

並列処理プロセッサ "Propeller" によるプラットフォームの検討

長嶋洋一†

新しい楽器をデザインする「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）において、8個の並列処理CPUを内蔵する"Propeller"プロセッサを活用した事例紹介を行う。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る。

Platform with the "Propeller" parallel processor

Yoichi Nagashima†

This is a report of physical computing to which people recently pay attention in design process field and media-arts field. I introduce and discuss the special approach with the "Propeller" parallel processor.

† 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[1-7]。また、アイデアを柔軟に検討したいデザインプロセスにおいて、あるいはシステムを実現する要素技術や開発の制約が先にある「シーズ指向」のシステム開発で有効な、実機プロトタイピングの新しいデザイン手法「物理コンピューティング(スケッチング)」の活用注目し、特にParallax社の組み込み用並列処理(8CPU内蔵)"Propeller"プロセッサについて研究している[8-12]。本稿では、本質的にマルチタスク処理の必要な、「新楽器」「映像と音響とインタラクションのあるメディアアート」などを実現するプラットフォームとして、"Propeller"プロセッサを活用するための実験・検討について紹介する。

2. 並列処理プロセッサによるシステム開発環境

2.1 Propellerプロセッサ

文献[8][11]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た。Propeller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、第一の特長として、ソフトウェア開発における「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。

また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接続して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の時だけ、ターゲットをPropellerチップの隣に置いたFlashEEPROMに変更すればよい。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」組み込みCPU、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 組み込みCPU とは桁違い(無尽蔵)のテスト繰り返しが可能となった[12]。

2.2 Propellerの開発環境

Propellerの開発支援環境は、全てParallax社のサイトから無償提供されている。プラットフォームがWindows環境(XP以降)に限られるのが玉に傷である。図1はこのPropeller開発ツールにおいて、複数のウインドウを開いている様子だが、これまでの組み込みCPUの統合開発環境の機能として美味しいところは全て取り込んだ、痒い所に手の届く優秀な開発環境である。なお、Propellerのソースプログラムはplain textファイルであり、その中に回路図データをフォントとして埋め込む(このフォントデータはPropellerの各Cog内のROMテーブルにも格納されている)ために、Windowsの「言語環境の設定」として、「英語(米国)」に設定変更しないと。例えば日本語Windows環境では、ソースエディタの画面内の文字が潰れて読めなくなるので注意が必要である[8]。Windows PCをPropeller開発に使う時には、開発Host PCの環境設定をまず「英語(米国)」に設定するところから始める、のが世界共通のポイントである。

