

# 多様なデザイン性を考慮した基板上の部品配置支援システム

長谷川 晃汰<sup>1,a)</sup> 塚田 浩二<sup>1</sup>

**概要:** 本研究では、電子回路におけるプリント基板設計を支援するための自動素子配置システムを提案する。基板上に部品を配置する際には、部品間の配線のしやすさや電気的特性のみでなく、外観の美しさも重要となり、これらをバランスよく基板設計に取り入れるのは困難である。そこで本提案では、オープンソースの基板設計ソフト KiCAD を拡張し、配線の容易さと外観の美しさを考慮したデザイン案を生成することでユーザの試行錯誤を支援するシステムを提案する。

## 1. 背景

電子工作で基板設計を行う際、設計者は基板の部品配置に苦勞する。配線が簡単であるか・はんだ付けがしやすいかといった作業性や、動作に支障をきたさないかといった電気的特性を考慮しながら、外観的にも美しい基板デザインを作っていく必要がある。

電気的特性としては、電源周りの部品をなるべく一個所に集めること・バイパスコンデンサを電源ピンの近くに配置すること・各ラッツネット（配線すべきパッド同士を結ぶ白い線）がなるべく交差しないように部品を配置するなどが挙げられる。

デザイン性については、部品同士を整列して配置したり、同種の部品をまとめて配置する等の一般的な観点以外にも、レジストでキャラクターや模様等を描くプリント基板アート<sup>\*1</sup>もよく知られている。さらに、プリント基板をキーホルダーやアクセサリ等の一部に活用する作例も多数存在しており、プリント基板はデザインの側面でも注目を集めている。

本研究ではプリント基板の部品配置作業に着目し、多様なデザイン性を考慮した部品配置案を生成することで、ユーザの試行錯誤を支援するシステムを提案する。

## 2. 関連研究

これまで、回路設計や基板設計を支援する研究が多数行われてきた。

J. Garza ら [1] は、電子機器の開発におけるプロトタイプと量産のギャップに着目し、PCB 対応のブレイクアウトボードを利用することでプロトタイプから製品への

効率化を図るシステムを提案している。

Leo ら [2] は、テキストエディタと GUI 上のグラフィカル表示を組み合わせて、HDL を用いた基板設計を支援するシステム SVG-PCB を提案している。Web ブラウザ上で動作するシステムを構築しており、リアルタイムにテキストとグラフィカル表示が同期されることで、設計をスムーズに進めることができる。

Chen ら [3] は電子部品のデータシートに含まれる図表を分類する ECS-YOLO モデルを提案しており、PCB の設計段階でデータシートを効率的に活用する支援を行っている。

Kim ら [4] は、ブレッドボードの下に LED マトリクスを組み込んだ SchemaBoard を提案している。アプリケーション上で部品のイラスト付きの回路図を提示することで、部品を選ぶ工程の支援を行っている。さらに部品の配置場所を光のパターンで視覚化することで、回路図からブレッドボードへの展開に必要な時間と作業ミスの削減を図っている。

Qi [5] らはブレッドボードや PCB に代わるプロトタイプ作成の新たな方法として、紙上で導電性箔テープを使用した回路設計手法の Sketching in Circuits を提案している。部品の実装を透明テープで固定するため取り外しが容易であり、紙を切り離すことで回路全体を別の回路に再構成することも可能である。

本研究では、基板設計における部品配置の工程において、配線の容易さ／外観の美しさ等を考慮した複数のデザイン案を生成することを独自性とする。

<sup>1</sup> 公立はこだて未来大学

<sup>a)</sup> b1021251@fun.ac.jp

<sup>\*1</sup> プリント基板アート:<https://www.kpd-jp.biz/art/>

### 3. 提案

本研究では、無料基板設計ソフト KiCAD を使用したプリント基板の自動素子配置システムを提案する。設計された回路に対して複数の作業性・電気的特性・デザイン性を考慮した配置パターンを提示し、多様なデザイン性を持つ基板設計を支援する。具体的には、素子の座標や角度・素子同士の重なりなどの基本パラメータをもとにユーザーが選択したデザインルールに基づいて、素子を配置していく。素子配置のデザインルールに関わる基本パラメータについて図1と表1に示す。

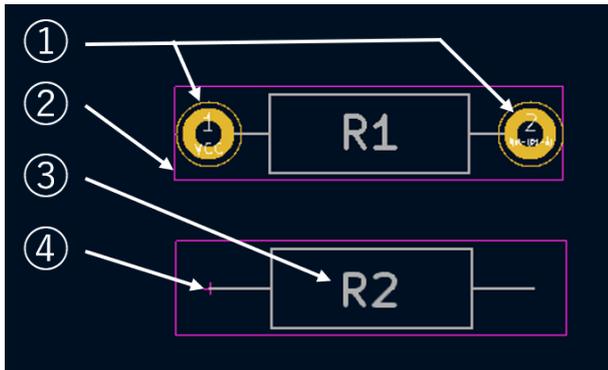


図 1: 素子と基本パラメータの対応例

番号	用語	役割
1	pad	素子の接続端子
2	boundingbox	素子のもつ領域
3	reference	素子の識別名
4	position	素子の座標の原点

