

デザイン・エンタテインメントを支援する プラットフォームについて

長嶋 洋一†

静岡文化芸術大学†

1. はじめに

本研究は、広義のエンタテインメントをデザインするための支援環境と、デザイン教育に関するプラットフォームの提供を目的とする。学生へのデザイン教育に、従来「エンタテインメントデザイン」として提供されてきたゲームやエンタテインメントプラットフォームを設計する、という開発側の立場であった。筆者は近年、デザイン側の「デザインとして行なわれるべきもの」が新しいエンタテインメントとなる可能性を提唱し検討している。これは、ニコニコ動画などで普及しつつあるCGM(Consumer Generated Media)を、コンテンツだけでなく物理的なシステムにまで拡張した視点である。

新しいシステムをデザインする上で、ソフトウェア設計(プログラミング)については、ProcessingやArduinoのように、オープンソース文化の普及とともに支援環境が充実してきた。そして近年、工学的知識を必要とするハードウェア設計の領域でも、物理コンピューティング(スケッチング)と3Dプリンティングの普及により、「ハードウェア・オープンソース」の時代となってきた。本稿では、筆者の進めてきた複数のCPUモジュールを統合したプラットフォーム試作の実例を紹介するとともに、この分野でのデザイン教育において、デザイン行為そのものがエンタテインメントとなる可能性について検討したい。

2. オープンソースによる新しい「ものづくり」

20世紀の日本を支えた「製造業」(高品質・高付加価値・大量生産)は、21世紀になって日本の有名な電機メーカーの多くが失速・消滅する事例に明らかのように、過去のものとなった。20世紀の「ものづくり」は、専門のメーカーが企画・設計・試作・製造・試験・販売まで全てを行う、というビジネスモデルであった。これは半導体技術がまだ開発途上であったため、CPUの処理能力の低さやメモリの小ささをカバーするたため、専用ハードウェア(ASIC)が必要で、その開発費用を量産の数量で回収し、専門的に特化した設計技術者を抱えること、独占的・排他的な製品提供を行ってきたからである。しかし21世紀になると、CPUクロックやメモリが十分に高性能化したことで、専用LSIを開発する領域は相対的に小さくなり、全ては「十分なメモリを抱え十分に高速なCPUがソフトウェアで行う」というシステムデザイン指針に変貌した。過去のプログラムがテクニクを駆使してきたプログラミングは、限られたCPU処理能力とメモリサイズ

との戦いであったが、21世紀のプログラマはそれのような工夫よりも、ソフトウェア部品の再利用と信頼性向上や例外処理のために、進化しようとした開発支援環境の中で、基本的には同じような(馬鹿正直で誰でも読める)プログラムを作ることになってきた。興味深い流れとして、プログラマのスキルが低くても(電子工学の専門知識を持たない文系出身者であっても論理的な開発が可能)昔より高性能の製品が実現できることで、20世紀に懸念されてきた「プログラマが不足する」クライシスをそこそこ回避している。

ハードウェアのシステムはPC/携帯/パッド/ゲーム機などの専用ブラックボックスに任せて、その上のアプリとしてのゲームやWebコンテンツなどをデザインする領域でも、オープンソースツールや開発支援環境などの充実、そしてオープンソース文化の普及によって、専門のプロフェッショナルだけでなく、一般・学生などアマチュアであっても十分に世界に発信できる(多数のアクセスやダウンロードとして支持される)時代となってきた。オブジェクト指向によるソフトウェアの再利用、CC(クリエイティブコモン)に支えられたフリードキュメント/ライブラリ/ツールの無償公開交換は、過去の企業の特許やカスタム化による閉鎖的文化と違って、「誰もがお互いに共有することで皆んなでhappyになる」という新しいデザイン文化として定着した。ここに乗り遅れない日本のメーカーの没落する姿は、ある意味では時代の象徴的必然である。

アジアが世界の製造工場となったのも、このオープンソースとインターネットが原動力である。新しいシステムのアイディアを、スケッチングのテクニックでプロトタイプを、量産時に金型を作る構造物も3Dプリンターで実現してしまうこと、ほぼ実機に近いレベルで仕様検討やマーケティングを行える。メーカーは大企業である必要はなく、少数のアイデア豊かなデザイナーが企画からプロトタイプまで行えば、KickStarterなどの支援によって開発費用の獲得から試作製造まで実現できる時代である。一部の高精度な製造現場ではまだ日本の伝統が優勢であるものの、原理的には製造機械の精度はますます向上するので、血管内を移動して病巣を切除するようなロボットの製造は、名工の手でなく「超高性能微細加工ロボットが作った超微細製造機械」でしか実現できないであろう。

3. Gainer・Arduinoの課題

過去のマイコン組み込みシステムでは、理工系の開発技術者が半導体メーカーごとにclosedに提供された各種マイクロコントローラを用いて、専用の電子回路(さらに専用のASIC)によってハードウェアを実現してきた。開発環境は非常に使いにくい専用ツールで囲い込まれ、開発段

Platform for design entertainment

† Yoichi Nagashima

Shizuoka University of Art and Culture

階のバグや仕様変更は、全てファームウェアを開発するプログラマによる「尻ぬぐい」によって対応してきた。これに対して、オープンソースの代表として登場したGainerは、Max/MSP/jitter・Flash・Processingのクロスプラットフォームに対応して、新しいシステムをデザインするための標準インターフェースとして新しい時代を拓いた。開発者の小林茂氏のポリシーでもあるためかサンハヤトのブレッドボードが流行し、環境問題もありそれまでのハンダ付けを駆逐する勢いとなった。PCホスト不要のスタンドアロン・システムの中核として世界的な標準となったArduinoもまた、オープンソースの開発環境とともにハード面ではブレッドボードを使うデザイナー(学生・アマチュア)が多い。

