

デザインプロセスにおける「スケッチ」と物理コンピューティング

"Sketching" in design process and physical computing

長嶋 洋一

デザイン学部メディア造形学科

Yoichi NAGASHIMA

Department of Art and Science, Faculty of Design

マイコンシステムの高性能化とオープンソース・ソフトウェアの潮流から最近注目されている「フィジカル・コンピューティング」について、デザインプロセスおよびメディアアートとの関係に注目して紹介する。具体的なシステムとして4種類のプラットフォームを解説し、大学院レベルのメディアデザイン教育への応用について検討した。合わせて、この分野で近年注目されている国際ワークショップ"Sketching in Hardware 2008"の参加報告を行った。

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 (1) four popular systems as platforms, (2) consideration for application in graduate course in media design, and (3) report of the international workshop "Sketching 2008".

1. はじめに

筆者は本学において、メディア・デザイン系の学生に対して、インストールなどのインタラクティブ作品の創作に関連する教育を行っている。本稿では、マイコンシステムの高性能化とオープンソース・ソフトウェアの潮流から最近注目されている「フィジカル・コンピューティング」について、デザインプロセスおよびメディアアートとの関係に注目して紹介する。具体的なシステムとして、国内外でよく知られた4種類のプラットフォームを解説し、大学院レベルのメディアデザイン教育への応用について検討する。合わせて、この分野で近年注目されている国際ワークショップ"Sketching in Hardware 2008"の参加報告を行う。なお、筆者のこれまでの関連研究/活動等の報告については、文献[1-33]を参照されたい。

2. メディア・インストールとソフトウェア・プラットフォーム

本学のデザイン学部メディア造形学科[34]では、映像系・情報系・コミュニケーション系という3つのコースを学生が選択し、作品制作や研究を進めている。また大学院デザイン研究科メディアデザインコースでも、映像/アニメーション/インストールなどの作品制作を行っている。これら学生作品を発表する機会としては、インターカレッジ[35]やSUACのMAF(メディアアートフェスティバル)[30-31]での発表展示だけでなく、最近ではNHKデジスタ[36]、文化庁メディア芸術祭・学生CGコンテスト[37]、アジアデザインアワード[38]、等のコンペに学生インストール作品が入選することで、学生相互の刺激にもなっている。理工系でないデザイン系の学生にとって、インタラクティブな仕組み(広義のセンサと広義のディスプレイ)や、インタラクションのアルゴリズムの実現(プログラミング)を工学的に全てゼロから行うことは不可能であり、アート/デザインの制作の裏側に、あ

る種のブラックボックスとして活躍する制作支援環境が必要となる。ここではまず、インストールを構築する要素、という視点から、3つの代表的なソフトウェア・プラットフォームを紹介する。

2-1. Max/MSP/jitter

1990年代初頭から現在まで世界的な定番となっている開発支援環境は、Max/MSP/jitterである[39]。筆者の担当するメディア造形学科専門科目「サウンドデザイン」「サウンドデザイン演習」「音楽情報科学」と大学院科目「メディアデザイン特論」においても、Max/MSP/jitterはその中心である。後述するハードウェア・プラットフォームであるAKI-H8やPICなどの組み込みマイコン/センサ系とMIDIで結び付き、さらにOSC[40]によってネットワーク経由で他の機器と情報交換したり、Wiiリモコンと[41]、あるいはiPhone/iPod touch[42]と通信できるのもMax/MSP/jitterの強味であり、インストール作品の開発環境/プラットフォームとして今後も重要である。サウンドの部分に重点を置いたインストール作品では、サウンド出身であるこの環境はなにより適合している。

2-2. FLASH

当初はWebコンテンツの機能向上やエンタテインメント系/ゲーム系のプラットフォームとして評判になったFLASHだが、最近ではケータイのアプリ環境としてJavaを凌駕し、情報系の学生がC/Javaを学ぶのと同様に、デザイン系の学生のプレゼン標準ツールとなりつつある。これまでも、タイムラインに沿ったFLASHムービーであれば、Max/MSP/jitter上でもQuickTimeが再生していたが[43]、後述するGAINERによって、外界とインタラクションするアルゴリズム自体をFLASH上で実現できるようになったのは非常に大きな進展である。実際に、2008年3月の卒展においては、インストールのプラットフォームとしてMax/MSP/jitterよりも

