

子どもは最強のUIテスター

長嶋洋一¹

「子どもロボット教室」の講師としてArduino電子工作の簡単なゲームを作って提供しているが、操作方法説明も不要でサッと慣れてしまう柔軟さに感服している。子どもは直感的アフォーダンスを試すための最良のUIテスターであり、また改善点をストレートに指摘してくれる最良のUIアドバイザーでもある。本発表では高機能LEDを活用した7種類のシステム事例を紹介し、デザインエンタテインメントの視点と共に検討したい。

Kids are the best UI testers

YOICHI NAGASHIMA¹

1. はじめに

筆者は2013年に「エンタテインメント科学とデザイン・エンタテインメント」と題した発表で、デザイナー側の**デザイン行為その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[1]、2014年に「『エンタテインメント科学』から『エンタテインメント学』へ」と題した発表で、工学でなく**科学としてのエンタテインメント研究**として、情報工学・認知科学・脳科学・人間工学・数理科学・複雑系科学など自然科学全般の領域から学会横断的・学際的な**エンタテインメント科学会**が設立されることを期待し、また心理学・美学・哲学・表象文化論・知覚認知学・文化人類学・臨床心理学・時間学などの人文科学がさらに加わった総合領域としての**エンタテインメント学会**が設立されることを期待する、と述べた[2]。本稿での報告は、ある意味でこの延長線上にあるとも言える。

筆者は現在**ジュニアロボット教室**及び**ロボティックプロフェッサー教室**の講師を担当しており、本来の教材だけでなくオリジナル**Arduino電子工作**作品を教室に持参して、余った時間に生徒(小3〜中2)に体験してもらっている。一種の簡単なゲームというような作品であるが、生徒は操作方法の説明なしに触り始めてサッと慣れてしまう柔軟さには驚かされる。デザインする側として「説明不要で楽しめる」ことを心がけているつもりだが、子どもはこの直感的アフォーダンスを試すための最良のUIテスターと言えよう。一方で生徒たちは、使い方の不自然さ・仕様の不満足なところ・改善点をストレートに指摘してくれる最良のUIアドバイザーでもある。本発表では2024年から2026年に開発した20種類ほどの作品のうち、高機能LEDを活用した7種類のシステム事例に限定して、メイキングと共に改良の経緯なども紹介し、開発者側の**デザインエンタテインメント**の視点と共に検討したい。

2. システムデザインの基本方針

これまで数多くの電子システムを開発してきた[3]筆者にとって、Arduinoスタンドアロンの電子工作というのはお茶の子さいさいの手慰みであり、プログラミング(昔ながらの技巧を凝らしたアルゴリズムへの拘り)は呆け防止のための脳活性化パズルとして楽しんでいる。今回紹介する7種類のシステムに共通した開発姿勢は以下である。

2.1. シーズ指向のデザイン

手元には多数のArduinoUNOやArduinoNanoやSeeedinoがあり、スイッチ・ボリューム・圧電スピーカ等の汎用部品も豊富にある。そこで新しいシステムを作るきっかけとして、「Amazon等で面白そうで安価な新デバイスを発見」が発端となる。過去にはPAWセンサを発見して多くの研究に発展した[4]が、今回紹介するシステムでは「高機能LEDモジュール」(3種類)が契機となった。この新デバイスを使って何か出来ないか、という**シーズ指向**の発想である。

2.2. 設計しないアドリブ開発

筆者の(悪い)癖だが、実験でも試作でも回路図を描かずに半田付けから始めるため、備忘録として簡単な回路図にまとめるのは最後になる。プログラミングもフローチャートは一度も描いた事が無く、いきなり新デバイスの実験から始めて、その後に中身を膨らませつつ整理して本番プログラムに成長させていく。即興を重視するLive Computer Music作曲家としての姿勢とも言えるが、エレクトロニクスの基本的知識[5][6]が不足している諸兄にはお薦めできない**アドリブ開発**(←これ自体が楽しい)である。

2.3. 空中配線と現物合わせ

ジュニアロボット教室のマイコンブロックの中身も実はArduinoであり、さらに**ロボティックプロフェッサー教室**では実際にArduinoUNOをむき出しで扱うために、ケースやユニバーサル基板などを使わず、見るからに**Arduino丸出し**重視の外見としている。ブレッドボードの信頼性を疑