しかし、過去に数多くのオリジナルシステムを開発制作してきた[1-6]筆者にとって、予備的な実験ならともかく、最終的に作品となる部分にブレッドボードを使う、というのは信じられない事である。Computer Musicのライブパフォーマンスに使うセンサでなくても、インスタレーション作品のインターフェースや入出力部分でも、部品を差し込んだだけのブレッドボードの信頼性の低さは致命的である。そして多くの作品では、GainerやArduinoが持つ入出力ポート数の少なさから、これを拡張する術を持たないデザイナーは、無意味に多数のGainerやArduinoを並べ、という愚直で美しくない方法を採用するために、さらにシステムの信頼性が低下する悪循環となり、最大の課題と言える。

4. 汎用実験試作ボードの開発

多くの作品においてGainer・Arduino・AKI-H8・Propellerなどを用いて、多数の入出力ポート数による表現のためのシステム拡張を手配線を実現してきた[7-12]筆者は、新しい作品を制作する過程を学生が楽しむ様子に触れた事をきっかけとして、スケッチング教育のための汎用プラットフォームとしてのカスタム基板を開発することを決意した。膨大な手配線の手間を基板化により解決しつつ、GainerやArduino単体では実現できない入出力ポートの拡張を実現することで、システムをデザインする、という本命のプロセスをエンタテインメントとして支援する、というのがその目標である。

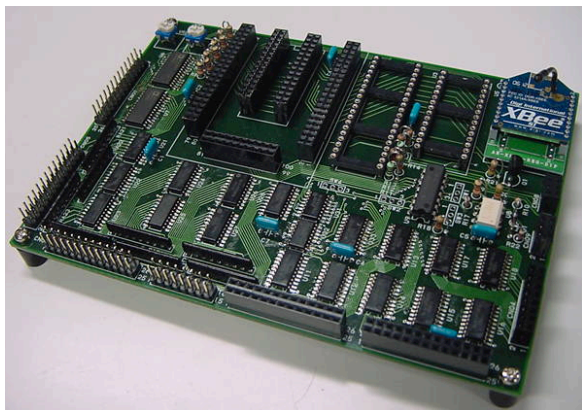


図1 汎用実験試作ボードの外観

2012年12月に完成した最初のバージョン(図1)では、基板上に搭載できるホストマイコンとして、秋月電子のAKI-H8、Arduino-Nano、Gainer Mini、そしてPropeller搭載のPropStick、という4種類に対応し、いずれかを選択して搭載する。マイコンにより実現できる拡張機能に差は

あるが、デジタル入力64ビット、デジタル出力(PWM対応)64ビット、アナログ入力24チャンネル、MIDI入出力、Xbee入出力、を標準機能として持ちながら、市販の汎用実験基板とスタックできる同一サイズ(160mm×115mm)を実現した[13]。そのために、基板上に21個のミニフラットパッケージICが並ぶこととなったが、これをFPGA化する可能性は今後の検討事項である。

5. 標準プラットフォーム化への道程

本稿執筆時点では「P板.com」社で試作した基板の動作確認と次バージョンのための仕様検討の段階にあるが、ここからデザイン・エンタテインメントのための標準プラットフォームに向けて発展させるシナリオについては既に検討してある。まず第一に、次バージョンあたりで基板を量産して安価(実費程度)に入手できるようにして、ハードウェアのオープンソースとして技術情報は全て公開する。特許のような秘密性は皆無である点がポイントである。

また合わせて、Gainerで小林氏が完備させたやPropellerのサイトに充実しているように、機能拡張のための標準ドライバ群(ミドルウェア)と、多数のサンプルソースを充実させて公開する、というソフトウェアのオープンソースも重要である。マイコンの8ビットポートから8個のラッチに分配出力したり、8個の8ビット3ステータゲートを選択することで64ビット入力するが、この原理を理解すること自体が、デザイナーにとって教育的効果が期待できる。当初から海外との交流を目指して、英語版として実現できないか、自分の首を締めるようでかなり苦しいが、検討中である。

これらの作業と並行して、主役であるデザイナーがこのツール(プラットフォーム)とどう取り組むか、という取材も兼ねて、2013年2月2日にSUACでワークショップを開催する計画を立てて、参加募集サイトをスタートさせた。ここでは、他大学/機関の専門家、デザイナーの卵(学生/院生)、メーカー技術者などの交流とともに、実際に試作プロセスをハンズオンで体験しながら本格的バージョンに対するリクエストを収集する予定である。さらに、実際の作品として活用するデザイナーに対する支援により、目に見える形での普及も目指していきたい。

6. おわりに

複数のCPUモジュールを統合した汎用プラットフォーム試作の実例を紹介し、この分野でのデザイン教育において、デザイン行為そのものがエンタテインメントとなる可能性について検討した。世界のオープンソースの潮流に乗りつつ、新しい可能性に挑戦していきたい。

7. 参考文献

- [1] <http://nagasm.org>
- [2] <http://nagasm.org/ASL/paper/sigmus0808.pdf>
- [3] <http://nagasm.org/ASL/paper/sigmus0809.pdf>
- [4] <http://nagasm.org/ASL/paper/sigmus0905.pdf>
- [5] <http://nagasm.org/ASL/paper/sigmus201112.pdf>
- [6] <http://nagasm.org/ASL/mse/>
- [7] <http://nagasm.org/1106/installation/>
- [8] <http://nagasm.org/ASL/Arduino/>
- [9] <http://nagasm.org/ASL/Processing/>
- [10] <http://nagasm.org/ASL/SuperCollider/>
- [11] <http://nagasm.org/ASL/Propeller/>
- [12] <http://nagasm.org/ASL/Propeller2/>
- [13] <http://nagasm.org/ASL/Propeller3/>