FLASHの方が作品数として多く使用された。ビデオカメラを画像センサとして利用した学生作品も登場している。

2-3. Processing

メディアアートのプラットフォームのもう一つの主流である Processing[44]は、日本語の解説本が改訂され[45]、GAINERも対応しており、何よりフリーである事から、こちらも今後の発展が期待されている。Max/MSPとFLASHを使えた程度の学生がゼロから独学でProcessingをマスターして作品に仕上げる過程を筆者も目撃しており、サウンドの部分での課題を別にすれば、ビジュアル系重視のメディアアートのプラットフォームとして重要である、と評価している。2008年にはビデオカメラを画像センサとして利用したProcessingによる学生作品も登場した。

3. インスタレーションのインターフェースとしてのハードウェア・プラットフォーム

理工系でないデザイン系の学生にとって、上述のソフトウェア・プラットフォームとともに必要なのが、現実の物理世界とインタラクティブな仕組みを実現するための「広義のセンサと広義のディスプレイ」というインターフェースである。これは実際には、マイクロコントローラを内蔵したハードウェア・プラットフォームとして作品それぞれに応じたカスタムメイドが必要となるために、インスタレーション制作において最大の壁となってきた。これがマイコンシステムの高性能化とオープンソース・ソフトウェアの潮流によって身近になってきた、というのが本稿の主題である。ここでは、インスタレーションを構築する要素、という視点から、4つの代表的なハードウェア・プラットフォームを紹介する。なお、紙面の関係でそれぞれの写真については文献[46]を参照されたい。また本稿末尾の図1に、この4つのチップ/システムを、メディアアートのプラットフォームという視点で比較してみた。

3-1. AKI-H8

秋葉原・秋月電子のAKI-H8は、これまでメディアアートのプラットフォームとして多くの作品の基盤として活躍してきた。筆者のサイト[47-48]を参考にシステムを製作する作家も多く、ICCでの2年間の常設展示(連続運転)など信頼性の上でも実績は多い。MS-DOS時代からの開発環境は最近では古めかしいが、スピードを求めなければC言語で、また性能を求める場合にはアセンブラによってファームウェアを開発し、実機上でEEPROMに書き込む手軽さはまだまだ現役である。従来のMIDI/RS232Cに加え、USBやイーサネットI/Fも登場した。MIDI等で通信するスタンドアロンシステムのプラットフォームであり、ソフトウェアDSP(信号処理)も可能である。

3-2. Arduino

イタリアのグループが開発したArduino[49-50]は、オープンソースの開発環境と低価格のボード、スタンドアロンおよびパソコンインターフェース(Max/MSP, Flash, Processingに対応)の動作モード、さらにZigbeeに対

応したXbee Shieldオプション等を売りにしている。ホストPCとはUSBで接続するが、シリアルポートが1ポート(Tx/Rx)しかなく、標準開発環境のC言語ベースでは、センサ情報のMIDI送信には問題ないものの、MIDI受信には対応できない問題点を筆者は確認した。あまり時間的な制約のない範囲で、多種のセンサ情報に対応した関係性を実現するスタンドアロンシステムの中核として有力な候補である。海外の作家が多く採用している理由も、「Webで簡単に購入できる」「とにかく安いから」であった。

3-3. GAINER

IAMASの小林氏の開発したGAINER[51]は、日本においてPhysical Computingの概念をセンセーショナルに普及させた記念碑的なシステムである。スタンドアロン動作を排してパソコンI/Fモードに限定し、ホスト環境としてはMax/MSP/jitterとFlashとProcessingをカバーしており、デザイン系の学生などがインタラクティブなインスタレーションを実現する最短経路を提供する。CPUのファームウェアを書き換える等の荒技を別にすれば、MIDI入出力や単独動作など出来ない事も多いが、16ビットのポートを入出力のいろいろな組み合わせとする支援環境などの充実により、最近もっとも注目されている。本学での筆者の講義/演習においても、メディア造形学科の全学生がGAINERを体験するプログラムとなっている。