¹ Art & Science Laboratory

うブレッドボード懷疑論者の筆者は、常に直接Arduinoに半田付け[7]を施しているの、基本的に**空中配線**である。Arduinoとスイッチ等とLEDモジュールとでおよそ3つのパートになるために、それらを固定して据え付ける台座には「段ボール+ガムテープ」という簡易素材などを使っている。これは筆者がKAWAI時代に学んだ**現物合わせ**という達人の技に起因している。

2.4. 「数理造形」デザイン

筆者はSUAC(静岡文化芸術大学)時代に、「サウンドデザイン演習」[8]と継承科目「メディア数理造形演習」[9]の中で、**数理造形デザイン**について力を入れてきた。学生の多くは「絵心」があってササッと美しい絵を描くが、デザインの領域には**数学的/物理的な合理性が導く「美」**もあり、プログラミングとの相性が良い。色々な表示パターンを固定データ化配列として持つのでなく、スマートなアルゴリズム記述によってライブ制御する「美」、というデザインエンタテインメントの世界も反映した。

3. システムデザインの詳細

ここでは7種類のシステム事例の詳細について述べるが、一応「作品」と呼ぶものの全てのシステムは「試作」であり完成品ではない。これは生徒からのリクエストに対応して改訂/成長すること、時には同じハードウェアに全く異なるArduinoスケッチを走らせて「別物」に化身する可能性を留保しているからである。互いを区別するために「仮の名称」が付いているが特段の意味は無い。それぞれの解説には、着想・セールスポイント・数理造形の視点・生徒の反応・クレームと改善・ゲーム的視点・教育的視点、などから特徴を強調するように意識してみた。

3.1. 1次元シューティングGame

最初の作品(図1)は、Amazonで見つけた安価なRGBテープLED: **NeoPixel** [10]を**シーズ**として活用する目的からスタートしたシンプルなシューティング(タイミング)ゲーム[11]である。ArduinoUNOのPWM出力ポート数は6本なので、それぞれRGBの発光強度を可変させるとRGB-LEDは計2個しか使えない。一方で**NeoPixel**は、WS2812B(RGB-LED)を1ブロックとして1メートルに144個ものブロックを連結させ、適当に切断して、Arduinoからはたった1bitのシリアル制御で全てのLEDブロックを何個でも個別制御できる。メーカーから提供されている**FastLED.h**というライブラリが煩雑な処理をBlackBox化しているので、プログラミングは単なる「配列」記述で完結している。このシステムの外観写真・操作YouTube動画・使用部品(シーズ)・回路図・Arduinoスケッチ(Source Program)は[11]から参照されたい。

電源のモバイルバッテリーを繋いで起動すると、長い方の**NeoPixel** (31個)の1箇所(ターゲット)がWhite高輝度で点灯し、別の1箇所刻々と変色発光する輝点は**NeoPixel**上を端から端まで等速度で往復し、端で折り返す際にはブザーの「反射」音が鳴る。プッシュスイッチはただ1個であり、何も説明しなくても、移動する輝点がターゲット上に来た瞬間にスイッチを押す、というシューティングは小学生で

も瞬時に理解できる。シューティング位置が外れた場合には「失敗」ブザー音が鳴って、もう1本の**NeoPixel**に赤い「失敗スコア」が複数、追加点灯する。失敗スコアが累積してOverflowする前にクリア(→ご褒美サウンド+全体の祝勝発光)する事が目標である。

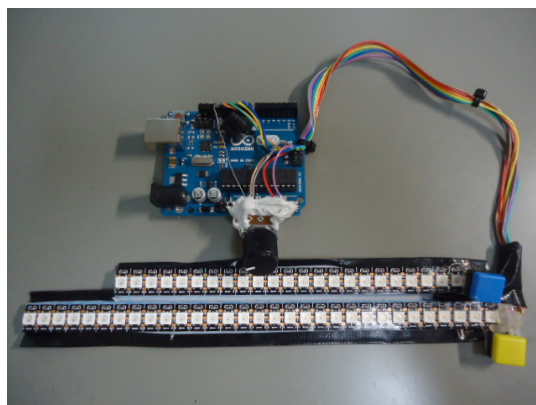


Fig. 1 1次元シューティングGame

Arduinoにはボリュームを1個付けていて、左に回すと輝点の移動速度は遅くなるものの失敗スコア幅が増加する。右に回すと輝点は相当な高速移動となって「ほぼ運ゲー」化するが、慣れるとリズム感を駆使すれば成功できる。生徒のリクエストを受けて、クリアした度にターゲットの位置をランダム設定するようにしたが、端っこに位置すると意外に難しくなるという発見もあった。このシステムの最大の特徴は説明不要で楽しめること、ボリュームで難易度を自分なりに設定できることのように、全7種のシステムのうち最初に作ったものののだが、最も多くの生徒を惹きつけ続ける(熱中させる)作品となった。

