メッセージを Max/MSP や Hypersonic で扱う値に変換する処理を行うオブジェクト群をまとめており、このオブジェクト内では、ゲート、デュレーション、ノートナンバー (音高) のそれぞれのパラメータ値の変換処理を行っている (図27)。

次項より、このオブジェクトの内部で行われるそれぞれの処理の内容について詳しく触れていく。

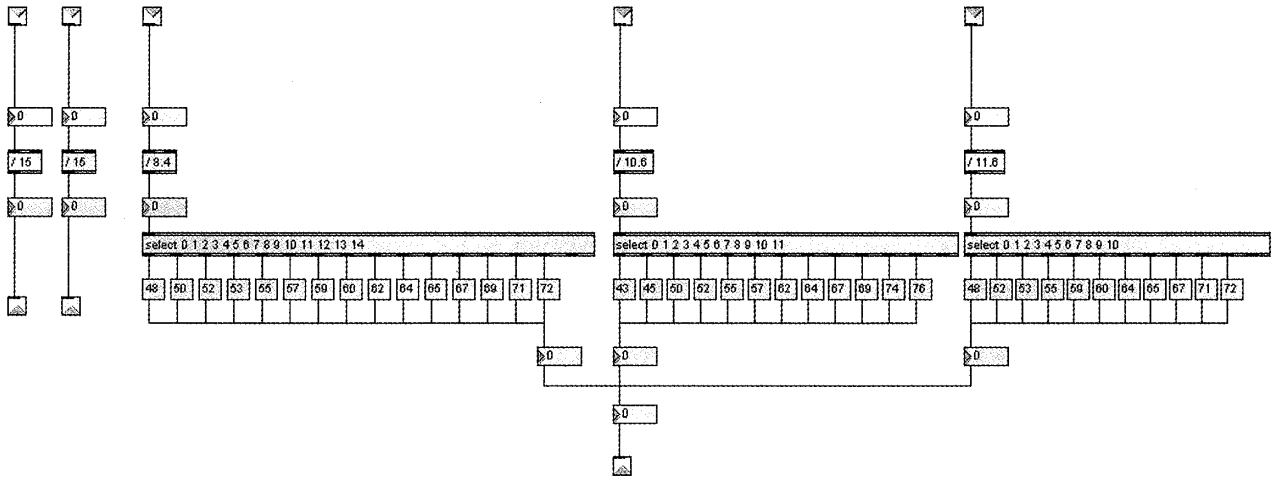


図27 p\_pcr\_to\_hypersonic オブジェクトの全体

### 5.3.3 ゲートのパラメータの値の変換

入力されたコントロール値をゲートと呼ばれる発音のタイミングを、指定したBPMをもとに全音符、二分音符、三分、四分、八分、十六分、二十四分、三十二分と8段階に変換させるための命令を行うメッセージに変換した。このことにより、発音のタイミングが決定される。段階選択を演奏者が任意に切り替え可能とするために、PCR-A30のコントロール番号6のノブを1つ割り当てこの番号をctlin オブジェクトに送った。

コントロール番号6のノブから取得出来るコントロール値は7bit、128段階で可変するが、目的のゲート操作には8段階の可変を要する。つまり128から8段階への可変域のマッピングを施す必要がある。これを行うために p\_pcr\_to\_hypersonic オブジェクト内で変換過程を行っている。0~127の128段階から8段階の可変域の変換過程については“ $127 \div 8$ ”で出た値の整数値で入力値を割ることで導き出せる。ゲートの変換に関して p\_pcr\_to\_hypersonic オブジェクト内ではこのような処理を行っている。そして、p\_pcr\_to\_hypersonic オブジェクトから、1~8の値が出力され、そのそれぞれの値を1→1、2→2、3→3、4→4、5→8、6→16、7→24、8→32のように変換し、ゲートの操作に割り当てた。(図28)。