3-4. Propeller

Basic Stampの開発元で知られるParallax社が新しく発信しているCPUが、Propeller[52-54]である。従来とは基本的概念から異なるユニークなCPUであり、8個の並列動作する32ビットCPUコアによって、非常に強力な性能のシステムをスタンドアロンで実現できる。UART等に頼らずシリアル通信でもCDレベルのデジタルオーディオ出力でもソフトウェアだけで実現し、さらにNTSC/PAL対応のビデオ信号まで8個のCPU(Cogと呼ぶ)のうち2個により生成できてしまう能力には注目したい。オブジェクト指向の並列処理記述言語Spinによる開発環境(アセンブラと協調)も充実し、Propellerチップを取り囲むライブラリ等の支援環境によって、デザイン系の初学者でもハイレベルのシステムを実現できる可能性を感じさせる。筆者は国内であまり知られていないPropellerを用いたインスタレーションをMAF2008に向けて制作するとともに、専門誌の記事によって紹介した[55-56]。

4. デザインプロセスと物理的コンピューティングとスケッチング

上述のソフトウェア・プラットフォーム(アルゴリズム制作支援環境)およびハードウェア・プラットフォーム(物理世界とのインターフェース)をまとめた概念として、近年「物理的コンピューティング(Physical Computing)」という概念が普及してきた。ここでは、インスタレーションなどのメディアアートだけでなく、広くデザインプロセスを飛躍させるツールとしての物理的コンピューティングについて整理・検討したい。

4-1. 物理的コンピューティングの概念

日本語版 Wikipedia には「物理的コンピューティング」や「フィジカルコンピューティング」という用語は記載されていないが、英語版の Wikipedia には解説が載っている[57]。これを粗く意識すれば、広い意味で「物理的コンピューティング」とは、アナログの世界を理解し反応できるソフトウェアとハードウェアの活用により、対話的な物理的なシステムを構築することを意味する。この定義だと、知的な交通管制システムやファクトリオートメーションプロセスなども含まれるが、ここでの物理的コンピューティングは、「デジタル世界との人間の関係を理解するための創造的なフレームワーク」という位置付けに重点がある。日本語版も出版されている「Make」誌にその典型があるが、実際には物理的世界のアナログ入力をソフトウェア・システムに翻訳するのにセンサとマイクロコントローラを使用し、モーター・サーボ・照明または他のハードウェアなどのデバイスを制御するという、DIY 趣味プロジェクトがその先駆である。MIT などでは工学教育的な意義を重視し、デザイン/アート系でもインタラクティブアートのプラットフォームとして注目している。

4-2. 物理的コンピューティングの実例

●ミュージアム

科学館などの体験型ミュージアム/学習型ミュージアムでは、単なる展示でなく、スイッチやタッチパネルなど、来場者が対話的に体験するインタラクティブな展示形態を早くから実現してきた。ある意味ではゲームセンターとの境界が曖昧になり、物理的コンピューティングの対象として、人間からの働きかけをセンシングするセンサ(入力)だけでなく、映像・音響に加えて、他の五感、特に「触覚」のチャンネルからもコンテンツを提示(広義のディスプレイ)する挑戦が進められている。

●アート

芸術、メディアアートの領域では、インタラクティブ・アートは25年前ほどから一つの中心であり、アルスエレクトロニカでもこの部門の重要性は揺るがない。物理的コンピューティングを実行するプロジェクトの一例として、Scott Snibbe、Daniel Rozin、Rafael Lozano-Hemmer、Jonah Brucker-Cohen などの仕事(リンクあり)が Wikipedia に紹介されている[57]

●プロダクトデザイン

物理的コンピューティングの手法は、プロダクトデザインの領域でも普及している。後述する「Sketching」のマネジメント手法は、新しいアイデアの製品企画・開発における重要なメソッドとして普及してきた。人間の手作業で試作したプロトタイプを、量産型の組み込み型システムとして実現する上で、ラピッド・プロトタイピングの費用対効果の意義は大きい。世界的には IDEO[58]のデザインプロセスのアプローチとしてよく知られている。

●工業製品への応用

企業の製品においては、ソニーの Eyetoy やゲームのダンスダンスレボリューションから Wii まで、インタラクティブな考え方の製品は新しいユーザインターフェースとして広く普及している。工場での FA においては、ロボットビジョン(画像認識)からの制御システムにおいて、さら