3.2. 1次元イジワル傾きGame

Amazonから仕入れた**NeoPixel**は144個/1メートルあったのでまだ余っていた。そこで45個(奇数というのがポイント)の**NeoPixel**を**30cmプラスチック定規**という土台に貼り、**SeeeduinoLite**というArduino上位互換ボード(チップ部品のため薄い)を定規の裏側に密着させた(図2)。3軸加速度センサ**MPU-6050**搭載のボード**KP-9150**から重力加速度を検出、定規全体を水平から左右に傾ける程度をセンシングして、**スイッチを使わないシステム**となった。このシステムの外観写真・操作YouTube動画・使用部品・回路図・Arduinoスケッチは[12]から参照されたい。

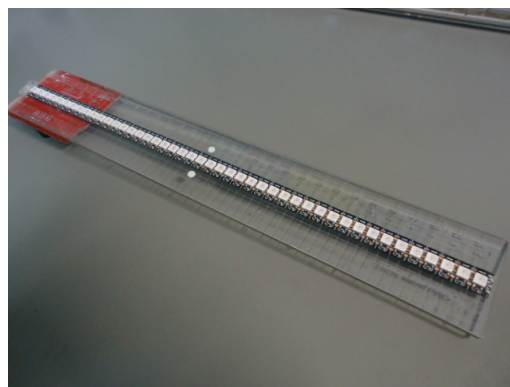


Fig. 2 1次元イジワル傾きGame

このゲームのタスクは、「ランダム色の輝点ポイントを移動させ、定規の中央点(ここでだけWhiteで発光)を**3秒以内に10回通過**させる」という目標である。3秒を超えると赤いLEDの「Gameover」がサウンドと共に表示されカウントはクリアされ、一方で定規の両端にあるレッドゾーンに入ると通過カウント数はリセットされる。何よりこのシステムの関係性でユニークなのが「傾きと輝点ポイントの移動速度との関係」であり、**大きく傾けると速度がグッと低下して、ほぼ水平に近いところでは高速で移動する**、という特性とした。地球上の通常感覚であれば大きく傾ければより速く移動する筈のところは真逆で遅くなり、一方で傾きが水平の近傍ではかなり高速に移動する。

これは生徒にとっても異常な仕様であり、**身体感覚と逆になるよ**、という説明が必要になった。コツとしてはリセット直後に定規を水平に把持したまま、輝点を微小に細かく左右移動させると往復10回通過は楽勝なのだが、つい日常感覚的に戻そうと定規を大きく傾けるとなかなか戻ってこなくて失敗する。この「勘所」をマスターするまで続ける生徒と、不自然さに耐えられずゲームを諦めてしまう生徒とに大きく分かれる結果となったのが興味深い。

3.3. 8*8マトリクス4列絵合せGame

この作品(図3)は、Amazonで見つけた安価な「8×8LEDドットマトリクス」が4ブロック連結したモジュール(切り離すことも可能)を使用、LEDは**単色**で色違いの3種類(赤・青・緑)があり、まずは印象的な青を採用した。モジュール上のコントローラMAX7219は定番であり、64個のLEDは基本的にON/OFFで全体の輝度しか調整できないために「あるパターンをLOOPアニメーション」という使い方を基本方針とした。メーカから提供されているMATRIX7219.hというライブラリが煩雑な処理をBlackBox化しているので、こちらも楽しい数値造形プログラミングとなった。このシステムの外観写真・操作YouTube動画・使用部品・回路図・Arduinoスケッチは[14]から参照されたい。

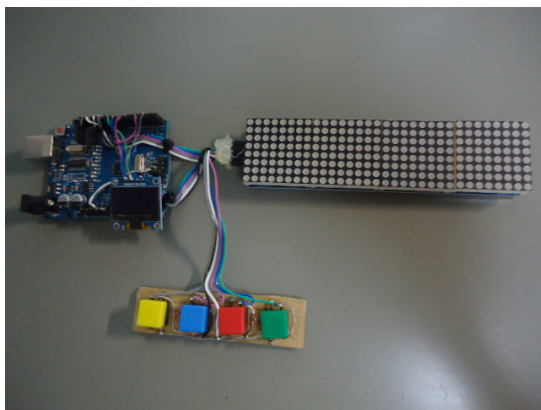


Fig. 3 8*8マトリクス4列絵合せGame

4ブロック連結したマトリクスLEDと対応して横一列にプッシュスイッチが4個並んでおり、それぞれ対応したマトリクスLEDのパターンを**ランダムに選ぶ**だけ、というゲームで「勝ち負け」も「達成(クリア)」も無い。しかしそれぞれのパターンは時間的に変化するアニメーションなので、類似するパターンがたまたま隣接すると連結して(位