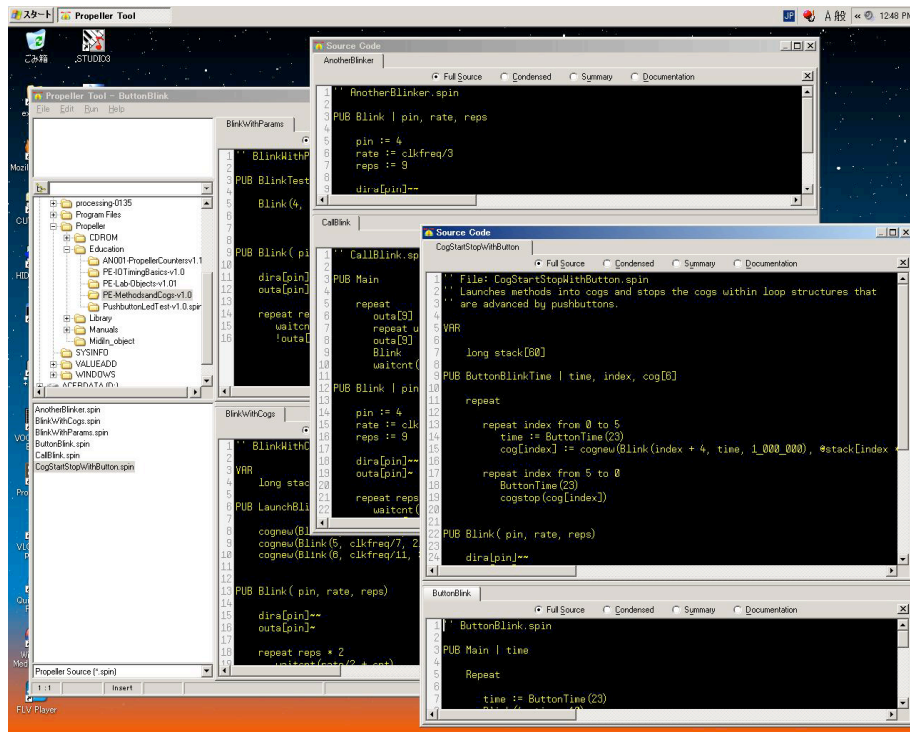


図1 “Propeller”の開発環境の画面例

2.3 Propellerのための汎用ライブラリ

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/デバッグ環境を実現した[8][11]。また、Propeller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。ここにPropellerコミュニティによりWeb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」も容易に実現できた。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機能である。図2はこの「汎用」実験回路である。

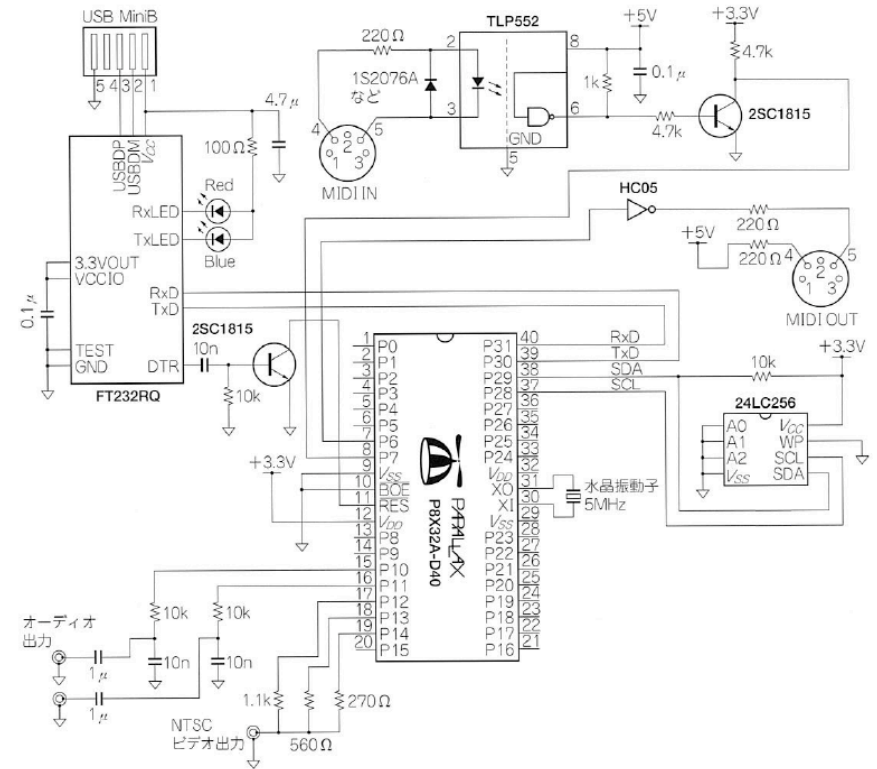


図2 “Propeller”汎用実験回路

図3は、図2のPropeller汎用実験回路をブレッドボード上で動作確認している風景であり、Max/MSPから送られたMIDI情報をソフトウェアによって(USARTでなく、1ビットずつ変化を監視するCogにより)受信し、その後にソフトスルーとして再びMIDI出力するとともに、16進数表示としてNTSCビデオモニタに出力している。また図1の回路では、MIDI入出力とNTSCビデオ出力に加えて、44.1kHzサンプリング/ステレオのオーディオ出力回路も付加しており、リアルタイム音響信号生成やフォルマント合成による4ボイス歌声合成サンプルライブラリを簡単に実験できる。音楽好きなParallax社のデザイナー(社長)のこだわりからか、Propellerの内部アーキテクチャ/レジスタマップ/命令のbit構成などは音響信号処理(DSP)を実現するのに効率的な設計になっていて[8]、最少の演算ステップ数で(高速に)高度なDSP処理を実行できる。CPU内部のROM領域には、32ビット精度のサイン関数/ログ関数/アンチログ関数のテーブル(信号処理などの数値演算に有益)も格納されている[8]。

