

Intuino: ビジュアルプログラミングによる フィジカルコンピューティング開発支援環境の構築

姉崎 祐樹[†]

脇田 玲[†]

Intuino: Environment for the Development of Physical Computing with Visual Programming

YUKI ANEZAKI[†]

AKIRA WAKITA[†]

1. はじめに

現在、我々の住環境にはすでに多くのコンピュータが遍在している。家電やパーソナルコンピュータのみに限らず、iphone や wii には加速度等の様々なセンサが搭載されており、身体的なインタラクションを可能にしたプロダクトが広まりつつある。ユビキタス環境下では、フィジカルコンピューティング[1]を取り入れたプロダクトや空間デザインの需要が高まると考えられる。

本来、フィジカルコンピューティングを行うためには、マイコンの開発環境、電子回路、プログラミングが必要となり、こうした専門知識を網羅的に獲得していないユーザにとっては敷居が高い。そのため、フィジカルコンピューティングを手軽に行うためのツールやソフトウェアの開発が注目を集めてきた。しかしながら、アクチュエータの細やかな動きの制御やセンサの値による条件分岐、そしてハードウェアとプログラミングのデバッグ作業のためには、プログラミングの能力が必要になる。また、LED の明暗の移り変わりや細かなタイミングの制御を行う場合、感覚的な思考よりも数学的な思考が求められる。本研究では数学やプログラミングに精通していない学生や建築家、デザイナーでも、コーディングなしに視覚的にインタラクションのデザインを行える環境を構築する。

2. 関連研究

現在、フィジカルコンピューティングのためのプロジェクトは多く存在する。電子工作の手間を省き、センサやアクチュエータを PC から直に扱えるようにす

るものとして Phidget[2]が挙げられる。また、近年注目を集めている Arduino[3]や Gainer[4]は基本的な入出力を備えた I/O モジュールと、Processing/Wiring 言語を実装した開発環境をサポートしており、I/O 出力用のピンの設定や、入力値の扱いのためのライブラリ群を整備することによって、より簡易にプログラミングすることができる。しかし、プログラミング能力は不可欠であり、建築家や空間デザイナー、プロダクトデザイナーにとってはこうしたプログラミングを習得してもらうこと自体が負担となる。本研究で提案するシステムでは、センサやアクチュエータといった入出力デバイスの関係性をビジュアルプログラミングで行う機能、タイムラインによるアクチュエータの動作を作成する機能、センサの値やアクチュエータの制御リアルタイムに GUI に反映させる機能を有することにより、より直感的で簡易なフィジカルコンピューティングの開発支援を行うことを目的としている。以下の本論文では、開発したシステムの構成と機能について説明する。

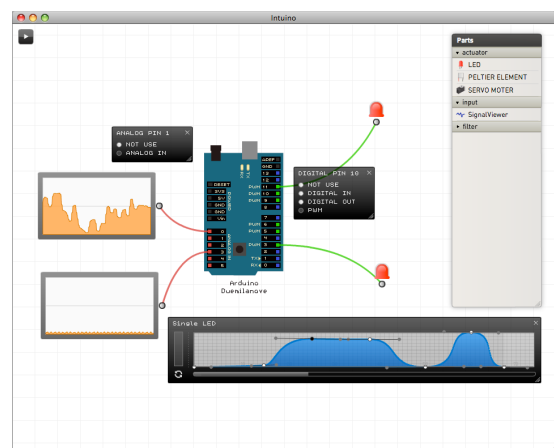


図1 システム概観

[†] 慶應義塾大学大学院政策・メディア研究科

3. システム構成

本研究では、システムを Adobe の提供する実行環境 AIR 上で動作するアプリケーションとして提供する。また、センサ・アクチュエータ類を扱うための I/O モジュールとして Arduino を使用し、Arduino がコンピュータと通信するためのプロトコルとして Firmata[5]を用いた。I/O モジュールとアプリケーションを通信させるために Funnel[6]サーバーを介することで、I/O モジュールとアプリケーションが常に連動する環境を用意した。