相はスイッチONに同期)、全体として予想外の美しい動きになったりする**意外性を楽しむ**ゲームである。

デザインエンタテインメントの視点から言えば、このArduinoスケッチをプログラミングする上で、**パターンをTableデータとして持たない**事を重視した。34種類のパターン全てを「数値造形」的に**実時間アルゴリズム生成**していて、バリエーションやアニメーションのために、「移動」「回転」「反転」など「配列を配列に変換するための配列」のテクニックを駆使した。作り手として、試行錯誤で上手く動くたびに手応え(エンタテインメント)を感じながら開発したシステムと言える。シューティングなど運動神経重視のゲームが苦手な一部の生徒は、この動的「絵合せ」パズルにかなり没頭してくれた。

3.4. 8*8スロットGame

この作品(図4)は、前作「絵合せGame」に触れた生徒から「次はSLOT」と熱烈リクエストを受けて製作した。3つのパターンが密着しては不自然なので、モジュール基板をカットしてやや分離、繋がっていた基板間の配線を手作業で延長した。普通のスロットマシンでは3画面のそれぞれにスイッチがあるが、ここでは究極のシンプルさを狙ってスイッチを1個だけとし、左端から右に順次、回転アニメーションのパターンを止めていく仕様とした。このシステムの外観写真・操作YouTube動画・使用部品・回路図・Arduinoスケッチは[15]から参照されたい。

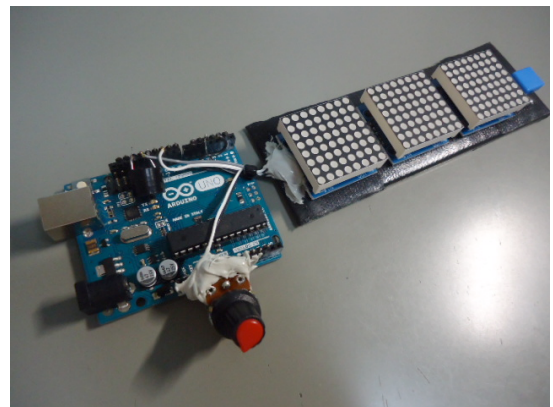


Fig. 4 8*8スロットGame

当初は3列それぞれのシリンダーのパターン(16種類)をランダム配置したが、恣意的/仮想的で美しくないとの苦情があり(←実に的確な指摘)、実世界に即して3列とも同一構成のシリンダーとして順に回るよう設定し、リセット時の起動位置(位相)だけランダムに変更した。ブザーからのサウンドはシンプルなもの、スイッチ操作にリアルタイム反応する関係性は好評だった。

3列それぞれのパターンは同期して自然にスクロールするので生徒はすぐに慣れた。失敗した時に1ラインずつ消えていくアニメーションに悔しがり、成功した時に全画面が反転点滅の様子を楽しんだ。1個のボリュームでシリンダーの回転速度が可変になっているのもお馴染みであり、最高速度あたりでは目視の識別不能で完全に「運ゲー」化するのもお約束の遊びとなった。上下左右の「矢印」が4個おきに登場する(秘)攻略法も密かに仕込んだ。

3.5. 8*8マトリクス縦横ずらし絵合せGame

この作品(図5)は、仕入れた「8×8LED-4連結ドットマトリクス」の3色のうち青と赤を活用して残った緑のために製作した。「4ブロック連結そのまま」「切り離して3個並び」の他に何かないか…と考えた末に、左右2個ずつを上下に密着させて外見上は16×16として、実際には左下1個の8×8パターンを上下鏡像反転と左右鏡像反転により**曼荼羅形式**に表示させる事にした。ただし、単に曼荼羅アニメーションするだけでないアイデアも思い付き、スイッチ1個に加えて3個のボリュームを並べた。このシステムの外観写真・操作YouTube動画・使用部品・回路図・Arduinoスケッチは[16]から参照されたい。