に民生機器ではマイクからの音響解析による状況分析と反応という領域で、セキュリティ関係ではカメラ画像からのジェスチャ認識において等、色々な研究が進められるとともに実際の製品やサービスとして展開されている。

●科学領域での応用

物理的コンピューティングのサイエンス領域での応用としては、カスタムセンサの開発とか、科学的実験(教育)の領域で歴史がある。最近の学研「大人の科学」はその発展であろう。

4-3. IAMAS のデザインプロジェクト

IAMAS で3年間にわたって進められた「ガングプロジェクト」[59-60]においては、新しい玩具のデザインから試作・展示までのデザインプロセスを、「リサーチ」「スケッチ」「プロトタイピング」「展覧会」という4つの工程として明確に整理し、実際に多くのコンペで入賞するような作品を生み出す成功をおさめた。ここで重要なのが、スケッチからプロトタイピングの段階で、GAINER などを活用して実際に動く試作を簡単に実現していく、という部分であり、「机上の空論」でないシステムの実現に貢献した。最近の「スケッチング」という概念は、ここで言う「スケッチ」と「プロトタイピング」の両方にまたがる領域となりつつある。

4-4. Digital Designer の Rediscover

参考文献リンク[61]にキャッシュしたのは、New York Times の Technology 欄の記事である。ここでは、デジタル領域のデザイナーが、自分の「手」とアイデアとの融合を改めて再発見するツールとしてのスケッチング、について紹介されている。この記事にあるように、アメリカではスケッチングを、MIT/スタンフォードなどIT関係の教育機関と、IT応用の企業が、エンジニア育成の点から注目し強化していることが伺われる。メディアコンテンツ立国が必須の日本においても、この重要性は同じであろう。

4-5. Sketching in Hardware

2006年から、この領域の関係者の国際会議として、「Sketching in Hardware」が開催されてきた[62-64]。この主催者 Mike Kuniavsky 氏が Tod E.Kurt 氏とともに活動しているのが、ThingsM というスタジオである[65]。その説明によれば、ThingM とはコビキタスコンピューティング装置スタジオであり、ネットワークでつながれた先端のエレクトロニクスによってユーザの体験をデザインし、問題を自己表現して解決する、オリジナルの製品とサービスについて研究開発しているという。Mike Kuniavsky 氏は、PacBell、Crayola、National Public Radio、McGraw-Hill、Cypress Semiconductor、Whirlpool、Macromedia、Corel、Qualcomm、Yamaha 等の企業のコンサルタント/デザイナーとしても仕事をしている[66]。国際会議「Sketching in Hardware」は、現在までのところ一般的な国際会議とは異なり、主催者である Mike Kuniavsky 氏に招待されたメンバーだけが参加できる、closed なカンファレンスであるが、open な場にするための議論は行われている。日本からは、GAINER/funnel の開発者の小林茂氏(IAMAS)が2007年に参加

し、引き続き 2008 年に参加する機会に依頼して招待されたため、筆者も 2008 年に参加することができた。2008 年の参加者については、リンク[64]の PARTICIPANTS(それぞれの紹介へのリンクあり)を参照されたい。また、Sketching に印刷された Proceedings は無いが、筆者のプレゼンは[67]に、レポートは[68-69]にある。関連した小林茂氏の記事は[70]にあり、小林氏は 2008 年 12 月に SUAC で開催するメディアアートフェスティバル 2008 において、「フィジカル・コンピューティング」ワークショップの講師として招聘の予定である(原稿執筆時点)。

5. おわりに

最近注目されている「フィジカル・コンピューティング」について、デザインプロセスおよびメディアアートとの関係に注目して紹介した。全てをマスターするには本学大学院の 2 年間でも不足するものの、効果的な汎用プラットフォームの助けによって、学生の作品制作レベルは年々向上している。今後、ますます発展するであろうこの領域において、メディアデザイン教育のための新しいオーサリングツール/汎用プラットフォームの実現に向けて、さらに検討していきたいと考えている。

