

## シーズ指向による新楽器のスケッチング

長嶋洋一†

新しい楽器をコンセプト(ニーズ指向)からでなく、材料/道具(外見)の活用というシーズ指向の視点からデザインする、「フィジカル・コンピューティング」(インタラクションの実機プロトタイピング)の事例紹介を行う。開発プラットフォームに並列処理プロセッサを用いることで、個々のタスクを増設・記述するだけで、タイミング設計なしに、時分割処理による高いパフォーマンスを実現できた。

### Sketching for new musical instrument with seeds-oriented development

Yoichi Nagashima†

This is a report of case study in design for new musical instruments. It is also an introduction of "Physical Computing" (Sketching - real prototyping for interactions of systems). I used a parallel processing CPU (Propeller), so it was very easy to construct multi-task system without critical design of timing.

† 静岡文化芸術大学  
Shizuoka University of Art and Culture

## 1. はじめに

筆者はこれまで、ライブ・コンピュータ音楽の領域において、マイクロエレクトロニクス技術やセンサ技術などを活用して、伝統楽器を改造したり新しいセンサを活用した「新楽器」を多数、製作してきた[1]。プロジェクトの形態には大きく2種類あり、筆者が作曲の一部として自分のComputer Music作品の公演に使うための新楽器を製作する場合と、作曲家等の依頼によりシステム開発者の役割で新楽器を製作する場合とがあった。後者では依頼者である作曲家は電子情報技術の専門家ではないために、ある音楽的表現や舞台表現のためにどのようなインターフェース/技術が欲しいのか、という具体的な戦略が指定されない場合が多く、まず最初に「何をしたいか」を取材/議論してから、センサを選定したり電子回路を試作したりソフトウェアを開発する、という流れとなった。これはいわば「ニーズ指向」のシステム開発と言える。

これに対して、造形部分が既にあったり、舞台表現の都合(舞台装置やパフォーマーの衣装など)により、センサやシステムを組み込む対象が最初から決まっていたり、伝統楽器の演奏法(ジェスチャ)を用いるために伝統楽器と似た構造/機構/フォルムにしたい、などの要請がある場合には、システムを実現する要素技術の選択や開発においての制約が先になり、いわば「シーズ指向」のシステム開発となるアプローチも存在する。本稿では、この後者、シーズ指向による新楽器の実験・試作・開発に重点を置いて、実機プロトタイピングの新しいデザイン手法である「物理コンピューティング(スケッチング)」の活用という視点から検討した。

## 2. ニーズ指向の新楽器の制作事例

### 2.1 国際会議NIMEでの事例(1)

筆者は、NIMEと呼ばれる「音楽/芸術表現のための新インターフェース」国際会議[2]の組織委員会メンバーであり、2004年にはNIME04を日本で開催した[3]が、ここでは毎年のように新しい楽器の開発報告がある。一例として文献[4-8]で紹介した事例では、それぞれ音楽的な要請から、本稿で言うニーズ指向的なアプローチでの開発の好例となっている。重要なのは、プロジェクトが工学寄りのメンバーによって自己満足的に企画されているのではなくて、作曲家・演奏家などとのコラボレーションにより、音楽的な目的意識(公演)を明確にしているところであり、新楽器を活用した作品の発表においては、技術デモに終わらない、音楽的に高度なコンサートとして充実している。

## 2.2 筆者(ASL)のこれまでの事例(1)

筆者の開発プロジェクトにおいてニーズ指向のアプローチをとる場合、音楽サイドとしてでなく敢えて技術者として、依頼者である作曲家・演奏家に協力する、というケースがほとんどである。作曲家・中村滋延氏の作品公演のための開発[9]においては、ステージ上での演奏家のパフォーマンスの色々なジェスチャー（オブジェを叩く、空間を手刀で斬る、身体を叩く、箱を持ち上げる、床を踏みつける、等）検出のためのセンサ群としての新楽器を作った。作曲家/演奏家の東野珠実氏のための開発[10]においては、伝統楽器「笙」を演奏する呼気と吸気の情報を通常の演奏動作に影響なく検出する、という音楽的な要請からセンサの選択/実験を開始した。また筆者自身が作曲の一部として新作のための新楽器を制作する場合にも、作品のテーマやパフォーマンスの要請という視点からニーズ指向的なアプローチをとった事がある[11]。