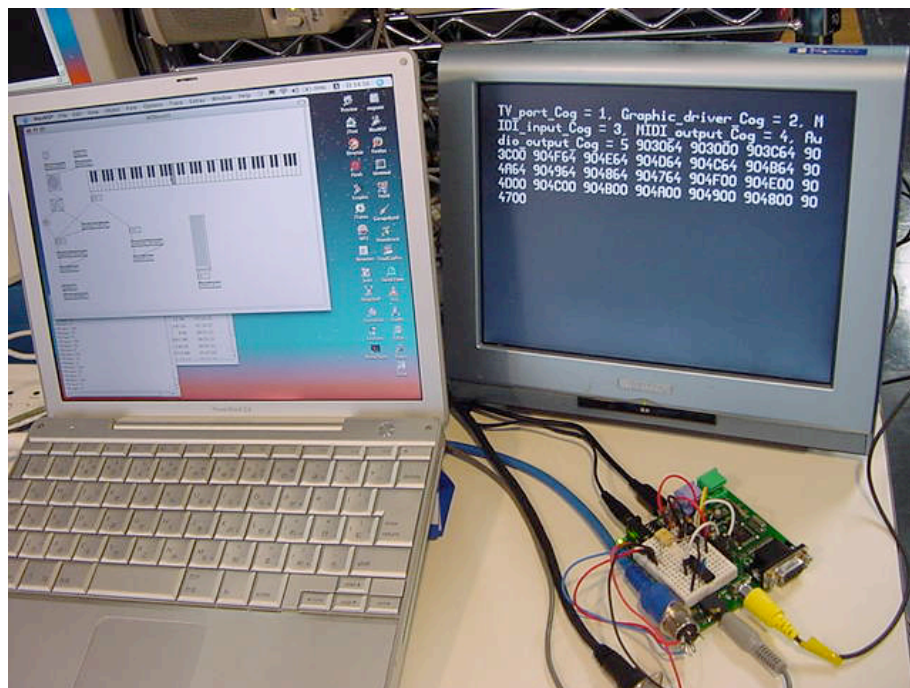


図3 “Propeller”汎用実験回路の動作風景

3. 並列処理プロセッサによるシステム開発事例

3.1 Propeller応用システム “DodecaPropeller”

Propellerを実際の作品に応用するアプローチの第1号となったのは、SUACメディアアートフェスティバル(MAF2008)で発表した、インタラクティブなインストール作品「電子十二影坊(DodecaPropeller)」である。このシステムでは、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4)でシステムが完結している。筆者はこのシステムの全回路図、全てのPropellerのソースプログラムをWebで公開しているので、興味のある方は参照されたい[13]。