参考文献/リンク

- [1] 長嶋洋一, 静岡文化芸術大学スタジオレポート, 情報処理学会研究報告 Vol.2000, No.118 (2000-MUS-38), 情報処理学会, 2000
- [2] 長嶋洋一, メディア・インストールを用いたインタラクティブ・パフォーマンスについて, 平成 13 年度前期全国大会講演論文集 2, 情報処理学会, 2001
- [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
- [4] Yoichi Nagashima, Composition of "Visional Legend", Proceedings of International Workshop on "Human Supervision and Control in Engineering and Music", 2001
- [5] 長嶋洋一, SUAC におけるメディアアート活動の報告(2000-2001), 静岡文化芸術大学紀要・第 2 号 2001 年, 静岡文化芸術大学, 2002
- [6] Yoichi Nagashima, "IMPROVISESSION-II": A Performing/Composing System for Improvisational Sessions with Networks, Proceedings of International Workshop on Entertainment Computing, 2002
- [7] Yoichi Nagashima, Interactive Multi-Media Performance with Bio-Sensing and Bio-Feedback, Proceedings of International Conference on Audible Display, 2002
- [8] Yoichi Nagashima, Interactive Multimedia Art with Biological Interfaces, Proceedings of 17th Congress of the International Association of Empirical Aesthetics, 2002
- [9] 長嶋洋一, インタラクティブアートの統合的システム・プラットフォームとしての Max/MSP, DSP サマースクール 2002 論文集, 静岡文化芸術大学, 2002
- [10] 長嶋洋一, メディア・アートと生体コミュニケーション, 静岡文化芸術大学紀要・第 3 号 2002 年, 静岡文化芸術大学, 2003
- [11] Yoichi Nagashima, Bio-Sensing Systems and Bio-Feedback Systems for Interactive Media Arts, Proceedings of 3rd International Conference on New Interfaces for Musical Expression, 2003
- [12] 長嶋洋一, 音楽的ビートが映像的ビートの知覚に及ぼす引き込み効果, 芸術科学会論文誌 Vol.3 No.1, 芸術科学会, 2003
- [13] Yoichi Nagashima, Combined Force Display System of EMG Sensor for Interactive Performance, Proceedings of 2003 International Computer Music Conference, 2003
- [14] 長嶋洋一, 音楽 / 芸術表現のための新インターフェース, 静岡文化芸術大学紀要・第 4 号 2003 年, 静岡文化芸術大学, 2004
- [15] Yoichi Nagashima, Drawing-in effect on perception/cognition of musical beats and visual beats, Proceedings of International Symposium on Musical Acoustics, 2004
- [16] Yoichi Nagashima, Controlling Scanned Synthesis by Body Operation, Proceedings of the 18th International Congress on Acoustics, 2004
- [17] Yoichi Nagashima, Measurement of Latency in Interactive Multimedia Art, Proceedings of International Conference on New Interfaces for Musical Expression, 2004
- [18] 長嶋洋一, メディアアート研究拠点化についての研究, 静岡文化芸術大学紀要・第 5 号 2004 年, 静岡文化芸術大学, 2005
- [19] 長嶋洋一, マルチメディアコンテンツのための自動作曲システム, 静岡文化芸術大学紀要・第 6 号 2005 年, 静岡文化芸術大学, 2006
- [20] Yoichi Nagashima, Students' projects of interactive media-installations in SUAC, Proceedings of International Conference on New Interfaces for Musical Expression, 2006
- [21] 長嶋洋一, コンテンツクリエイターのための著作権フリー音楽クリップ生成システム "FMC3", 芸術科学会論文誌 Vol.5 No.3, 芸術科学会, 2006
- [22] 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
- [23] 長嶋洋一, 映像酔いとサウンドによる抑止の検討, 静岡文化芸術大学紀要・第 7 号 2006 年, 静岡文化芸術大学, 2007
- [24] Yoichi Nagashima, GHI project and "Cyber Kendang", Proceedings of International Conference on New Interfaces for Musical Expression, 2007
- [25] 長嶋洋一, マルチメディア心理学実験のためのプラットフォームについて, 日本音楽知覚認知学会 2008 年春季研究発表会資料, 日本音楽知覚認知学会, 2008
- [26] 長嶋洋一, サウンド・インストールのプラットフォームについて, 情報処理学会研究報告 Vol.2007, No.50 (2008-MUS-75) (2008-HCI-128), 情報処理学会, 2008
- [27] 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
- [28] 長嶋洋一, メディアアートのための汎用インターフェースのプラットフォームについて, 情報科学技術フォーラム 2008 講演論文集, 情報処理学会・電子情報通信学会, 2008
- [29] 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
- [30] <http://1106.suac.net/news2/installation/>
- [31] <http://1106.suac.net/news2/installation2/>
- [32] <http://nagasm.org/>
- [33] <http://1106.suac.net/>
- [34] <http://www.suac.ac.jp/media/>
- [35] <http://icwg.dacreation.com/>
- [36] <http://1106.suac.net/news2/digista/>
- [37] <http://www.cgarts.or.jp/scg/2007/prize/kodama/>
- [38] http://adaa.jp/2007/j/12_list.html
- [39] <http://www.cycling74.com/>
- [40] <http://opensoundcontrol.org/>
- [41] http://www.iamas.ac.jp/~aka/max/#aka_wiiremote
- [42] <http://www.iamas.ac.jp/~aka/iphone/>
- [43] <http://1106.suac.net/news2/docs/sakata.mp4>
- [44] <http://processing.org/>
- [45] <http://www.amazon.co.jp/gp/product/4861005582>
- [46] <http://nagasm.suac.net/ASL/paper/sigmus0808.pdf>
- [47] <http://nagasm.suac.net/ASL/mse/>
- [48] <http://nagasm.suac.net/SSS/>
- [49] <http://www.arduino.cc/>
- [50] <http://nagasm.suac.net/ASL/Arduino/>
- [51] <http://www.gainer.cc/>
- [52] <http://www.parallax.com/Default.aspx?tabid=295>
- [53] <http://www.parallax.com/tabid/407/Default.aspx>
- [54] <http://nagasm.suac.net/ASL/Propeller/>