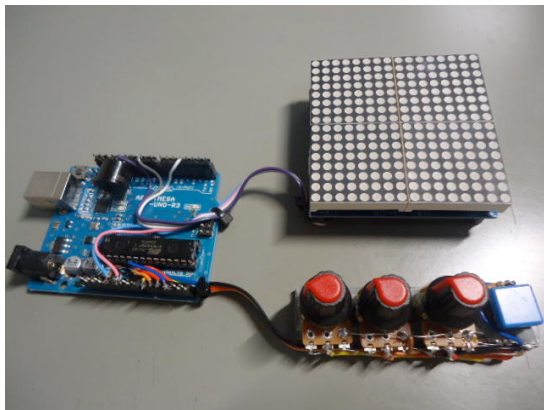


Fig. 5 8*8マトリクス縦横ずらし絵合せGame

このシステムで3個並んだボリュームのうち左端は、いつものような動作スピードであり、全体のアニメーション速度が変化するのは生徒も受け入れた。ただし残る2個のボリュームは、上下鏡像反転と左右鏡像反転の「**時間的位相遅れ**」の程度(左に回しきるとゼロ)というやや難しい概念のため、生徒への説明は必須となった。これを活用すると、4つに分割同期して鏡像パターンだったものが上下や左右でズレてくることで、「4画面の動きが別々に組み合わせあった全体が面白い(美しい)」というインタラクティブ数理造形Gameとなった。ただし、これに熱中する生徒はもっとも少なく、**刺激も報酬も無いGameの限界**を見た。

3.6. 2次元8*8シューティングGame

ここまで紹介した5種類のうち最初の2つは「RGB-LEDテープ」という1次元のディスプレイ、続く3つは「単色8×8ドットLEDモジュール」を複数個並べたものだった。これに続いてAmazonでたまたま見かけたディスプレイモジュール[17]は、なんと2次元の8×8マトリクス(64個)に個別点灯RGB-LEDを並べたもので、**680円**という驚くべき価格だった(現在は販売終了で入手困難)。さらに2次元と言っても、実は1次元の**NeoPixel**と同じWS2812Bチップを縦横に並べているだけであり、アドレス順にLEDを点灯させる実験によって、LEDの配置順は漢字の「弓」のように1行ごとに反転していると判明した。これにより(x, y)の2次元位置は簡易な変換行列とライブラリ**FastLED.h**によって容易にシューティング指定できることになった。

1次元versionと同様の2次元シューティングGameにする

ために、輝点の2次元移動の軌跡についてのパズルを検討した。輝点は刻々と色変化しながら2次元平面内を移動するのだが、単純に「角度45度で反射」するのでは元に戻ってきてしまう。そこで**反射して一周した4回目に1ドットずらす**というアルゴリズムを実験したところ、違和感なく輝点が平面内をくまなく移動する感じになった。この座標計算も、基準となる変換行列を90度ずつ回転させる事でスマートに記述でき、美しい「教材」となった。

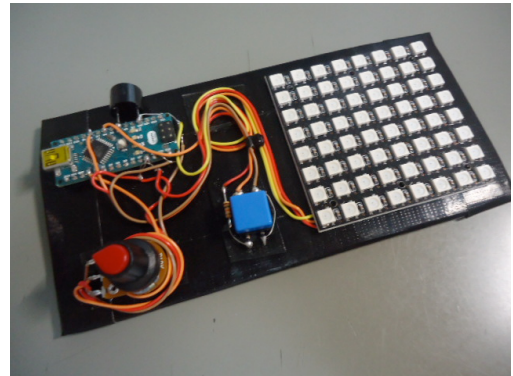


Fig. 6 2次元8*8シューティングGame

この作品(図6)はディスプレイの形状から全体を小型化するためにArduinoNanoを使用して、土台は段ボールをガムテープで巻き、1個のプッシュスイッチと全体の速度を変えるボリューム1個を両面テープで貼っただけの簡素な構成となっている。このシステムの外観写真・操作YouTube動画・使用部品・回路図・Arduinoスケッチは[18]から参照されたい。ちなみに1次元versionではボリューム最高速でも何とか視力追従できたが、こちらは容赦なく本気で無駄な足踏みを消したので、完全に「**運ゲー**」化した。