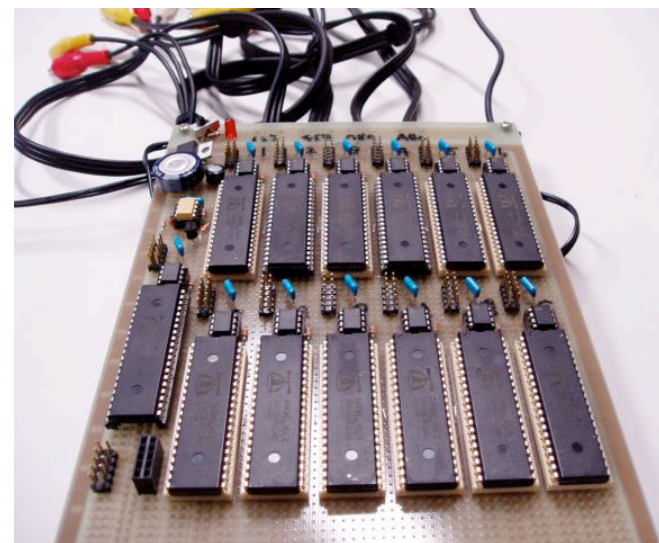


図4 “Dodeca Propeller”のボード全景

このインストールの制作にあたっては、SUACデザイン学部メディア造形学科3回生の希望者2人による制作プロジェクトを組織し、初めてのプログラミング挑戦として、Propellerのspin言語によって公開されたグラフィック・ライブラリ(Open-GLに類似)を活用したリアルタイムCG生成プログラムを開発した。図5はこの作品の展示風景であるが、多数のグラフィック生成を実現できたことで、デザイン系の学生のプログラミング教育のためのツールとしても活用できるPropellerの可能性を確認できた。

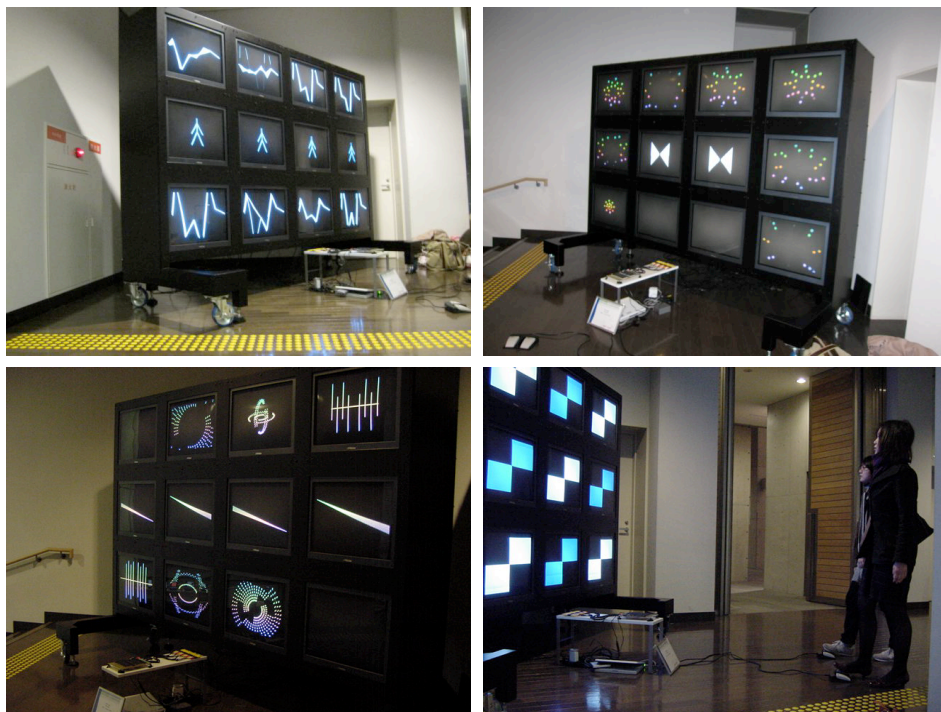


図 5 “Dodeca Propeller”の展示風景

3.2 Propeller搭載・超小型ディスプレイモジュール

Propellerを超小型カラーディスプレイモジュールの宿主(コントローラ)として搭載した「μOLED-96-PROP」という製品[14]を知って、筆者はさっそくこのモジュールを使ったシステムを2種、開発実験してみた。図6はその第1号機であり、ディスプレイと同一のサイズの基板に3個のプッシュスイッチとMIDI入力回路を搭載し、プッシュスイッチの操作によってリアルタイムグラフィック生成のパターンを切り替えるとともに、外部MIDI入力をカラー16進表示する「MIDIモニタ」機能にも切り替えられるようにした。メーカーやユーザグループからは、対応した各種のグラフィックライブラリが提供されているので、このディスプレイ(96×64画素、RGB合わせて8ビット[3+3+2]の変則疑似フルカラー)についての詳細知識はほとんど不要で、スタンドアロンのカラー表示システムを製作できる可能性を実感できた。