## 3. シーズ指向の新楽器の制作事例

### 3.1 国際会議NIMEでの事例(2)

「音楽/芸術表現のための新インターフェース」国際会議NIMEにおいては、ユビキタスコンピューティングやモバイルコンピューティングなどの工学的なセンサ技術を応用した新楽器・新インターフェースの開発報告が、世界中の研究機関や大学から行われている。図1および図2の写真は、NIME06(パリ)でのコンサートの模様であるが、明らかに「形から入る」新楽器デザインの好例である。本稿ではいちいち紹介する紙面がないので、NIMEのサイトにある過去のオンラインProceedingsを参照されたい [12]。



図 1 NIME06での演奏風景より(1)

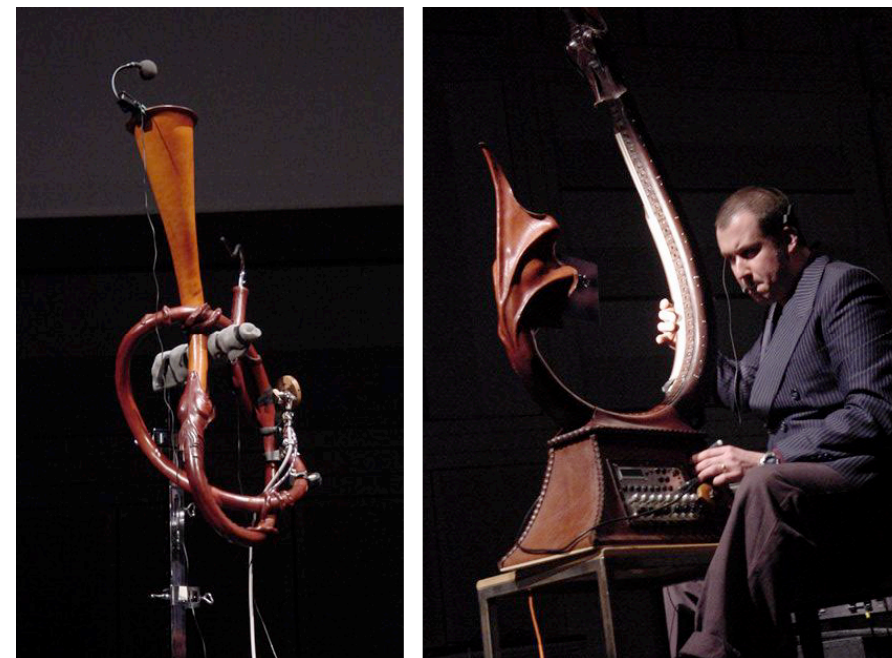


図 2 NIME06での演奏風景より(2)

### 3.2 筆者(ASL)のこれまでの事例(2)

筆者の開発プロジェクトにおいてシーズ指向のアプローチをとる場合、多くは特定の作品創作公演に向けてというよりも、日常的なシステム実験からスタートする。日進月歩のエレクトロニクス技術の領域では、過去には不可能とされたデバイス/技術が新たに登場したり、過去には高額で使用困難であったデバイスが民生機器市場によるコスト低減効果で活用できる事例に事欠かない。後者の例としては、3次元加速度センサやジャイロセンサは、筆者が開発した事例においても、部品コストとして10年間で2桁から3桁の低価格化を経験している。プラスチック光ファイバセンサを用いた新楽器「光の絃」や静電タッチセンサを用いたインターフェースの場合には、インストールのインターフェースとして、あるいは音楽作品の楽器として活用した[13]。

アナログ回路技術の専門家・照岡氏とのコラボレーションによって進めた筋電センサの実験研究においては、3世代にわたる“MiniBioMuse”と名付けた筋電センサを開発し[14]、これを楽器（身体表現のセンサ）として活用して作曲した作品群は、国内だけでなくフランス・ドイツ・オランダ・カナダ・台湾などでも公演してきた。

