

失敗から学びを得るSTEAM教育の実践事例

Learning from failure: Practice report on STEAM education.

西澤 利治
株式会社電脳商会

子どもたちが電子工作を行う機会が増えている。そこで本研究では、電子工作がうまく動作しない場合に不具合の状態を確認して原因を突き止め、適切な対策を考えて修正するという電子工作のプロセスにおける学びを支援するため、電子工作の「失敗」を整理して可視化する支援ツールを開発して、実際の電子工作ワークショップで実践したので報告する。

キーワード：STEAM 教育，教育方法，教材開発，問題解決力育成，失敗研究

1. はじめに

若年層を対象とするプログラミング教育では、教育用のハードウェアとしてRaspberry Piやmicro:bitが選ばれることが多い。これらのシステムは、一般のパソコンと比べると汎用入出力（GPIO）が充実しているため、外部に電子回路を接続してプログラムで制御することが容易である。そこで、学習者である子どもたちに、これらのハードを使用したシステムの組み立てや簡単な電子回路を組み合わせた上でプログラミングを体験させる、という取り組みが見られる。ところが、主たる学習目標をプログラミング教育に置いている場合は、電子工作は興味関心を引きつける添え物となっており、そこから学びを得るようにデザインされておらず、子どもたちがシステムの組み立てに失敗しないことを最優先して、作業の手順を細かく指示するといった運用がなされている。

確かに、電子工作の工程を細分化して作業の進行を管理すれば再現性は高まるであろう。しかし、その反面、学習者が得られる学びは少ない。実際の子どもの向け電子工作ワークショップで、例えば「LEDの長い脚はブレッドボードのB2の穴に挿し込む」といった指導を行うものがあった。このように指示すれば、確かに作業上の間違いは回避されるが、参加した子どもに聞いてみると「LEDはB2に入れる」という返事が戻ってきたことがあった。

作業を細分化し、指示を具体化すれば電子工作の失敗は防げるかもしれないが、学習者の理解は外形的なものにとどまり、電子工作の本質を把握し損なう可能性が高い。仕事としての電子工作であれば作業効率や品質を優先するのは当然だが、教育とし

ての電子工作は、いかにそこから学びを得られるようにデザインするか、という配慮が不可欠であろう。

そうした問題意識の元に、電子工作の初心者を対象として、電子工作での失敗を可視化し、失敗をデバッグして修正するプロセスで学びを得ることを目的とした支援ツールを開発し、実際に電子工作ワークショップで使用して評価を行ったので報告する。

2. 本研究の目的と背景

学習者が、メンターの指示通りに作業を行った電子工作が正常に動作したとしても、それが学習者の理解の上で実現されたのか、偶然そうだったのかは判別できない。STEAM教育の視点から電子工作を考えれば、学習者は工作するシステムの目的を理解してどのように実現するかを考え、目的どおり動作しないなら不具合の状態を確認して原因を突き止め、適切な対策を考えて修正する、というプロセスを体験することが重要であると考え。そのためには、学習者を上手に失敗させ、失敗を成功につながる学びのチャンスとしてデザインする学習環境が重要になってくる。とはいえ、電子工作の経験が少ない学習者にとっては、苦勞して制作した電子工作が動かなければ、自信を失ってやる気をなくし、途中で投げ出してしまうことも考えられる。電子工作の経験が少ないほど、原因や正体のわからない失敗は不安材料でしかない。だが、これを逆に言えば、学習者が不具合のある「失敗」状態が想定範囲内であり原因が推定可能であると理解すれば、学習者は不安を覚えず学びにつながるだろう。こうした仮設の下に、電子工作の失敗を可視化するツールを設計開発した。

