

2色での入力・表示可能なマトリクス LED ブロックを用いた タンジブルなプログラミング教材

川崎 基輝^{1,a)}

Nguyen Tran Huu Nguyen^{1,b)}

秋田 純一^{1,c)}

近年、プログラミング教育の必要性が注目されており、プログラミングを学ぶことで得られる経験や考え方の、IT 業界に限らず社会全般における有用性が認識されつつある。プログラミングの入門過程においては、分かりやすさを重視したプログラミング言語が必要であり、多くの研究と実践がある。本稿では、著者らがこれまで開発してきた 2 色の入力と表示が可能なマトリクス LED を用いたブロック型デバイス bLED Tile を用い、ブロック毎に可視光によって定義可能な命令と変数を持ち、それらのブロックを組み合わせることでプログラムを組む、命令と数値を動的に書き換え可能な、タンジブルなプログラミング言語システム Tiles について述べる。

Tangible System for Programming Education Using Matrix LED Blocks with Bi-color Pattern Draw and Display Capabilities

Mototeru Kawasaki^{1,a)}

Huu Nguyen Nguyen Tran^{1,b)}

Junichi Akita^{1,c)}

In recent years, the need for programming education attracted attentions and the worth of learning programming experiments is widely recognized not only in IT industries, but also in social lifes. There are a lot of researches on easy-to-use programming environment. In this paper, a tangible programming language, named "Tiles", is described, which uses the matrix LED block device with bi-color pattern draw and display capabilities the authors have developed. It implements the tangible programming environment composed instruction and parameter blocks the user can define by visible light, with dynamic reconfigure capability.

1. はじめに

近年、プログラミング教育の必要性が注目されており、プログラミングを学ぶことで得られる経験や考え方は、IT 業界に限らず社会に広く役に立つものとして着目されている。そのため、義務教育への導入や幼少時にも学べるような環境の構築が始まりつつある。

プログラミング教育の導入において、C 言語や Java などの Character User Interface(:CUI)は子供には理解が難しいため Graphical User Interface(:GUI)環境をもったプログラミング言語によるプログラミング教育が行われることが多い。しかし、この GUI によるプログラミングも、画面に向かってマウスやキーボードなどの入力インタフェースにより操作を行うため、導入部での敷居の高さが残っている。

そこで最近では Tangible User Interface(:TUI)による直接触って動かすことができるプログラミング言語がいくつか開発され、GUI 環境のプログラム言語よりも理解しやすいとされている[1]。

著者らはこれまで、可視光による入出力可能なドットマトリクス LED ブロック"LED Tile"を開発し、とその拡張

やアプリケーションを開発してきた[2], [3], [4], [5]。本稿では、2色での入力と表示が可能なマトリクス LED デバイス bLED Tile[5]を用い、各ブロックに命令の割り当てや命令の書き換えが可能な TUI のプログラミング言語 "Tiles"の開発を行った。bLED Tile を用いたタンジブルなプログラミング教材が、実際に触れることによる理解のしやすさや動的に書き換え可能なことによる自由度などにおいて、より理解や導入がしやすいことが期待される。

2. 関連研究

教育用に開発された子供向けプログラミング言語は使いやすさを重視される。シーモア・パパートが開発した LOGO は、画面上に映るタートルと呼ばれるものと交流するためのコンピュータ言語の総称である。タートルとはコンピュータの画面に表示される三角形の呼称であり、キーボードでコマンドを入力することで操作できる。タートルは移動した軌跡に線を描画する事ができ、それにより様々な図形や絵を描くことができる[6]。LOGO のタートルに影響された言語は多く、視覚的なフィードバックを得られるため理解しやすく、デバッグもしやすい。Scratch は MIT メディアラボにより、低年齢向けプログラミング教育のために開発され、画面上で絵とそれに対する命令を表したブロックを組み合わせることでアニメーションやゲームなどを作成できる。基本的な移動・回転の他、繰り返しや条件分岐などの制御ブロックも存在し、正しい構文の書き方を