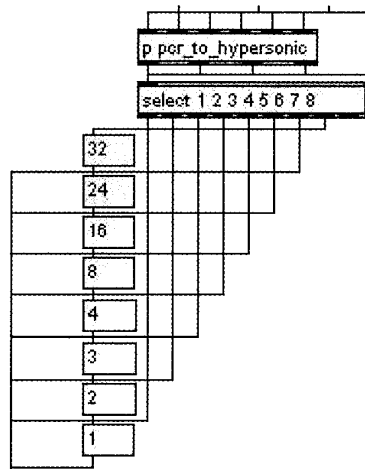


図28 ゲートの値を変換する処理

### 5.3.4 デュレーションのパラメータの変換

発音タイミング以外に、発音のコントロールを担うパラメータとして、デュレーションと呼ばれるパラメータが存在する。このパラメータは音の長さを調節するためのパラメータであり、指定した時間の長さ分だけ音を発音させる命令を音源に対して行う。デュレーションとは MIDI のメッセージではなく、厳密には、音を鳴らす MIDI メッセージであるノート・オンと言われるメッセージが与えられてから、音を止める MIDI メッセージであるノート・オフと言われるメッセージが与えられるまでの時間の長さを指す[5]。通常このデュレーションと言われるパラメータは、従来の楽器型インターフェースによる演奏時には、あまり意識する必要のないものである。例えば、鍵盤型のインターフェースでは、鍵盤を打鍵したときにノート・オンが生成され、鍵盤を離したときにノート・オフが生成されるように設定されている。このように、従来の楽器型インターフェースには発音のための一連の動作の中で、ノート・オンとノート・オフが生成されるように設計されているため、そのような楽器型インターフェースを用いる場合は、デュレーションに対して意識する必要がないが、本システムのような非楽器型インターフェースによって MIDI の制御を行う場合には、このデュレーション値を別に制御するようなシステムが必要になる。そこで、今回はこのデュレーション値の制御を前述のゲートのパラメータのようにノブによって音高のパラメータとは別系統としてコントロールするシステムを採用した(図29)。

仕組みとしては、ゲートの制御同様にノブによって音の長さを8段階で可変する内容になっており、これによって、音の長さを歯切れよいものから長尺なものまで自在に操ることが可能になっている。

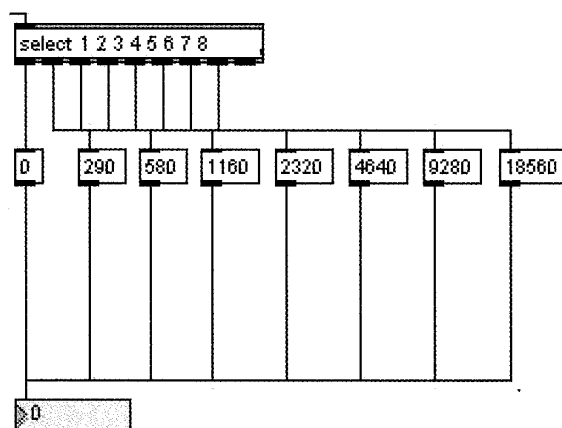


図29 デュレーションの値を変換する処理

### 5.3.5 スケールマッピング

音高操作のパラメータとしてノートナンバーの値を利用している。このノートナンバーのパラメータの操作にはフェーダを使用しており、1つのフェーダによって一定間隔で発音される音の音高を可変させる内容になっている。この機構を考えるにあたって、ノートナンバーの並びをどのような順序で割り当てれば、発音される音の連続がより音楽的な内容になりうるかということを試行錯誤して検証する必要があった。はじめに、単純に1オクターブ分の音程を連続的に並べて割り当てたものを試作したところ、音楽的に合わない音程、いわゆるスケールか

ら外れた音をしばしば選択して発音してしまう状態になった。そこで、音階(スケール)に則って音高を配する、いわゆるスケールマッピングを施すことで発音される音の連続を音楽的な内容に出来るのではと考え、ハ長調や呂音階、琉球音階などを試してみたところ、比較的、音楽的に成立した内容を保ち、かつ、それぞれの音階が持つ独自の雰囲気や旋律を生成することが出来た。システムを構築するプロセスとしては、コントロールに用いるフェーダを選び、一つのフェーダにスケールを一つ毎に配していくという方法を用いた、例えば、ハ長調音階の場合、使用するフェーダのコントロール番号を指定した `ctlin` オブジェクトからコントロール値を取得し、`per_to_hypersonic` オブジェクト内で0~127である可変域を必要なノートナンバー上の範囲に変換を行う。この場合、ハ長調音階の C2から C4までの2オクターブ分を範囲に設定することで、可変域は15段階となり、128段階から15段階への変換は、ゲートのパラメータ値の変換同様にして" $127 \div 15$ " で出た値の整数値で入力値を割ることで行い、結果1~15の値が出力されるため、このそれぞれの値を1→48、2→50、3→52、4→53、5→55、6→57、7→59、8→60…以下省略…15→72のように変換しノートナンバーの値に割り当てることでスケールマッピングを行うことができる(図30)。