この頃になるとだいぶArduinoプログラミングの感覚も思い出してきたので、ちょっとした工夫を添えている。固定ターゲットの高輝度White輝点に対して、シューティングする輝点が2次元平面内をランダム色変化しつつ移動するのだが、1点だけでなくその**軌跡の残像**を5ステップの漸減輝度で実装している。これは生徒には気付き難かったが、指摘して説明すると「やるね!」と好評だった。

3.7. 2次元16*16消しものGame

上述の2次元ディスプレイモジュールよりも上位は、もはや16×16ドットという事になり、コストから期待していなかったのだが、どうも中国でWS2812BチップやRGB-LEDチップが大量生産されているのか、なんとフレキシブル基板に本当にWS2812B(RGB-LED)チップを256個搭載して、またまた**NeoPixel**と同じ制御でOKというものを発見した[19]。流石に3000円だったが、注意点として「256個LEDの全てをmax白色表示させると**15A以上を消費**(max 77W)して基板が溶けるので駄目」という本気のモジュールだった。

シーズとしてこのモジュールを仕入れてから実際に作るまでの期間は長かった。とりあえず色々なパターンで1ドットの高輝度白色ポイントが2次元平面内を移動したり、時にはチカチカとrandom発色させたり(同時点灯数には常に注意)、という実験試作の段階[20]を教室に持参して生徒に見せたりしたが、前作の8×8のアルゴリズムを単純に

16×16に増やして同様のGameに仕立て上げる、というデザイン方針だけは毛頭、頭に浮かばなかった。

この作品(図7)は、スイッチでなくセンサとしてPAWセンサ[4]を使う、という方針が決まってからようやく進展した。大判のディスプレイモジュールには裏にPCBシートを貼って補強し、本体部分はArduinoNanoとPAWセンサだけを、台座の磁気カード(NIME2009で参加したカーネギーメロン大学のGuestCard)上に載せた[21]。このシステムの外観写真・操作YouTube動画・使用部品・回路図・Arduinoスケッチは[22]から参照されたい。

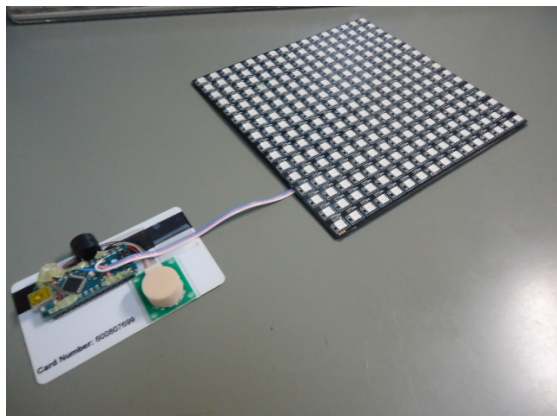


Fig. 7 2次元16*16消しものGame

筆者はこれまで、PAWセンサを用いた研究ではArduinoでなくmbed(NucleoF401RE)を活用[23]してきたので初のArduino制御となったが、250 μ s単位でのセンシングは容易に実現できた。そして2次元ポインティングの実現については、まず(1)PAW→Arduino→Maxという実験システムでMax上の2次元でポインティング出来ることを確認し[24]、(2)センサ値(PAWから届く4つのparameter)に対するMax内での重み付け値をArduinoに移植してポインティング点灯させ[25]、(3)全体をシューティングでなく「消しもの」という動作に変換した[26]。リセット時に全体がランダム発光するのだが、その最大輝度は33/255とかなり少なくても十分に明るいと確認した。

これは最近の作品なのでまだ教室で生徒が体験したサンプルは多くないのだが、やはりシューティングの爽快感に対して、地味に「消していく」というターゲットは面白くないらしい。さらに、慣れないPAWセンサ(筆者から「壊さないようにそっと触って」と言われるので、萎縮して画面全体をなかなか掻破しきれない)の操作性も、子どもというターゲットには再考が必要であると思われる。Game性を高めるために、ぐずぐずしていると消えたLEDがまた点灯して増えるようにしたのだが、この頻度を厳しくすると本当に萎えた気分になる事も実感した。

4. その他のArduinoシステム

本稿で紹介した7種のArduinoシステムは、共通点としてLEDモジュールを活用した「光りモノ」であり、スイッチ等の操作はかなりシンプルで、ブザーによる操作音も全て鳴らすようにしたものである。その他にもこれまでに試作したシステムがあるが、Denmarkの友人に依頼されて