シーズ指向でのもう一つの新楽器の制作として、材料となる「何か」が先にある、という例も少なくない。対象は初代ファミコン用のPowerGloveやインド民芸品のおもちゃのエークタール（1弦楽器）だったり[15]、北京のお土産品の琵琶だったり[16]するが、ワンボードマイコンを活用することで、それぞれのフォルムや形状を生かしたスタンドアロンの楽器とすることができる[17-18]。インドネシアの民族楽器Kendangを材料として、「楽器が(呼吸するように)光ってもいいじゃないか」というテーマで開発した新楽器“CyberKendang” (図3)は、太鼓としてのアコースティック特性に影響しないように一切の穴あけや接着を行わずに製作した。この楽器においては自発的に光るスタンドアロン動作を求めたために、コンピュータにMIDIを送る機能さえも省略して、太鼓演奏のサウンド（音量）に反応して光のパターンと輝度が変化する。この楽器を活用した作品はNIME07に採択されてニューヨークで公演したが、作曲を開始したのは楽器が完成してからであり、典型的なシーズ指向による新楽器開発の事例となった。



図 3 新楽器“CyberKendang”

## 4. 新楽器の制作における課題とスケッチング

### 4.1 新楽器の開発プロジェクトの概要

これまで述べたように、メディアアートに色々な形態があるのと同様に、新楽器といってもそれぞれの開発プロジェクトは色々なアプローチで進められる。しかし技術的に見れば、共通の特性を見いだすこともでき、これはメディアアート全般の「開発プラットフォーム」に向けた汎用化の検討の第一歩となる。特にシーズ指向の新楽器の開発においては、一般的に以下のようなプロセスをとる。

- まず先にシーズ（材料、センサ、改造する自然楽器、部品etc）がある
- このシーズを組み込んだ試作開発機がそのまま最終的な実機となる事が多い
- 組み込みマイコンやセンサはソケットでなく直接実装されるが多い

一般のシステムハウスなどが開発製造する特注組み込みシステムとの違いとして、基板やケースのサイズ、さらに内部配線などの実装上の空間的制約が厳しいために、基板上に組み込みマイコンボードのソケットを置いて差し替えることは出来なかったり、センサを交換するソケットやコネクタが演奏時の接続不良の原因となるので直接ハンダ付け/取り付けを行うことが多い。量産を前提としない一品料理であるからこそ可能な方法であるとも言える。

### 4.2 新楽器の開発における問題点

このような新楽器の開発において、筆者が文献[15][17][18]などで指摘しているのは、スタンドアロンシステムの中核となるボードマイコン（AKI-H8やPICやArduino）のファームウェアを書き込む内蔵Flashメモリの「繰り返し書き込み」の問題点である。昔であれば組み込みマイコンのプログラムはシステム内に紫外線消去型EPROMのソケットがあり、ここに最終的なプログラムを搭載したが、ここ10年以上の主流は、CPU内部のROM領域がFlashEEPROMになっていて、CPUのシリアルポートを経由してここにプログラムを書き込む、という方式である。図4および図5は、筆者が秋月電子のAKI-H8を使って開発している風景の一例であるが、開発中の実機に搭載されたAKI-H8のRS232CポートからROMプログラマボードとUSB-RS232Cアダプタを経て、開発ホストであるWindowsパソコンと接続している。このようにオンボードマイコンに直接、開発中のプログラムをダウンロードできることで、システム開発中のEPROMソケットの抜き差しや「ROMエミュレータ」のソケット接続の手間から解放された [21]。

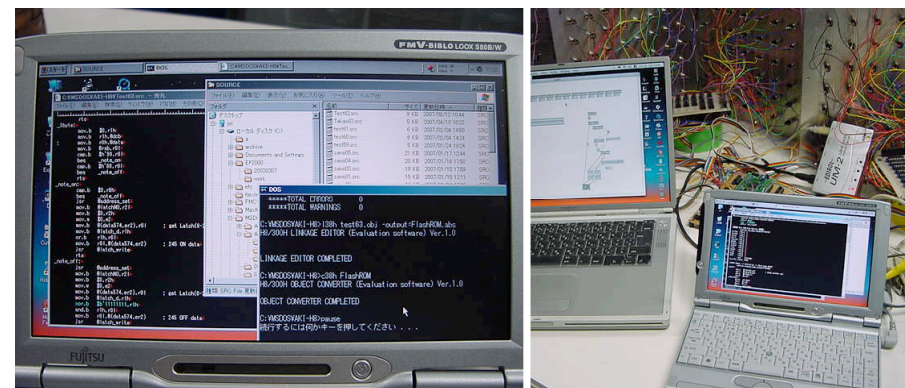


図 4 AKI-H8による開発風景(1)