### 4. 実装

提案システムは素子配置のデザインルールを記述した Python スクリプトを、KiCAD 内でユーザーが指定／実行することで動作する。図2に本システムの動作例を示す。ユーザーは KiCAD 内で回路図エディタにて回路図を作成する。次に、素子に対応するフットプリントの割り当てを行い、PCB (プリント基板) エディタにて回路図情報に基づいた部品データを出力する。フットプリントとは、PCB に電子部品を実装するために必要な設計情報であり、本稿では主に図1で示す内容に対応する。さらに、基板の外形線とグリッド原点を指定して、フットプリント群を基板の外形線の外側に配置することで準備が完了する。

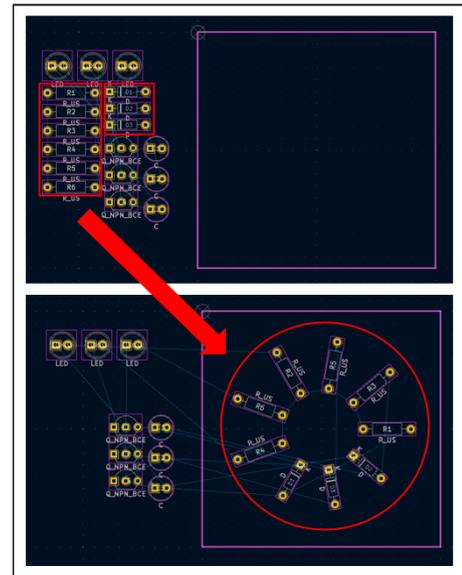


図 2: システムの動作例

#### 4.1 操作方法

現段階では、コンソールにコマンドを入力することでシステムの操作 (Listing1) を行うことができる。自動配置を行う素子の選択・デザインルールの選択・配置の完了・再配置などの機能を持っている。%で始まる行でユーザーに inputs を促しており、<--で始まる行で入力結果を示す。ユーザーは複数のデザインルールを組み合わせることで適切な部品配置を模索し、基板デザインを決定する。

Listing 1: コンソール上の操作の例

```

Unselected Elements
R1R2R3R4R5R6
Q1Q2Q3
LED1LED2LED3
D1D2D3
C1C2C3
% Select the elements to place automatically.
<-- R1,R2,R3,D1,C2,Q1,Q2
Unselected Elements
R4R5R6
Q3
LED1LED2LED3
D2D3
C1C3
Selected elements
R1R2R3
Q1Q2
D1
C2
% Complete element selection?(yes or no)
<-- yes
% Select a design rule from the list
    
```

```

1. circular arrangement-1
2. circular arrangement-2
3. random arrangement
...
<-- 3
% Complete automatic placement?(yes or no)
<-- yes
% Continue placing elements? (yes or no)
<-- no

```

## 4.2 基本的な配置処理

ここでは、基本的な配置処理として、部品の座標と重なり、部品間や外形線との距離について説明する。

**部品の座標：** 試行回数を減らすことを目的に 100mil(1mil=0.0254mm) のグリッド上に原則として配置する。

**部品の重なり：** 部品の重なりは boundingbox を検出し、hittest 関数を利用して判別している。

**部品間の距離：** フットプリント同士の最短距離は、pad 間の最短距離を制限することで設定する。標準では 100mil に設定している。

**外形線との距離：** 部品が基板の外側にはみ出ることを防ぐために、pad の座標が基板の外形線より 100mil 以上内側となるように制限している。

## 4.3 自動配置のデザイン例

本章では、回路図(図3)に示した回路に対して自動配置を行った結果を示して配置パターンについての説明を行う。この回路は3段の非安定マルチバイブレータの回路で、LEDが順番に点灯する。

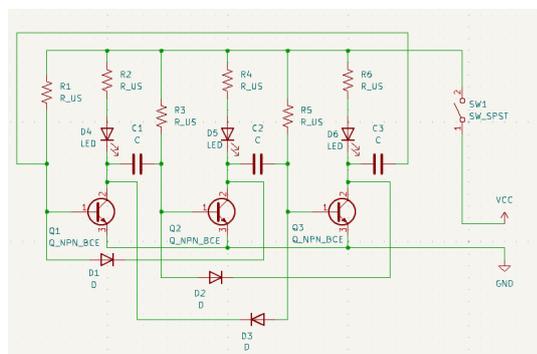


図 3: 対象とする回路図

### 4.3.1 図形状の配置

指定した図形上に部品を配置する例として、円形状の配置を実装した(図4)。この自動配置は配線の長さなどは配慮しておらず、ユニークな外観の配置パターンを提案する。円形状のような配置は、手動では角度の計算に手間がかかり、試行錯誤することはこれまでは難しかった。