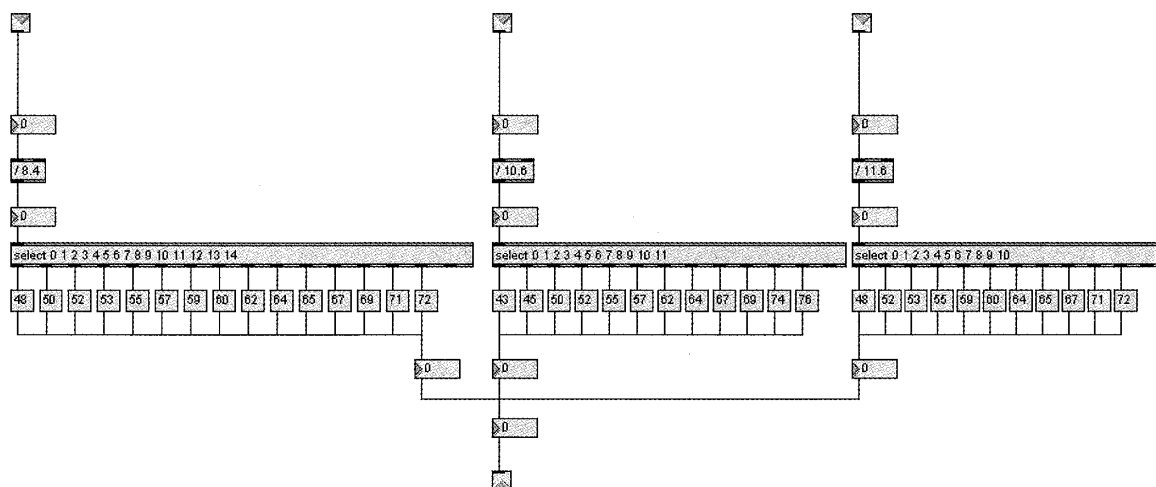


図30 スケールのマッピングを行う処理

### 5.3.6 ハーモニーを生成する処理

発音される音に対して複数音を付加し和音での発音を行うことで、音に厚みを増す変化を加えることが出来ると考え、フェーダで操作する音を基にコード(和音)を生成するパッチを作り、状況に応じて和音で演奏することも試みた。和音とは原理上、複数の異なる音高の音の集合が同時に響くことで得られ、音高の組み合わせだけ響き方も存在し、根音となる音を基準にそれ以外の音との音高上の相対的距離の組み合わせによってコードネームという分類がされている。この処理においても、付加する音のノートナンバーを、基になる音のノートナンバーの値に対する相対的な距離によって表記することで指定し、和音の生成を行っている。(図31)

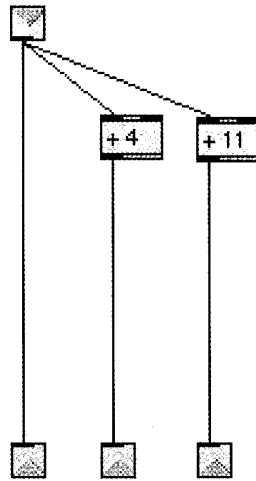


図31 ハーモニーを生成する処理

### 5.3.7 ソフトウェア音源への接続過程

前述した入力装置からのパラメータ処理過程の全てを統合し、ソフトウェア音源の Hypersonic に入力する MIDI データを生成するために makenote オブジェクトを用いた (図 32)。makenote オブジェクトは、ノート・ナンバー、ベロシティ(音量)、デュレーションの 3 要素を受け取り、ノート・オン、ノート・オフを生成するオブジェクトである。この処理の後、Hypersonic の処理に必要なヘッダー情報を付加し、変換した MIDI 情報を Max/MSP 内で VST ソフトウェア音源の使用を可能にする vst~オブジェクトに入力することで変換した MIDI 情報を基に Hypersonic で演奏することが可能になる。(図33)