[55] 長嶋洋一, Propellerを使った体験型アート作品の製作(前編), トランジスタ技術 2008年10月号, CQ出版社, 2008
 [56] 長嶋洋一, Propellerを使った体験型アート作品の製作(後編), トランジスタ技術 2008年11月号, CQ出版社, 2008
 [57] http://en.wikipedia.org/wiki/Physical_computing
 [58] <http://www.ideo.com/>
 [59] <http://www.iamas.ac.jp/project/gangu/>
 [60] <http://www.iamas.ac.jp/project/ams/openstudio/project.html>
 [61] <http://nagasm.suac.net/ASL/sketch/>
 [62] <http://www.sketching06.com/>
 [63] <http://www.sketching07.com/>
 [64] <http://www.sketching08.com/>
 [65] <http://thingm.com/>
 [66] <http://thingm.com/about-us/team/mike-kuniavsky.html>
 [67] <http://1106.suac.net/PhysiCom/>
 [68] <http://1106.suac.net/Sketch08/>
 [69] <http://nagasm.suac.net/ASL/paper/sigmus0809.pdf>
 [70] <http://techon.nikkeibp.co.jp/article/NEWS/20080825/156872/>

項目 \ System	AKI-H8	Arduino	GAINER	Propeller
CPU	Hitachi H8/3048 32bits	Atmel ATmega 8bits	Cyperss 8bits CY8C29466	32bits* 8CPU
clock	16/25 MHz	16 MHz	12 MHz	80 MHz
RAM	4K bytes	1K bytes	2K bytes	32K bytes
EEPROM	128K bytes	16K bytes	32K bytes	32K bytes (external only)
Power Supply	+5V	+5V	+5V	+3.3V
IDE	MS-DOS batch	Processing like	Max/MSP Flash Processing	original IDE
Language	Assmbler c	c	Max/MSP Flash Processing	Spin Assembler
PC interface	RS232	USB	USB	USB
Standalone	○	○	×	○
Serial Ports	2	1	4(max)	8(max)
A/D	12bits/8ch	10bits/6ch	14bits/12ch	16bits/28ch(max)
Audio D/A Out	100KHz 8bits 2ch	6ch PWM	×	44.1KHz 16bits 14ch(max)
Video Out	×	×	×	NTSC/PAL 2ch(max)
Character/Font Table	△	×	×	○
inter process Communication	interrupt/polling hand-shake	polling	polling	shared memory polling
fast response	◎ interrupt	×	×	◎ parallel CPU
MIDI Out	○	○	○	○
MIDI In	○	×	×	○