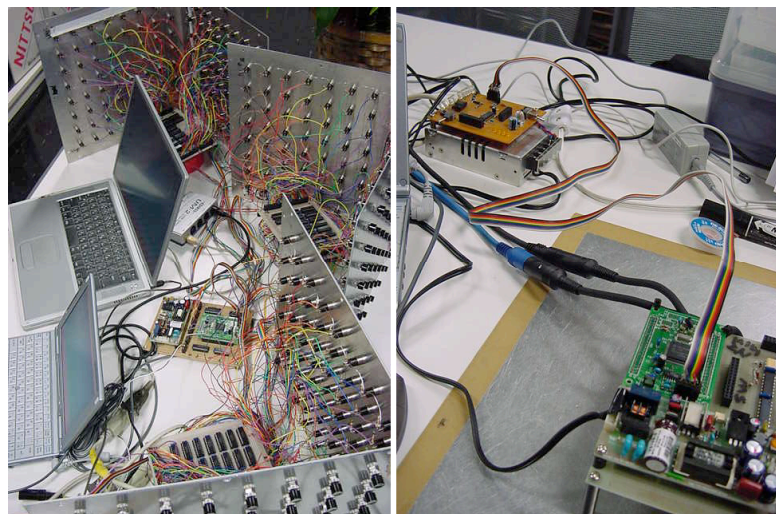


図 5 AKI-H8による開発風景(2)

ここで問題となるのは、カタログデータではCPUの内蔵FlashEEPROMの繰り返し書き込み回数は10万回以上とされているものの、CPU個別のばらつきや周辺ノイズ環境などにより、実際には100回程度のプログラム書き込みでもエラーが起きることがあった、という経験的事実である。ICE(インサーキットエミュレータ)などを使わずに、開発プログラムをシンプルなものから漸次拡張していき、EEPROMに書き込みしてはリセットしてデバッグする、という開発プロセスにおいて、完成度を上げるためには何度でも繰り返すことが必須である。ところが「書き込み回数が増えるとエラーが起きることがある」とか「もう書き込めない(使えなくなる)段階がいずれ迫ってくる」という状況は、プログラマにとって水面下でのプレッシャー/ストレスとなる。実際に筆者の経験でも、ある程度まで完成したプログラムについては、そこから繰り返し細かく改訂するというよりも、システムの使い方に対応するとか、MIDIを経由してやりとりするホストシステム(Max/MSPなど)の側のプログラミングによって細かい改良の部分を吸収する、という戦略は、ほとんど無意識的に採用している「常識」である。

### 4.3 新楽器のスケッチングにおける課題

このような、特製/試作システムを実機で開発/プロトタイプングするデザイン手法として、近年「スケッチング」という名称で大きな流れが生まれてきている。その背景には、ArduinoやGAINERに代表される新世代の高性能ボードマイコンの登場(ハードウェア)と、オープンソースによるフリー開発環境の普及(ソフトウェア)と、ロボット技術

の流行から各種のセンサやアクチュエータが出回っている環境があると思われる[22]。音楽情報科学の領域ではNIMEのように1980年代から一つの主流であった新楽器の制作が、ここにきて一般のsketcherにも拡張している、とも言えよう。

楽器に限った話ではないが、一般にこのようなスタンドアロンシステムでは、以下のように多数のタスク(ジョブ)が必要になってくる。

- パネルスイッチや外部センサからの入力
- パネルLED/LCDやモータ制御などの出力(タイミング制御)
- 開発ホストPCとの通信(開発中のみ)
- システムホストとの通信(MIDI/イーサネット/無線)
- デバッグ用の動作表示、数値表示など