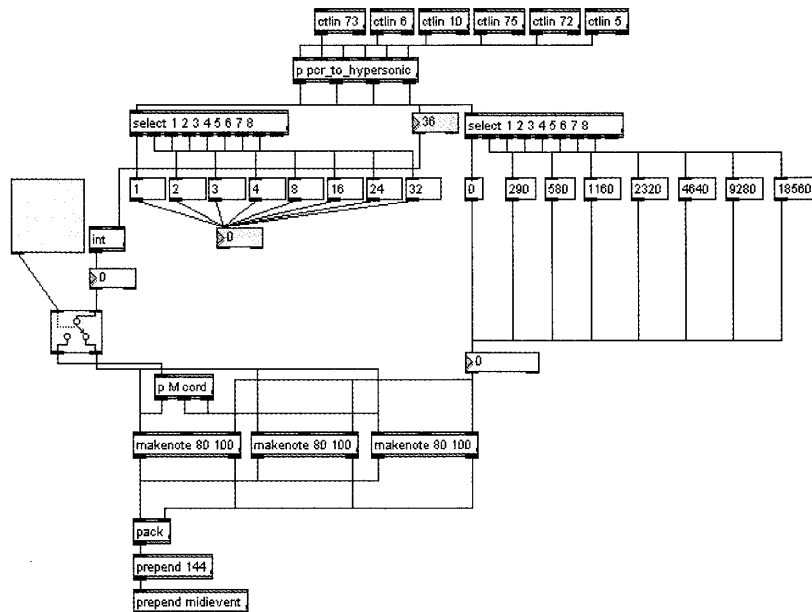


図32 MIDI 情報を生成するパッチの全体図

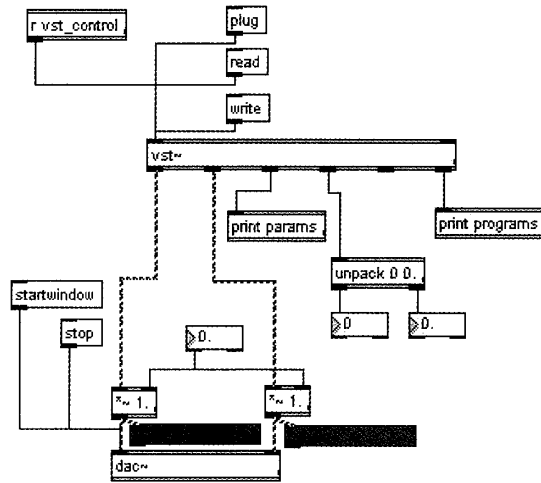


図33 VST 音源の制御を行う処理

## 5.4 演奏

### 5.4.1 演奏スタイル

音価値とゲートタイムを操作するノブを左手側、音高を操作するスライダを右手側にレイアウトした。そして、それぞれの操作子を両手で同時に操作を行う。感覚をつかむための多少の試行は必要であるが操作自体は簡便であるため、演奏は数分で行うことが出来る。スライダを上下させる動作と、ノブを左右に回す動作を数回行うことにより、音源から持続的に音が発音されるので、一旦リズムをつかむことが出来れば、何らかのフレーズを展開することが出来る。(図34)



図34 演奏スタイル

### 5.4.2 演奏情報

本システムを用いた演奏を実際にどのような演奏情報が出力されているかを視覚的に確認するために MIDI レコーディングプログラムとスタンダード MIDI ファイルを作成し、シーケンスソフトによって視認出来るようにした。類似フレーズをハ長調音階、呂音階、琉球音階のスケールにマッピングし、ピアノロール表示を行った。(図35)

本来ならば、全く同じフレーズの方がより比較対照としてわかりやすいが、今回はシステム側に静的な MIDI ファイルを読み込ませる機能を付加していないため、各音階についてリアルタイムで演奏を行っている。よって、各音階の演奏に若干のずれが生じていることを断っておく。