3. 研究の方法

3.1. 「失敗」を整理し可視化するツール

本研究では、個人が電子工作を行うレベルの作業を想定し、電子工作における失敗を可視化するツールとして以下のものを設計開発した。

- ア 「失敗」を整理する特性要因図
 - イ 「失敗」を予防する事前チェックシート
 - ウ 「失敗」を修正する原因究明チェックシート
- シーモア・パパートは、「個人的に興味あるものを作っているときに、特に効果的に新しい知識を構築する」との学習理論である構築主義を提唱したが、その際にネガティブな感情が生じるなら、知識の構築はうまくいかないであろう。これらのツールは「失敗」は対応可能な状態であることを可視化し、学習者のネガティブな感情の解消を目的にしている。

3.2. 「失敗」を整理する特性要因図

まず、電子工作における失敗とは何かを把握するために、インターネットから電子工作における失敗事例を収集して不具合の状態を抽出し、整理のためのおおまかな軸を考案した。その情報を元に、個人で電子工作を行ういわゆる「メイカー」と称される層にヒアリングを行い、実際に自身で体験した電子工作の不具合や失敗の事例を提供いただいた。次に、収集した失敗のパターンをQC 7つ道具と呼ばれる特性要因図を応用して、系統的に整理した(図1)。

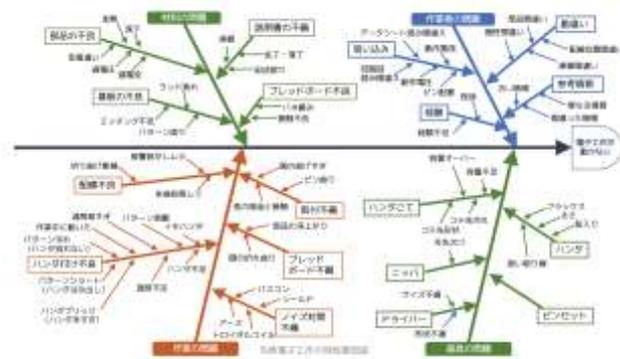


図1 電子工作「失敗」の特性要因図

電子工作「失敗」の特性要因図(以下、特性要因図と表記)では、失敗の状態に4つの軸を設けて失敗のパターンを整理している。「作業者の問題」「道具の問題」「材料の問題」「作業の問題」である。

特性要因図の使用イメージは以下のとおりである。学習者は、電子工作の状態を観察して特性要因図から該当する失敗の原因となるパターンを見つけ出し、メンターはその不具合の枝の有効な対策を指導することを想定している。

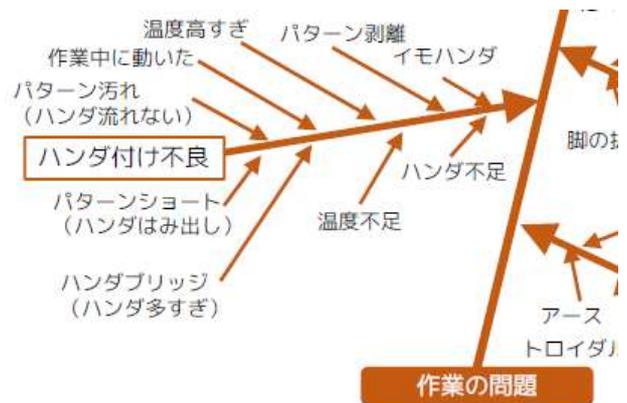


図2 特性要因図「ハンダ付け不良」の枝

図2を用いて具体的なプロセスを説明する。ハンダ付けを例にとると、学習者(電子工作の作業員)は自分の行ったハンダ付けの状態を観察して、どのような不具合に該当するのかを考える(例:ハンダが多すぎる)。学習者の判断について、メンターはこの不具合は「ハンダ付け不良」に分類されることを説明し、ハンダ付け不良がどのような症状を発生させるのかと、これを解消するための対策を指導する。

3.3. 「失敗」を予防する事前チェックシート

この失敗の特性要因図を元にして、失敗を自己評価するチェックシートを作成した。なお、本研究では、ハンダ付けで配線を行う電子工作を対象とした。ハンダ付けによる配線は作業後に配線を修正することが難しいため、作業前の確認が重要である。そこで、失敗の特性要因図からハンダ付けに関わる項目を抽出して、ハンダ付けを行う前に事前に自己評価して失敗を予防するチェックシートを作成した。