図4の配置は、二種類の円形状配置アルゴリズムを使用して作成した。

抵抗・コンデンサ・ダイオード・LEDについては、選んだ素子の数で360度を分割して、円の半径・角度を調節することで配置している。内側のパッドまでの半径100mil、角度0度を初期設定とする。既に配置済みの部品と重なりがあった場合、全ての部品を15度ずつ回転させて配置可能な角度を探す。一周しても配置可能な角度が見つからない場合は半径を50milずつ増加させて再度回転を繰り返すことで配置可能な場所を探していく(図5)。

トランジスタについては、選んだ素子の一つずつ半径を調節しながら円形状に配置している。前項の配置は選んだ素子を円状にあらかじめ配置してから回転させるのに対して、素子の一つずつ配置することで横並びに素子を円形に沿って配置したり、細かい隙間に配置することができる。

(図6)

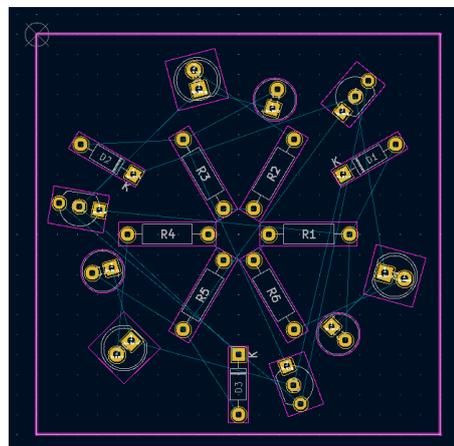
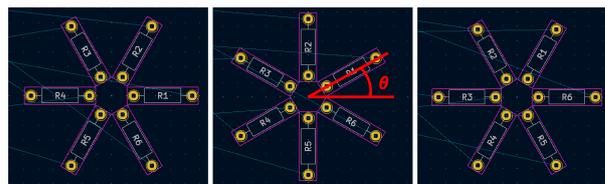


図 4: 円形状配置の例



回転角  $\theta = 0$

回転角  $\theta = 15$

回転角  $\theta = 30$

図 5: 円形状配置：手法 1

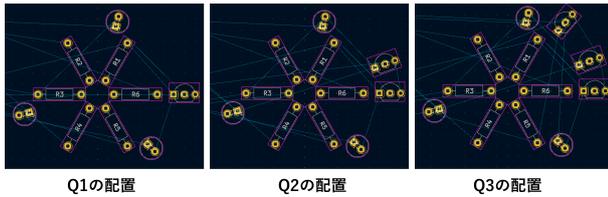


図 6: 円形状配置：手法 2

#### 4.3.2 ランダムな配置

あえて部品の位置と角度をランダムに配置する事例を実装した。一般的に部品配置の際には、回路図との関係性を考慮して角度や位置をある程度揃えて配置する。これは、基板の動作をわかりやすくするためである。一方、本手法では、部品をランダムに配置することで複雑な外観となり、部品同士の関係も分かりにくくなる。これにより、一見すると動作が予測できないような基板を作ったり、内部回路を隠蔽しやすくなる可能性がある。

図 7 の配置は、素子の座標を 100mil のグリッド上の点からランダムに抽出し、角度は 0 から 360 度までの数値をランダムに抽出し配置している。さらに素子間の最短距離を 100mil に設定することで、ランダムながらも pad の距離が近くなりすぎず、部品の取り付けに支障がない配置となっている。

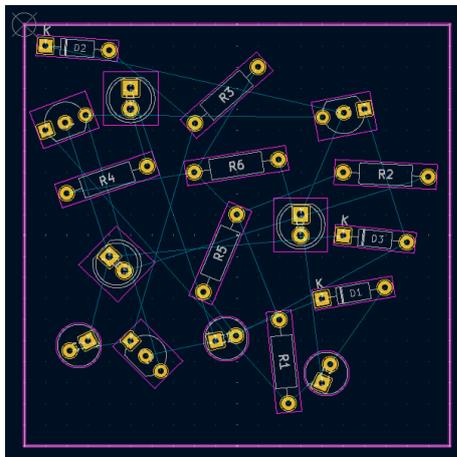


図 7: ランダム配置の例

#### 4.3.3 配線情報を考慮した部品選択

図 8 は配線のしやすさに着目し、回路図から取得した配線情報をもとにした配置である。ネット（お互いに配線される pad のリスト）の情報を利用することで、まず、左上の原点を基点として、接続されている pad が一直線になるように 100mil 間隔で y 軸上に部品を並べる（図 8 (1)）。次に、配置された部品の boundingbox を参照し、最も大きい x 座標から 200mil 右にずらした場所を次の基点として、同様の処理を繰り返す（図 8 (2)~(4)）。