YouTube紹介情報ページ[27]を設置して日々、逐次追記しているので、興味のある方はこちらを参照されたい。

ロボット教室(ミドル/アドバンス)ではマイコンブロックの中にArduino互換が入っているものの見えないので、生徒にとっては、教室で使うマイコンブロックに筆者が独自改造インターフェースを接続したシステム(図8の4連光センサ楽器[28]、図9のスライドボリューム楽器[29])の方が身近だった。ただし教室でタブレット開発するプログラムは「変数」概念すら無いなど非力であり、Arduinoむき出しでも高機能な筆者のシステムは、より好評だった。

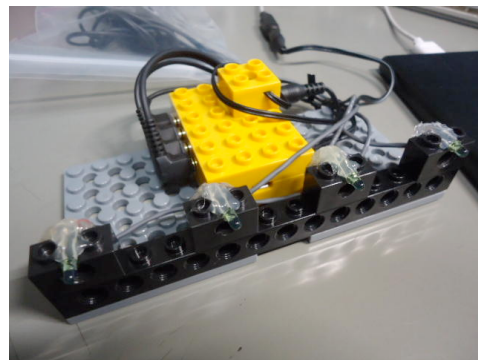


Fig. 8 4連光センサ楽器

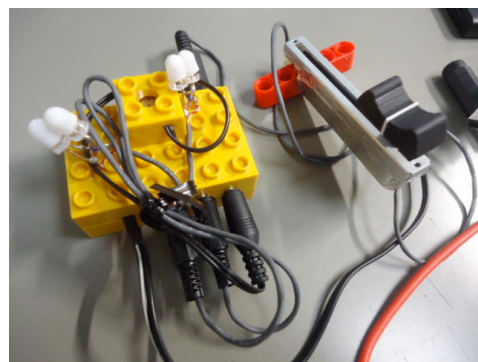


Fig. 9 スライドボリューム楽器

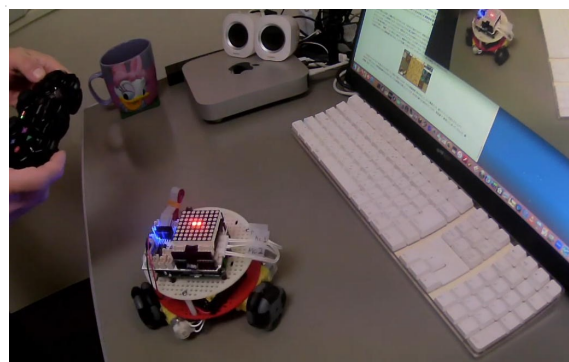


Fig. 10 OmniWheelRobot

上位のロボプロ教室では実際にパソコンでArduino開発している。ただしHumanAcademyオリジナルのArduinoシールドの詳細は伏せられて、基本的には提供されたサンプルArduinoスケッチ(BlackBox的)をHumanAcademyライブラリと共にそのまま走らせて、ごく一部だけソースを改訂してC言語に慣れていくという教育スタイルである。複数のタスクを時分割多重化するテクニック等はかなり後まで登場しないので、筆者は図10のオムニホイールロボット(PS2リ

モコンで操縦する3輪ロボット)[30]に、テキストではまだ登場していない8×8ドットマトリクスLEDを搭載して、リモコン操縦と同時にアルゴリズム(無操作時には規則的、操作時にはランダム)点灯を並列動作させるようオリジナル改造した例を示すと共に、Arduinoスケッチを生徒に配った。前節の7種のシステムについてもプログラミングとしてはテキストのレベルを大幅に超えているが、参考資料としてそのArduinoスケッチを生徒に提供することで、アルゴリズム教育の有益な教材となった。

5. 最強のUIテスターとの日々

学校と違ってロボット教室というのはクラス(レベル)と生徒の学年は無関係であり、生徒の取り組み姿勢も達成度もそれぞれで、毎回「テキスト」に従ってロボットを組み立ててプログラミングするが、時には未完成のまま時間切れ(→終了)になる事もある。時間が余れば自由なカスタマイズ(改造)を推奨しているが、生徒によってはタブレットでYouTubeを観たりKUMONドリルに没頭する。

そんな運営マニュアル化されたロボット教室で、筆者は勝手にパーツを改造したりArduinoオリジナル作品を別途に持参する、やや浮いた存在である。最初は面白がってGameしたものの早々に飽きる生徒もいれば、「次は何??」と新作をリクエストしてくる生徒もいる。使い勝手に文句や注文を出したり(これがかなり適切)、試作段階のシステムにかなり無茶な仕様希望を出したり、インタラクション自体が筆者にとって無類のエンタテインメントである。