最終的には実機で必要なくなる機能であっても、デバッグの過程ではわざわざハードウェアを試作増設して、ファームウェアの動作確認、数値データなどの外部出力による動作モニタリングは、スケッチングにおいて開発期間の短縮と完成度の向上に大きく寄与するので、ちょっとした回り道であっても定番のテクニックである。筆者の場合には、もう18年ほど使っている“Max/MSP/jitter”を作品全体のホストとすることが大部分であるために、どんなシステムであってもまずはMIDI入出力ポートを設置して、開発やデバッグの際には、システムの内部状態をMIDI出力したり、MaxからのMIDIを受けて動作切り替えなどを行う。最終的にMIDIとは無縁のスタンドアロンシステムの開発においても、である。

ここで問題となるのは、MIDI入力には割り込みが必要であり、またタイミング設計のためにCPU内部のタイマによるタイマ割り込みを使うことも多いことである。ポーリングとかハンドシェイクのようなシンプルなジョブの積み重ねだけで必要な処理を実現できれば問題ないが、小型軽量低価格を目指す組み込みマイコンではクロック周波数やメモリ(RAM)容量に限界があり、Cだけでなく時にアセンブラも組み合わせた、本格的なプログラミングスキルを要求することも少なくない。楽器は本質的にリアルタイム・マルチタスクシステムだからである。

ここに、前述のCPUプログラム書き込み回数の制限という要因が組み合わされることで、事態は非常に困難な状況となる。つまり、仕様としての機能と開発/デバッグ用の機能をシステムに盛り込むことは、タイミングも精度も異なる複数のタスクを、割り込みを必要とするレベルで調停しつつリアルタイムに実現するという要請であり、転送一発でそう簡単には実現できない。デバッグ資源/機能を活用しつつ、何度もトライすることで完成度を上げていく。ところが、このトライ回数に暗黙の上限が制約として加わることで、開発そのものの足を引っ張るという二律背反の状況となるのである。この世界になかなか若手エンジニアが入ってこれない理由の一つは、ベテランなら経験的に分かっているこのあたりの勘所に不慣れなためであろう。

## 5. 並列処理プロセッサによるシステム開発事例

### 5.1 Propellerプロセッサ

文献[23-25]で紹介したように、筆者は2008年に、BasicStampで有名なParallax社の提供するPropellerプロセッサと出会い、その概念や細部を解析するとともに、いくつかの実験システムを試作し、さらにインタラクティブなインストール作品を実際に制作して公開展示した[26]。このシステム“DodecaPropeller”では、Propellerチップを13個用いて、縦3台・横4台、計12台のビデオモニタにそれぞれ個別のリアルタイムCGを生成表示し、その12個の画面が来場者からの働きかけや会場の環境音に反応してダイナミックに変化するが、従来のようにホストPCを必要とせず、たった1枚の基板(図6)でシステムが完結している。

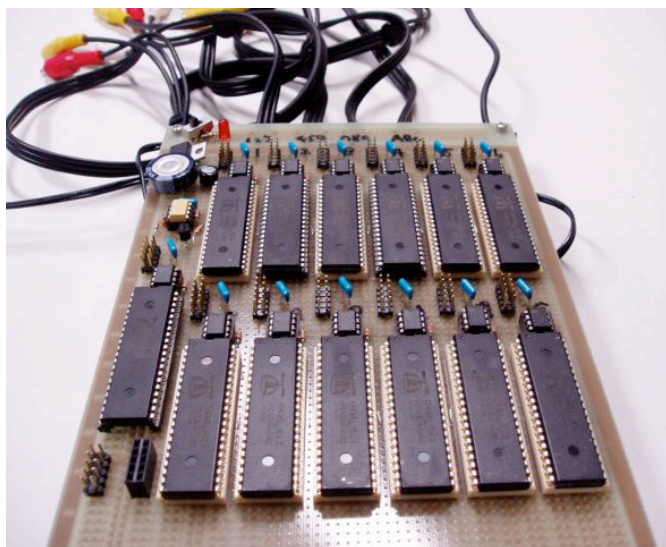


図 6 “Dodeca Propeller”のボード全景

### 5.2 Propellerを活用したスケッチング

このPropellerチップは内部の8個のCPUがハードウェア的に並列処理を実行するために、ソフトウェア開発において「割り込み」の概念が不要である。Cogと呼ばれる各CPUのタスクを起動すれば、相互のタイミングを設計しなくても、各Cogはそれぞれの処理を中断なく実行する。これは新楽器のように、本質的にマルチタスクなシステム開発(スケッチング)においては、非常に有効なプラットフォームである。