1 金沢大学

Kanazawa Univ.

a) m_kawasaki@ifdl.jp

b) nguyen@ifdl.jp

c) akita@is.t.kanazawa-u.ac.jp

覚えることなくプログラミングができる。ブロックの操作はマウスで動かし、キーボードによる数字入力を使用する[7]。Scratchはプログラミング導入において、世界で広く使われており、作品も多く公開されている。複雑に命令を組み合わせることでゲームも作成できるため、使いやすいながらも様々なものを作れる楽しさがある。原田康徳によって開発された VISCUIT はコンピュータ上で描いた絵を配置し、メガネと呼ばれる唯一の命令を用いてアニメーションやゲームを作成できる。メガネには2つの丸枠があり、2つの枠に入れた絵の位置関係によって絵の動作が決定される。メガネを複数使用することで様々な動きを実現できる。配置はマウスやタッチパネルによって行う[8]。命令が唯一であることから理解のしやすさがありながら、子供の想像力が養われる。上述した3つの例はGUIによるプログラミング言語である。GUIでは移動や回転、繰り返しや分岐制御の他、自ら絵を描いて、それ自体に命令を付加して動かすことができるというものが多い。また、TUIと比べて複雑な命令を組みやすく、ゲームを作成できるものもある。

教育向けに作られたプログラミング言語 Tern は、木製の命令ブロックを組み合わせてロボットを制御することができ、TUIを実現している。制御ブロックは移動や回転、ピープ音や繰り返しの命令を行うことができ、作成したプログラムはカメラで画像解析され、ワイヤレス通信でロボットに送信される[1]。TurTanはLOGOのようなタートルを持ち、机上に命令ブロックを置き、その位置や前のブロックからの角度によってタートルの動きを制御し、机上に様々な図形を描画させることができるプログラミング言語である。机上に映し出されるタートルを指でスワイプすることで移動、ピンチイン・ピンチアウトで拡大縮小もできる[9]。Scratchをより分かりやすくするため、画面上の命令ブロックの一部を実際に触れるアクリル製の命令タイルとして作り、TUI化している例もある。プログラムの作成はアクリル製の命令タイルで行い、PCに接続されたUSBカメラがそのブロック列を画像解析し画面上の車の動きを決定する[10]。TUIの言語は、実際に触るものでプログラムを組むという前提から、カメラによる画像解析を用いるものが多い。その関係でカメラの撮影範囲に邪魔なものや他の命令ブロックが入ってはならず、どうしてもある程度広い場所が必要となる。

3. システム構成

3.1 概要

本節では開発した2色の入力・表示が可能なブロックデバイス“bLED Tile”を用いたプログラミング言語“Tiles”の概要について述べる。“Tiles”では、移動や回転などの命令と、命令に付加する数値を持ったブロックを組み合わせて一つの動きを組み立てていき、表示部に表示された画像や部品（タートル）を組み立てた動き通りに動かすシステム

となっている(図1)。ブロックの組み合わせを実際に触ってプログラムを組み立てることができ、動的に命令と数値の書き換えが可能である。可視光により命令と数値を書き換えることができるため、キーボードやマウスなどを使用しないまま、PCを介さずにブロックの機能を自らの手で変えられる。この動的書き換え機能により、“組み立て”だけではなく“書き換え”によるプログラミングをTUIにて実現する。

Tilesはプログラム部と表示部で分かれており、プログラム部に、書き換え可能なブロックデバイス bLED Tileを用いる(図2)。今回、表示部としてPCを用いており、ブロックと通信をして受け取ったデータをPC側で処理し表示する(図3)。この処理のアプリケーションはVisual C#で実装した。