4. 機能

4.1 操作対象の視覚化

センサ、アクチュエータといった部品はイラストによって表現され、各部品をクリックすることで個々のプロパティが表示される。(図2左上)。Arduino の各ピンを入力として扱うか、出力として扱うか、もしくは PWM として設定するかといった操作はラジオボタンで行い、アクチュエータの動きはタイムラインで調整することができる。各部品に即して用意されたプロパティを設定していくことで、各部品の機能を理解しながら操作することが可能となり、基本操作に対するミスを軽減できる。

4.2 タイムラインによる動作生成機能

アニメーション作成や映像編集ソフトで使われるタイムラインを用いて、LED やサーボモータなどのアクチュエータの動きを編集する(図2右上)。タイムライン上にベジェ曲線を描写し、時間軸にそってパラメータの値を操作することで、細かな動きや滑らかな動きを作成することが可能になる。

4.3 リアルタイム連動

本アプリケーションと Arduino の I/O モジュールは常に連動し、Arduino に接続されたセンサ値がリアルタイムにアプリケーション上に反映される(図2左下)。アクチュエータの動作も、画面上で組み立てた通りに即座に実行され、ユーザは画面上に表示されるセンサ値を見ながら、トライ&エラーを短時間に繰り返し、インタラクシオンを組み立てて行くことが可能になる。

4.4 イベント用のラインドロ잉

インタラクシオンを組み立てるために、センサの値とアクチュエータの動きを連動させる必要がある。こうした割り込み処理や同期処理を行うためのインターフェイスを実装している。図2(右下)にあるようにセンサの値とアクチュエータのタイムラインをイベン

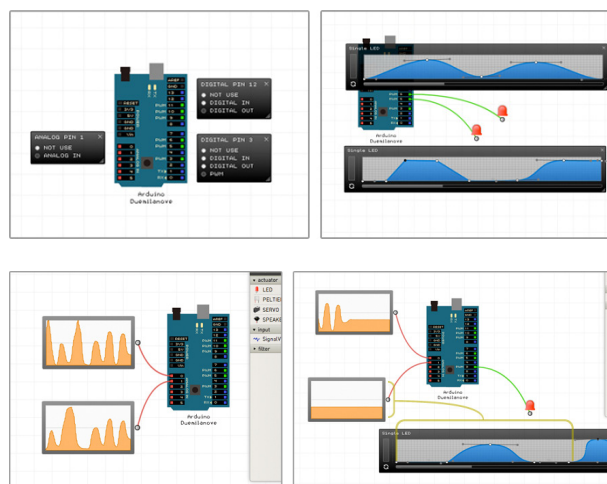


図2 インターフェイス例

ント用のラインで結ぶことで、センサの値に応じたアクチュエータの動きや、割り込み処理等の実現をする。例えば人が触れると回転するモータや、温度に応じてフルカラーLED の色を変えるといった処理を GUI 上で組み立てることが可能になる。

5. 今後の展望

本稿では、プログラミングを行わずに、タイムラインやイベント用のラインによるセンサやアクチュエータを制御する環境を構築した。現在、メディアアートやプロダクトデザインを志す学生を中心に、本アプリケーションを使用した作品づくりのワークショップを行っている。こうしたワークショップを通して、各機能の有用性や利便性を検証する。また、将来的には Gaiern や FIO[7]といった、Arduino 以外の I/O モジュールでも利用できるようにさらなる機能追加や応用性を考慮した開発を行っていく。

参考文献

- 1) Dan O'Sullivan and Tom Igoe, Physical Computing: Sensing and Controlling the Physical World with Computers, Course Technology, 2004
- 2) Saul Greenberg, Chester Fitchett: Phidgets: Easy development of physical interfaces through physical widgets, Proceedings of UIST 2001, pp.209-128, 2001
- 3) Arduino, <http://www.arduino.cc/>
- 4) 原田克彦, 小林茂: GAINER: メディア・アーティストのための再構成可能な I/O モジュール, 情報処理学会研究報告[音楽情報科学], vol.2005, No.129, pp.7-11, 2005
- 5) Firmata, <http://www.firmata.org/>
- 6) Funnel, <http://funnel.cc/>
- 7) FIO, <http://funnel.cc/Hardware/FIO>