これに続いて製作した第2号機では、ディスプレイと同一のサイズの基板に2次元加速

度センサとMIDI出力回路を搭載し、ほぼ水平にしたモジュール全体を上下左右に傾けると画面内のカラー円環が転がるように移動し、この2次元座標情報をMIDI出力することで、その動きを外部のシステムでもシミュレーションできるようにした。似たようなものは、iPhoneアプリでも有名であるが、Propellerだけでグラフィック表示からシステム全体までを単体で実現できる事は、パソコンを使わないで完結したインストール作品の制作などに、多くの可能性を示唆している。

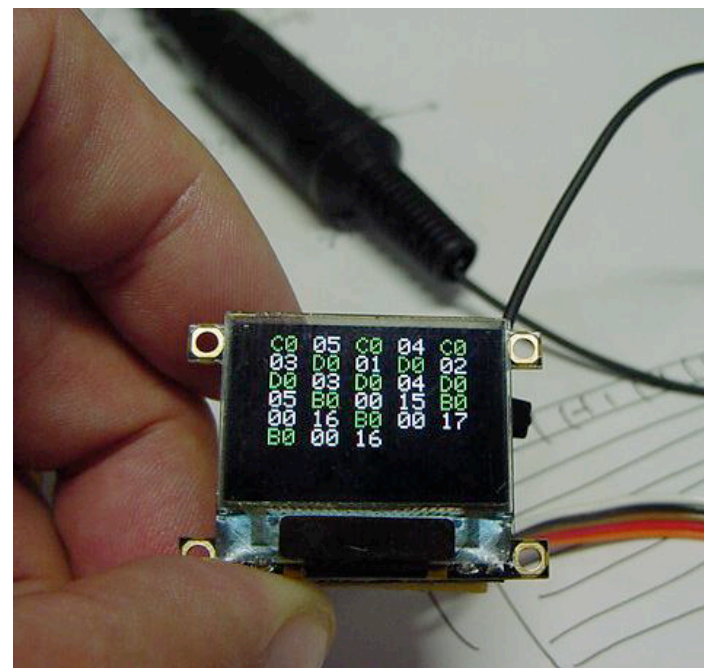


図 6 Propeller搭載・超小型ディスプレイモジュール

3.3 4マウス・インターフェース

SUACデザイン学部メディア造形学科の4回生が企画したインストール作品のためのインターフェースとして、Propellerを活用した「4マウス・インターフェース」装置を開発した。この作品はFLASHで開発して外界とはGAINERでやりとりするが、機能として「同時に4人が操作する4個のマウスの情報が欲しい」という要請があった。通常のパソコンでは、マウスは1個しか使えず、同時に複数のマウスをUSB接続しても、それらの情報は重畳してしまい意味を為さない。

そこで、USBよりもプロトコルの単純なPS/2マウスを4個使用し、Propellerの8個のCogのうちの4個を、それぞれのPS/2マウスとシリアル通信する、というシステム構成とした。Parallax社のサイトには、PS/2マウスやIBM PCキーボードと通信するソフトウェアライブラリは多数公開されており、実験はものの1時間ほどで簡単に完了してしまった。このあたりは、スケッチングの汎用ツールとしての可能性そのものと言えよう。