また、表示部は今後ブロックで構成することも考え、ブロックのみで独立してプログラミングと表示を行えるようにする予定である。

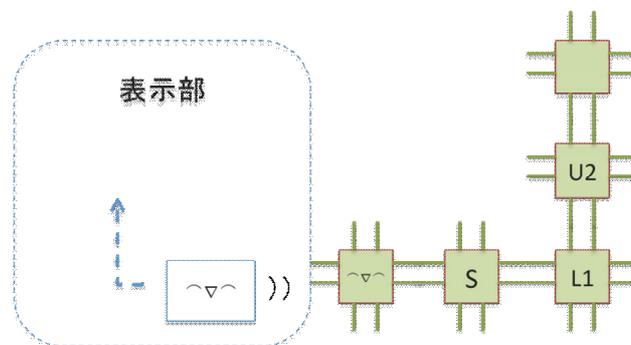


図1 Tiles 概要



図2 Tiles プログラム部

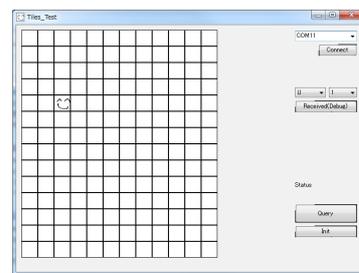


図3 Tiles 表示部

3.2 2色の可視光で書き換え可能なブロック

本節では、著者らが開発した、2色入力・表示が可能な8×8のマトリクスLEDブロックデバイス bLED Tile(図4)の構成について述べる。このブロックデバイスは赤と緑の2色で表示ができ、2色の入力を判別して動作する。複数のブロックを用い、ブロック間で通信し合うことができる。

3.2.1 ハードウェア構成

bLED Tileは、2色マトリクスLED(QFM-1088ARG)とマイコン(Atmel社 ATmega640)、LEDドライバ(Texas Instruments社 ULN2803A)、および加速度センサ(Freescale MMA7455)を搭載する。マイコンとマトリクスLEDとは図5のように接続されている。1ブロックのサイ

ズは縦横がマトリクス LED のサイズと同じ 32mm×32mm×22mm である。

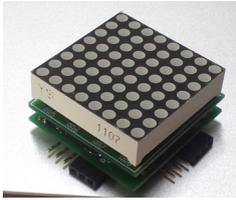


図 4 bLED Tile

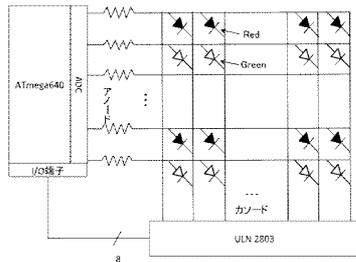


図 5 bLED Tile 接続図

3.2.2 動作原理

LED は光を当てることで微弱な電流が発生することが知られており、その電流を検知することで光センサとして扱うことができる。マトリクス LED を入出力デバイスとして使用する場合の動作原理を図 6 に示す。図 6 のようにマトリクス LED の行と列を R_i と C_j として、それぞれマイコンの I/O 端子に接続されている。また、アノード側の端子（図 6 では R_i 側）にはアナログ入力の端子に接続する。

(1)~(6)の動作によって入出力動作を実現する。

動作は光センサ動作と表示を各列時分割で行う。入力されたパターンと出力するパターンは独立しており、様々な実装に対応できる。(4)の読み取り時間は、LED の発光表示時間を十分にとるために、得られる電圧値と表示の強さを考慮して設定する必要がある。

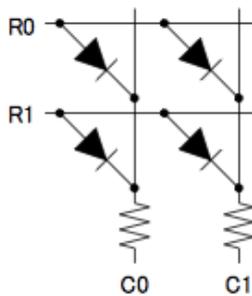


図 6 動作原理

- (1) R_i, C_j すべてに '0' を設定する (リセット動作)
- (2) C_0 を '0' のまま、他の C_j と R_i を入力モードに設定する
- (3) C_0 につながった LED に、当てられた光によって光電流が発生し、それによりそれぞれの R_i に電圧が発生する (蓄積動作)
- (4) 一定の時間 (約 1ms 程度) 後、それぞれの R_i にかかっている電圧を A/D コンバータで読み取り、閾値を超えているかどうかを判断する (入力読み取り動作)
- (5) C_0 を '0' のまま、その列のパターンに基づいた R_i を '0' か '1' に設定する (出力動作)
- (6) それぞれの列で(1)~(6)を繰り返す