さらにPropellerでは、開発用ホストPC(高級言語spin/アセンブラを共存可能)からUSB接続されて開発中のプログラムを書き込むが、そのターゲットとしてPropellerの内部RAMを指定できる。つまり開発中はPropellerの内部RAMに転送したプログラムで実験/デバッグし、最終的な完成プログラムについては、ターゲットをPropellerチップの隣に置いたFlashEEPROMに変更するだけでよい。リセット時にPropellerはまずホストPCとの接続を捜し、接続があれば開発中として内部RAMのプログラムで走る。接続されていなければ外部EEPROMから内部RAMに自動転送したプログラムで走る。この機能の意味するところは非常に重要で、RAMであるから開発中は「何度でも無限にリトライ出来る」マイコン、という事である。細かい実験や試行錯誤や修正はRAMで行い、ある段階まで完成したところでソースを保管するなり外部EEPROMに書き込む、というプロセスにより、従来のマイコンとは桁違い(無尽蔵)のテスト繰り返しを可能とする。

筆者はPropellerで外部とMIDI通信するモジュールをオリジナル開発し、Max/MSPと合わせた開発環境を実現した[23-25]が、さらにPropellerでは内部のCogのうち2個を使い、40ピンの入出力ピンのうち3本の外部に3本の抵抗を接続するだけで、NTSC/PALビデオ信号を簡単に生成出力することができる。PropellerコミュニティによりWebフリー公開されているグラフィックドライバなどをブラックボックスとして呼び出すだけで、デバッグのために「ビデオモニタに10進/16進形式での数値表示」やOpen-GLにも似た「カラーグラフィック表示」も容易に実現できる。これは、システムとして必要なマルチタスクを実機として動作させながら開発する中で、デバッグのために簡単に内部動作をモニタできる、いわば「内蔵デバッグ装置」のように活用できる。

### 5.3 現在進行中の新楽器スケッチング

新楽器というのは登場した時のインパクトも命なので本稿ではまだ写真を公開しないが、筆者は新たなアプローチとして、2009年にPropellerを活用した新楽器の開発をスタートさせた。まさにシーズ指向の開発であり、今回の対象はなんと「ケース」である。音楽情報科学研究会のベテラン・松島俊明氏(東邦大)から何年か前に譲り受けた、往年の名機・NeXTcubeコンピュータの筐体(図7)を眺めているうちに、これをアナログ風味の新楽器に進化させてみたい、というイメージが湧いてきたのである。

ケースでは何も出来ないので、秋葉原や日本橋のジャンク屋を巡り、最終形を想定せずに、意識的にパーツ(シーズ)としての存在感を求めて、大きなアナログメーター、ボリュームツマミ、パーニヤダイヤル、ロッドアンテナ、シート状キーボード、放熱用フィン(+ステッピングモータ)、カメラ温度計測機、など多数を仕入れた。分厚いマグネシウム製ケースにチタン刃ドリルで穴開けし、ばらばらのパーツを現物合わせで取り付け、PropellerチップのCogをデバッグ用ビデオ出力とともにCR時定数型A/DやPWM型D/Aとして利用し、各部分が少しずつ動き始めた、というのが本稿執筆時点での状況である。まさに試行錯誤の開発に好適なプラットフォームであると実感している。





図 7 NeXT cube

## 6. おわりに

新楽器の製作において、ニーズ指向とシーズ指向のアプローチによる事例紹介・検討とともに、材料/道具(外見)の活用というシーズ指向の視点からのデザイン「フィジカル・コンピューティング」(インタラクションの実機プロトタイプ)の事例紹介を行った。開発プラットフォームに並列処理プロセッサを用いることで、個々のタスクを増設・記述するだけで、タイミング設計なしに、時分割処理による高いパフォーマンスを実現できた。今後、さらにメディアアートのシステム開発のための汎用プラットフォーム化という視点でのライブラリ整備を進めて、より多くの事例を重ねつつその可能性を検討していきたい。