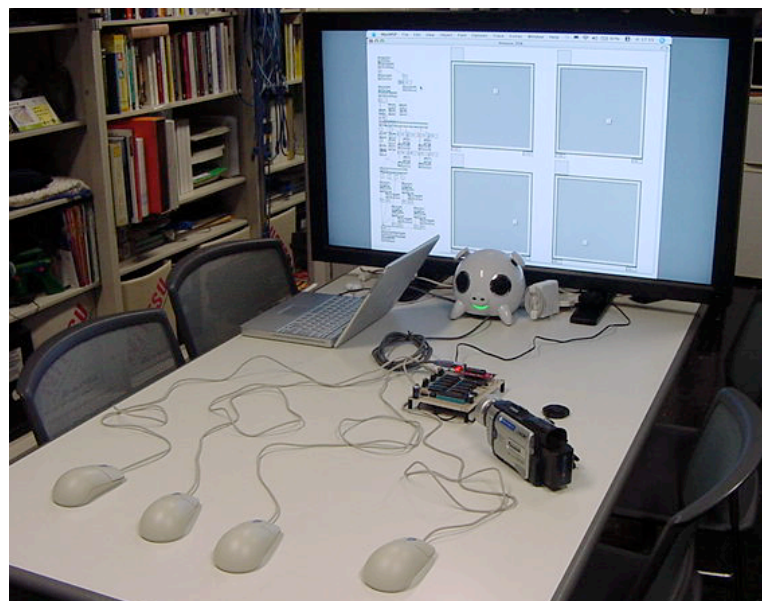
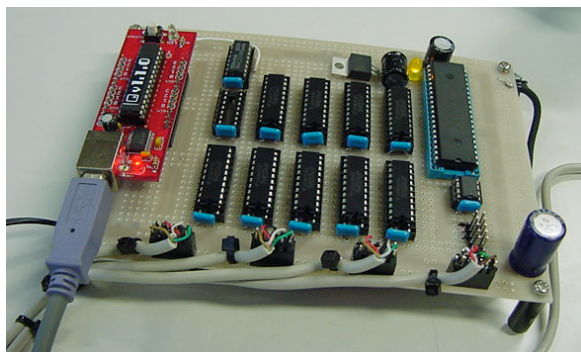


図 7 4マウス・インターフェース

図7は、製作した「4マウス・インターフェース」装置であり、PropellerとGAINERの搭載された基板には、4個のマウスそれぞれのX情報(8bits)・Y情報(8bits)・左右スイッチ(2bits)のための多数のラッチ(74HC574)が並んでいる。当初はこれらの情報を全てデジタルとしてGAINERから取り込む計画だったが、GAINERの標準ファームウェアにおいて、デジタル入力に「過去の状態からの変化を記憶する」という機能が強制的に付加されていることが判明し、ファームウェア書き換えなしにはデジタル入力モードが使用できないと判断した。そこでやむなく、9個の8ビットラッチ全ての出力にR-2Rラダー抵抗を接続してD/A変換し、GAINERの8チャンネルアナログ入力モードにてマウス座標情報を取得した。マウススイッチの8ビットについては、GAINERのデジタル出力から3ビットのセレクト信号を与えて、GAINER上のテスト用スイッチの信号に時分割選択的に供給することで実現した。

図8/図9は、このインターフェースを使用したインストール作品の体験展示発表の様相である。この作品(見崎央佳・作)は一種のゲームであり、画面内にランダムに出現する「音符」を4人の参加者がそれぞれ自分のマウスで操作するキャラクターによって素早く集めることで、対応して生成するサウンドによって、全体として一種の音楽を形成する、というものである。

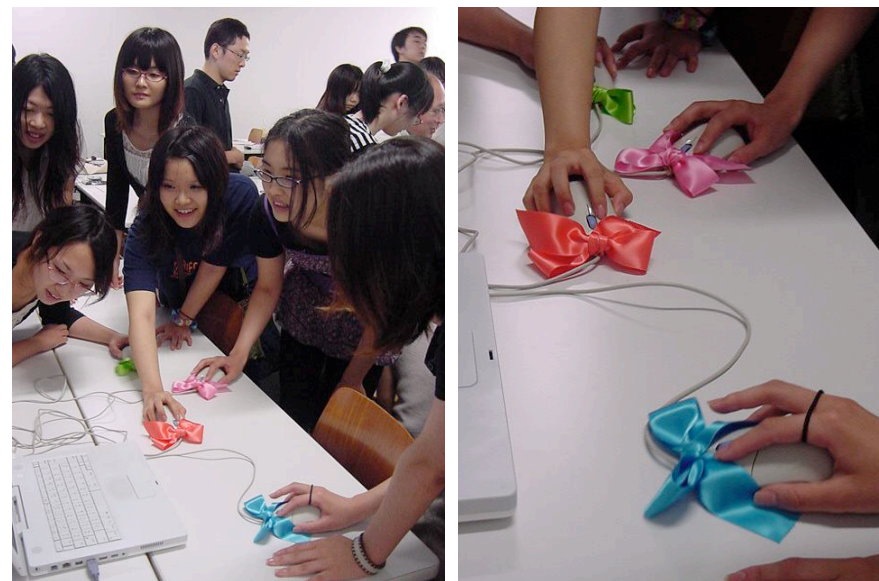


図 8 4マウス・インストール作品の展示発表風景(1)