3.2.3 2色の判別

緑色の光は赤色の光よりも波長が 100nm ほど短く、光の

波長が短いほど光がもつエネルギーは高い。受光側のダイオードのエネルギーバンドギャップを超えるエネルギーが与えられた場合に電子が導電帯に励起し、光電流が流れるため、原理的には LED に、その LED が発する光の波長よりも短い波長の光を当てることで光電流が流れることになる。ただしこの光電流によって LED の両端に発生する電圧は、LED のもれ電流や接合容量などの素子パラメータに強く依存する。

今回用いるマトリクス LED(QFM-1088ARG)に赤と緑の可視光として高輝度 LED の光およびレーザーポインターの光を照射して測定した電圧値を表 1 に示す。マトリクス LED の赤 LED に、光源として光強度が強く波長の短い赤 LED や緑 LED を当てると十分な電圧値が得られた。また、マトリクス LED の緑 LED に、赤 LED を当てても電圧値は得られず、光強度の強い緑 LED を当てた場合だけ電圧値が得られた。この特性を用い、赤 LED と緑 LED が反応した場合、緑色が当てられたと判断し、赤 LED のみが反応した場合、赤色が当てられたと判断することができる。

表 1 可視光照射時に各 LED に発生する電圧値

	Red LED	Green LED
Red Laser (650nm)	1.5[V]	0.0[V]
Green Laser (532nm)	1.2[V]	1.7[V]
Red LED (650nm)	1.2[V]	0.0[V]
Green LED (525nm)	1.0[V]	1.5[V]

3.3 プログラミング言語 Tiles の設計

3.3.1 命令とブロック間の通信

命令として S, E, A, B, O, U, R, D, L, G, J が用意されている。それぞれの命令の意味は表 2 に示す。また、数値は 1~9 を使用する。

表 2 命令一覧とその意味

命令 (使用文字)	意味
Start (S)	プログラムの始まり
End (E)	プログラムの終わり
Repeat Start (A)	繰り返しの始まり
Repeat End (B)	繰り返しの終わり
Junction (O)	分岐点
Up (U)	上方向に移動
Right (R)	右方向に移動
Down (D)	下方向に移動
Left (L)	左方向に移動
Right Turn (J)	右に 90 度回転
Left Turn (G)	左に 90 度回転

ブロック間の通信は、通信の際に親と子の関係が設定され、各々が図 7 に示すような”子に問い合わせ”, ”受信 OK”, ”命令と数値を親に渡す”, ”受信 OK”のデータを送り

合うことで行う。自分の命令を送り終わった後は、親からの問い合わせをそのまま子に送り、子からの命令をそのまま親に送る動作をする(図8)。この際データはすべて(‘命令’, 数値)という形で送られている。

親と子の関係は、問い合わせた側が親、問い合わせられた側が子となり、ブロック毎に親と子は1つまで持てる。

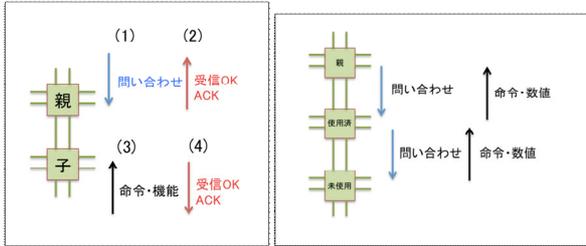


図7 親子間の通信手順 図8 使用済みブロックの機能

3.3.2 命令の書き換え

ブロックの命令や数値の書き換えを行う際は、ジェスチャ操作により書き換えを行う。図9、図10に示すように、それぞれに対応したジェスチャを行うことで、赤なら命令を、緑なら数値を書き換えることができる。

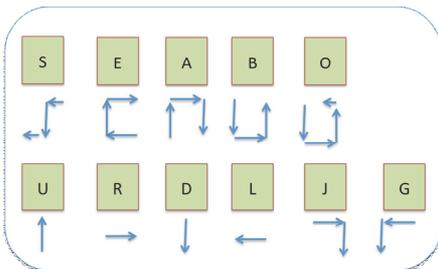


図9 命令ジェスチャ対応表(赤入力)