## 参考文献

- 1) Art & Science Laboratory <http://nagasm.org>
- 2) 長嶋洋一: 音楽/芸術表現のための新インターフェース(NIME), ヒューマンインタフェースシンポジウム2006論文集, ヒューマンインタフェース学会 (2006).
- 3) 長嶋洋一: NIME04/MAF2004開催報告, 情報処理学会研究報告 Vol. 2004, No. 111 (2004-MUS-57), 情報処理学会 (2004).
- 4) Cormac Cannon, Stephen Hughes, Síle Ó Modhráin: EpiE - Exploration of the Uilleann Pipes as a Potential Controller for Computer-based Music, Proceedings of the 2003 International Conference on New Interfaces for Musical Expression (2003).
- 5) Eric Singer, Kevin Larke, David Bianciardi: LEMUR GuitarBot - MIDI Robotic String Instrument, Proceedings of the 2003 International Conference on New Interfaces for Musical Expression (2003).
- 6) Dan Overholt: The Overtone Violin, Proceedings of the 2005 International Conference

on New Interfaces for Musical Expression (2005).

- 7) Juan Pablo Cáceres, Gautham J. Mysore, Jeffrey Treviño: SCUBA - The Self-Contained Unified Bass Augmenter, Proceedings of the 2005 International Conference on New Interfaces for Musical Expression (2005).
- 8) Frederic Bevilacqua, Emmanuel Flety, et al: The augmented violin project - research, composition and performance report, Proceedings of the 2006 International Conference on New Interfaces for Musical Expression (2006).
- 9) 長嶋洋一: 音群技法による音楽作品のための演奏支援システム, 平成2年度後期全国大会講演論文集I, 情報処理学会 (1990).
- 10) Yoichi Nagashima: "It's SHO time" - An Interactive Environment for SHO(Sheng) Performance, Proceedings of 1999 International Computer Music Conference, International Computer Music Association (1999).
- 11) 長嶋洋一: インタラクティブ・メディアアートのためのヒューマンインターフェース技術造形, 静岡文化芸術大学紀要・第1号2000年, 静岡文化芸術大学(2001) <http://1106.suac.net/news/docs/suac2000.pdf>
- 12) NIME <http://nime.org>
- 13) 生体センサによる音楽表現の拡大と演奏表現の支援について <http://nagasm.suac.net/ASL/sensor03/>
- 14) 新・筋電センサ "MiniBioMuse-III" <http://nagasm.suac.net/ASL/SIGMUS0108/>
- 15) センサを利用したメディア・アートとインスタレーションの創作 <http://nagasm.suac.net/ASL/sensor01/>
- 16) Hyper Pipa <http://nagasm.suac.net/ASL/pipa/>
- 17) オリジナルMIDIセンサを作ろう <http://nagasm.suac.net/ASL/original/>
- 18) 長嶋洋一: 作るサウンドエレクトロニクス <http://nagasm.suac.net/ASL/mse/index.html>
- 19) Yoichi Nagashima: GHI project and "Cyber Kendang", Proceedings of the 2007 International Conference on New Interfaces for Musical Expression (2007).
- 20) Yoichi Nagashima: GHI Project - New Approach for Musical Instrument, Proceedings of 2007 International Computer Music Conference, International Computer Music Association (2007).
- 21) <http://1106.suac.net/news2/20070204/> <http://1106.suac.net/news2/20071202/> <http://1106.suac.net/news2/20071208/> <http://1106.suac.net/news2/20080705/> <http://1106.suac.net/news2/20070107/> <http://1106.suac.net/news2/20070124/>
- 22) 長嶋洋一: フィジカル・コンピューティングとメディアアート/音楽情報科学, 情報処理学会研究報告 Vol. 2008, No. 89 (2008-MUS-77), 情報処理学会 (2008).
- 23) 長嶋洋一: 並列処理プロセッサを活用したメディアアートのための汎用インターフェース, 情報処理学会研究報告 Vol. 2008, No. 78 (2008-MUS-76), 情報処理学会 (2008).
- 24) Propeller日記 <http://nagasm.suac.net/ASL/Propeller/>
- 25) 長嶋洋一: Propellerを使った体験型アート作品の製作(前編/後編), トランジスタ技術 2008年9月号/10月号, CQ出版社 (2008).
- 26) MAF2008 <http://1106.suac.net/MAF2008/>