ネットに接続されている pad に合わせて部品の位置と角度を調整することで接続されている pad 同士が近くなる

ように配置し、配線やはんだ付け作業が容易になるように配慮している。

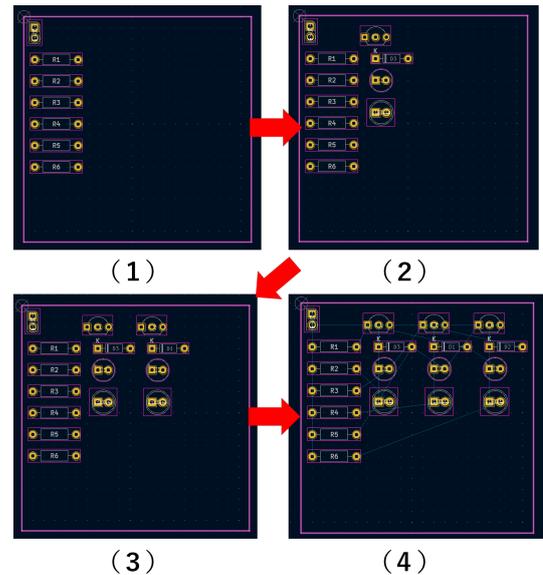


図 8: 配線情報を考慮した部品選択の例

#### 4.4 配置後の配線について

配置後の配線は手動でも行えるが、自動配線ツール「Freerouting」を利用して自動配線も行える。図 9 はランダム配置（図 7）に自動配線を行った例であり、トレース幅: 0.3mm, クリアランス: 0.5mm に設定した。

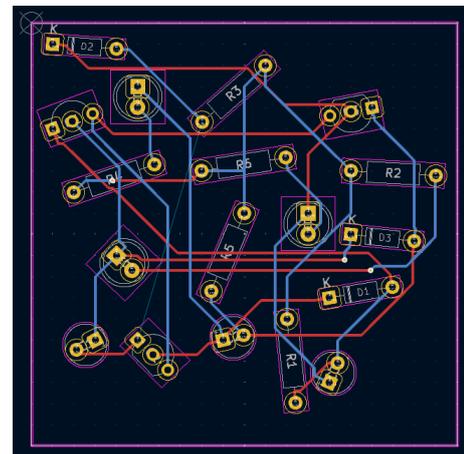


図 9: ランダム配置した部品に自動配線した例

### 5. まとめと今後の展望

本論文では、データに基づいて、多様なデザイン性を持つ部品配置案を提示することで、ユーザの試行錯誤を支援するシステムを構築した。今後はより多様な作業性・電気的特性やデザイン性に対応した配置アルゴリズムを開発していく。次に、CUI から GUI によるクリックやドラッグ

での操作に移行することで操作性の改善を行う。さらに、システムに従って設計した際の配線のしやすさ・外観の美しさ・基板設計の速度などの項目について、有効性の検証と課題の整理を行う。

## 参考文献

- [1] J. Garza and Steven Swanson. Pcb-ready breakout boards: Bridging the gap between electronics prototyping and production, 2023.
- [2] Leo McElroy, Quentin Bolsée, Nadya Peek, and Neil Gershenfeld. SVG-PCB: a web-based bidirectional electronics board editor. In *Proceedings of the 7th Annual ACM Symposium on Computational Fabrication, SCF '22*, New York, NY, USA, 2022. Association for Computing Machinery.
- [3] Kuan-Chun Chen, Chou-Chen Lee, Mark Po-Hung Lin, Yan-Jih Wang, and Yi-Ting Chen. Massive figure extraction and classification in electronic component datasheets for accelerating PCB design preparation. In *2021 ACM/IEEE 3rd Workshop on Machine Learning for CAD (MLCAD)*, pp. 1–6. IEEE, 2021.
- [4] Yoonji Kim, Hyein Lee, Ramkrishna Prasad, Seungwoo Je, Youngkyung Choi, Daniel Ashbrook, Ian Oakley, and Andrea Bianchi. SchemaBoard: Supporting Correct Assembly of Schematic Circuits using Dynamic In-Situ Visualization. In *Proceedings of the 33rd Annual ACM Symposium on User Interface Software and Technology, UIST '20*, p. 987–998, New York, NY, USA, 2020. Association for Computing Machinery.
- [5] Jie Qi and Leah Buechley. Sketching in circuits: designing and building electronics on paper. In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems, CHI '14*, p. 1713–1722, New York, NY, USA, 2014. Association for Computing Machinery.