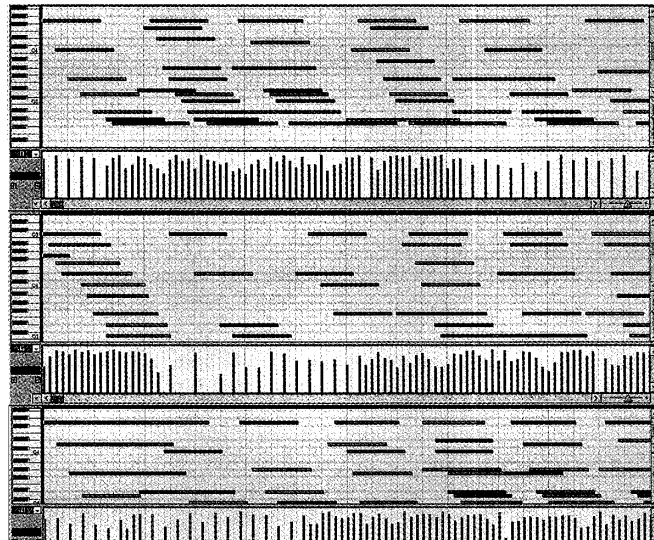


図35 ピアノロール表示 (上：ハ長調音階 中：呂音階 下：琉球音階)

次に、ハ長調音階を基にした基本フレーズにハーモニーを生成する機能を用いて演奏情報を MIDI レコーディングした (図36)。単純な生成規則であるが、付加した状態としていない状態では聞こえに顕著な差が生じた。

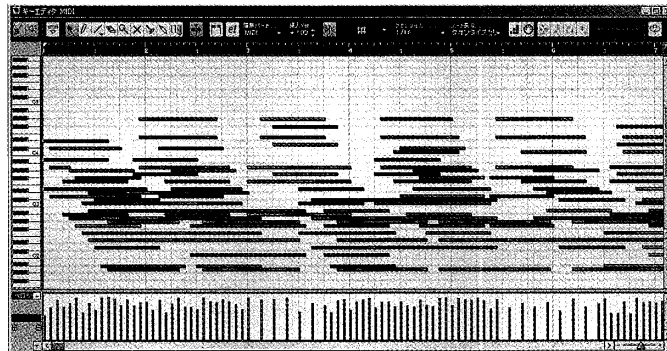


図36 メインフレーズにハーモナイズされた MIDI 情報のピアノロール表示

さらにこの演奏情報をピアノロールではなく、スコアとして表示を行った (図37)。

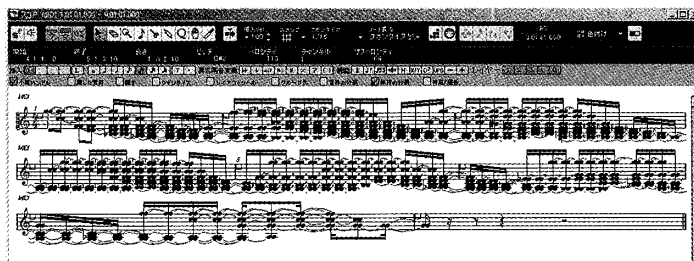


図37 スコア表示

## 6. 今後の課題

このように楽器としての演奏機能をもったプロトタイプは完成した。不完全な箇所や改善点は多いが、基本コンセプトを具体化することは出来た。

当該システムの想定利用者として、音楽の専門教育を受けていない小中高生や音楽的素養が平均程度の一般的な人を前提としているが、そのような想定利用者による性能評価および機能向上が今後の課題である。

そして、将来的にはリトミックといった情操教育や芸術教育などの教育現場への実践導入や、映像情報処理過程を加えてメディア・アートとして発展させていきたいと考えている。

## 参考文献

- [1] Curtis Road:コンピュータ音楽 歴史・テクノロジー・アート、東京電機大学出版局、2001
- [2] 高橋信之：コンプリート MIDI ブック、リットーミュージック、2003
- [3] ノイマンピアノ(赤松正行+佐近田展康)：トランス Max エクスプレス、リットーミュージック、2001
- [4] ノイマンピアノ(赤松正行+佐近田展康)：2061Max オデッセイ、リットーミュージック、2007
- [5] カメオインタラクティブ:The Max Book、カメオインタラクティブ、2004