図 9 4マウス・インストール作品の展示発表風景(2)

3.4 32チャンネル非接触センサ楽器 "Peller-Min"

2009年の夏に新たな「新楽器」プロジェクトとして製作したのは、Propellerを活用した32チャンネル非接触センサ楽器 "Peller-Min" である。この楽器の実験デモは、2009年10月31日、SUACにて開催された、メディアアートフェスティバル(MAF2009)・文化庁メディア芸術祭のMAF2009パフォーマンスとして、初めて公開した。このイベントでは、文化庁の目玉としてヤマハの「テノリオン」が5台展示されていたが、この「256個ものスイッチをいちいち触らないといけない楽器」に対抗する意図があったのは事実である。また、2009年6月にカーネギーメロン大学で開催された国際会議NIME09のコンサートセッションでは、「テルミン風」の新楽器を使った作品公演がいくつかあり、それを見ての演奏表現上の問題意識から、この楽器の基本コンセプトが生まれた。

アナログ32チャンネル入力のうち16チャンネルについては、SHARPの赤外線距離モジュールを8個ずつ、円環状のパイプ上に配置している。これは伝統的なテルミン演奏に似た演奏表現を実現する。また残り16チャンネルについては、小型テーブル状の台に左右8個ずつの高輝度ブルーLEDを密に配置し、それぞれのLEDと隣接して高感度光トランジスタセンサを並べている。その上空に掌をかざすことで、掌に反射した光量に対応したポリフォニックな連続値が得られることで、当初からのイメージ「触らずに音を捏ね回す」ニュアンスの演奏表現を実現することが出来た。

現在の "Peller-Min" は最終形ではなく、まだシステム基板には32チャンネルの非接触スイッチ回路(製作済)のためのコネクタも残っているので、本稿にはその発展途上の写真は掲載しないことにした。2009年12月5日に国立音楽大学で開催されるコンサートにおいて、この "Peller-Min" を用いた作品 "controllable untouchableness" を世界初演の予定なので、まずはここでこの楽器と出会っていただければ幸いである。

4. おわりに

デザイン領域、メディアアートの領域で注目されている「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)の事例紹介として、Propellerプロセッサの活用に焦点を当てた。開発プラットフォームに並列処理プロセッサPropeller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。今後もさらに、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) 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).
- 3) 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>
- 4) 長嶋洋一: 新・筋電センサ "MiniBioMuse-III" <http://nagasm.suac.net/ASL/SIGMUS0108/>
- 5) 長嶋洋一: センサを利用したメディア・アートとインストール作品の創作 <http://nagasm.suac.net/ASL/sensor01/>
- 6) 長嶋洋一: 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/>
- 7) Yoichi Nagashima: GHI Project - New Approach for Musical Instrument, Proceedings of 2007 International Computer Music Conference, International Computer Music Association (2007).
- 8) 長嶋洋一: Propeller日記 <http://nagasm.suac.net/ASL/Propeller/>
- 9) 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 2008-MUS-77, 情報処理学会 (2008).
- 10) 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 2008-MUS-76, 情報処理学会 (2008).
- 11) 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).
- 12) 長嶋洋一: シーズ指向による新楽器のスケッチング, 情報処理学会研究報告 2009-MUS-80, 情報処理学会 (2009).
- 13) 電子十二景坊 <http://nagasm.suac.net/ASL/12Propeller/>
- 14) μ OLED-96-PROP, Little PCB Solutions, <https://www.littlepcbsolutions.com/uOLED-96-PROP.html>