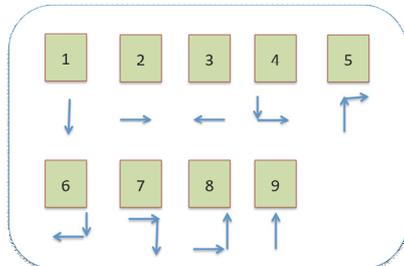


図10 数値ジェスチャ対応表(緑入力)

4. 予備的な実験と改善方針

今回開発した”Tiles”を試行的に用いたところ、以下のような改善すべき点が明らかとなった。

まず、今回は命令の表現としてアルファベットを用いたが、より理解しやすいように矢印や記号にすべきであると思われる。

またタイトルは今回 PC 側であらかじめ設定した画像に固定されているが、bLED Tile 上で描いた絵をそのままタイトルに使用できるようにすることでより関心や興味が惹ける可能性があると思われる。

また今回は命令について上下移動、回転、制御命令を用

意したが、他にもあると有用な命令があるとも考えられるため、実際の教育実践をとおして必要な機能の選定を行うべきであると考えられる。

また命令の順序が分かりやすいように、ブロック毎に未使用か使用済みかを示すドットや、通信速度を遅くして通信していく課程を辿れるようにする、といった工夫も考えられる。

5. まとめ

本稿では、2色の可視光で入力と表示が可能なブロック型マトリクス LED デバイス bLED Tile と、それを用いたタンジブルなプログラミング言語”Tiles”について述べた。

今後は、4節で述べたような改善を行った後、GUI プログラミング言語との比較実験を通して、”Tiles”の特長である動的書き換え可能な TUI のプログラミング言語と、GUI のプログラミング言語との比較検討を行う予定である。

参考文献

- [1] Michael S. Horn, et al. : Comparing the Use of Tangible and Graphical Programming Languages for Informal Science Education, CHI'09 Proceedings of the SIGCHI Conference on Human Factors in Computing Systems, pp.975-984 (2009)
- [2] J.Akita: Interactive Matrix LED Unit with Functions of Pattern Drawing and Size Extension, IPSJ Journal, 52(2), pp.733-736 (2011).
- [3] J.Akita: Matrix LED Unit with Pattern Drawing and Extensive Connection, Proceedings of SIGGRAPH2010, Emerging Technologies (CD-ROM) (2010).
- [4] J.Akita: Interactive Block Device System with Pattern Drawing Capability on Matrix LEDs, The ACM SIGCHI Conference on Human Factors in Computing Systems (CHI) 2012, Interactivity (in DVD-ROM) (2012).
- [5] Nguyen Tran Huu Nguyen・川崎基輝・秋田純一, Extended LED Tiles: Large and Bi-color Matrix LED Unit with Pattern Drawing Capability, インタラクシオン 2013 論文集(CD-ROM), 2EXB-42, (2013)
- [6] シーモア・パパート: “マインドストーム—子ども, コンピューター, そして強力なアイデア”, 未来社, 東京(1982)
- [7] RESNICK M, et al : Scratch : Programming for All , Communications of the ACM, 52, 11, pp.60-67, (2009)
- [8] 原田康徳: 子供向けビジュアル言語 viscuit と そのインタフェース, 情報処理学会研究報告- ヒューマンインタフェース, 2005, 116, pp.41-48, (2005)
- [9] Gallardo, D., Julia C. F., & Jorda S: TurTan: a Tangible Programming Language for Creative Exploration, Third annual IEEE international workshop on horizontal human-computer systems (TABLETOP), pp.89-920, (2008)
- [10] 八城 朋仁・迎山 和司: 物質プログラミング: 物質によるプログラムの可視化と開発環境の制作, 映像情報メディア学会技術報告 37(17), pp.223-226, (2013)