スマホやタブレットの操作に慣れている世代の生徒にとって、筆者の提供する「ディスプレイとスイッチと速度設定ボリューム」だけの超シンプルなシステムはかなり古風で貧相に見える。しかし、それでも操作性を即マスターして没入する姿には、正に「子どもは王様」(最強のユーザ)という基本を思い出させてくれる。気付かなかった不自然さを指摘してくれたり、逆に意図した仕様の反応を純粹にモニタできるという意味で、子どもは最強のUIテスターというのは日々積み上げられた実感である。

6. おわりに

ロボット教室の講師という立場で、Arduinoを活用した簡単なシステム(一種のゲーム)を手作りする(実感としては趣味のデザインエンタテインメント)日々から、各種システムのメイキングやUIの変遷などを報告した。操作をマスターする生徒の柔軟さに驚かされ、時には使い方の不自然さ・仕様の不満足なところ・改善点をストレートに指摘する生徒の感性にたじろぐ日々はなかなか楽しく、今後もさらに新しいシステム開発を追求してみたい。

生成AIについて

和文タイトルの選定にChatGPTの助けを借り、英文タイトルを作成するためにWeb版Google翻訳を用いた以外、本稿では全てにおいて一字一句たりとも、所謂「生成AI」の支援を用いていない事をここに明記する。

(追補)

本稿のここまでを執筆(ポスターも作成)してEasyChairにuploadした後に新たな動きがあって改訂したので、参考として付記しておく。Amazonで見つけた面白そうなデバイス[31]に触発されて「8番目の新作」[32]が完成したのでEC2026のデモ用に持参予定である。同時期に筆者が担当していた新科目「生成音楽論」[33](全7週にわたって、音楽生成AIのsunoを通して音楽全般を見つめ直す)に追われて作業が前後したために追補となった事をお詫びしたい。

参考文献/URL

- 1.<https://nagasm.org/ASL/paper/SIGEC201305.pdf>
- 2.<https://nagasm.org/ASL/paper/sigec201408.pdf>
- 3.<https://nagasm.org/ASL/ASL.html>
- 4.<https://nagasm.org/Sketching/PAW-double.html>
- 5.<https://nagasm.org/ASL/sensor02/index.html>
- 6.<https://nagasm.org/ASL/mse/index.html>
- 7.https://nagasm.org/1106/kiso_E/Soldering2.html
- 8.<https://nagasm.org/1106/sounden/index.html>
- 9.<https://nagasm.org/1106/mediasurien/index.html>
- 10.<https://nagasm.org/ASL/RRR04/fig2/034.jpg>
- 11.<https://nagasm.org/ASL/EC2026/index.html#01>
- 12.<https://nagasm.org/ASL/EC2026/index.html#02>
- 13.<https://nagasm.org/ASL/RRR05/fig2/021.jpg>
- 14.<https://nagasm.org/ASL/EC2026/index.html#03>
- 15.<https://nagasm.org/ASL/EC2026/index.html#04>
- 16.<https://nagasm.org/ASL/EC2026/index.html#05>
- 17.<https://nagasm.org/ASL/RRR13/fig2/015.jpg>
- 18.<https://nagasm.org/ASL/EC2026/index.html#06>
- 19.<https://nagasm.org/ASL/RRR13/fig2/017.jpg>
- 20.<https://www.youtube.com/watch?v=76K13pMANtU>
- 21.<https://nagasm.org/ASL/EC2026/DSC00032.JPG>
- 22.<https://nagasm.org/ASL/EC2026/index.html#07>
- 23.<https://nagasm.org/Sketching/index.html>
- 24.<https://www.youtube.com/watch?v=o8DKNgNeqSY>
- 25.<https://www.youtube.com/watch?v=2m6tvYyRPTg>
- 26.<https://www.youtube.com/watch?v=EgVVvG254TE>
- 27.https://nagasm.org/ASL/Robot_Arduino.html
- 28.<https://www.youtube.com/watch?v=V5MD933DqN8>
- 29.<https://www.youtube.com/watch?v=u3QCQKuiLrw>
- 30.<https://www.youtube.com/watch?v=ltvvIWAZa2I>
- 31.<https://nagasm.org/ASL/RRR13/fig2/016.jpg>
- 32.<https://nagasm.org/ASL/EC2026/index-add.html>
- 33.<https://nagasm.org/ASL/KSU/2026/index8.html>