事前チェックシートの使用イメージは以下のとおりである。学習者は、電子工作の作業に入る前に項目ごとの対応状態を確認して自己評価して、失敗しそうな「自信のない」作業項目を見つけ出して、必要であれば対応することを想定している(図4)。

4. 実践と評価

これらのツールを実際のワークショップで使用し評価したので報告する。

4.1. 動かない電子工作キットを動かす

作業員である中学2年の男子から、市販の電子工作キットを配線したが動作しないので、動くようにアドバイスしてほしいとの依頼を受けた。このキットは、DCモーターの速度をICとトランジスタで組んだ発振回路で制御するものであるが、後述するようにオリジナルの基板の一部変更が加えている。

そこで作業員に対し、特性要因図を用いて失敗を系統化する考え方を説明し、その上で自分の行った作業を振り返る形で原因究明チェックシートを記入させた。次に作業員とメンターが協力して基板の表面を確認し、特性要因図に従って部品の取り付け状態や極性ある部品の向き、他の部品との接触などを評価した。ハンダ付けの状態をチェックしたところ、ハンダの量、表面の状態、基板の焦げから、作業員に使用したハンダごてが電子工作用でないと推測されることを指摘したところ、作業員は鉄道マニアで、鉄道に使用された放出機材の配線をするため、それ用の太いハンダごてで作業したことが確認できた。

評価を続けるうちに、基板上に未配線のホールが発見されたため、回路図と照合したところ、そのホールに必要な配線がなされていないことが判明した。キット付属のマニュアルでは、その部分は文章で「端材を加工して接続する」と記述されていたが、作業員はマニュアルに掲載された実体配線図を参考に作業を進め気付かなかったためである。そこで線材を追加したところ、正常動作するようになった。

作業員は、このキットで鉄道模型の速度制御を考えており、キットに用意された基板用半固定ボリュームを使用せず外部に配線を引き出して別に購入した小型ボリュームを接続していたが、その変更が不具合の原因ではないかと不安視していた。だが、その工夫は問題なく動作することが確認できたため、満足していた。さらには、特性要因図を確認しているうちに、作業員自身がボリュームに接続する捻り線の処理が甘くヒゲが出ていることに気づき、自らこれを処理して固定するよう工夫するようになった。

ワークショップ開始時の作業員は、どこから手をつけてよいかわからず途方にくれた状態だったが、特性要因図に沿って基板の状態を確認し「この作

ハンダ付けで失敗しないよう始める前に準備しよう

次の項目を確認してみよう。

- 作業員の質問

とりつける部品は間違いないか?	(心配かも)	だいじょうぶ
部品の向きは間違いないか?	(心配かも)	だいじょうぶ
作業の順序は間違っていないか?	(心配かも)	だいじょうぶ
説明書を読んで理解しているか?	(心配かも)	だいじょうぶ
- 道具の質問

ハンダごての容量は適正か?	(大きい小さいかも)	だいじょうぶ
コテ先の形は適正か?	(心配かも)	だいじょうぶ
コテ先はきれいにしていないか?	(心配かも)	だいじょうぶ
ハンダの種類は適正か?	(心配かも)	だいじょうぶ
フラックスはあるか?/ヤニ入りか?	(心配かも)	だいじょうぶ
- 作業の質問

パターンを汚れを落としたりしたか?	(落とさなかったかも)	だいじょうぶ
部品の脚を曲げたか?	(曲がなかったかも)	だいじょうぶ
部品の脚を曲げすぎていないか?	(心配かも)	だいじょうぶ
部品の脚が接触していないか?	(心配かも)	だいじょうぶ
失敗したときの対応が分かっているか?	(心配かも)	だいじょうぶ
- 材料の質問

基板のパターンが割れていないか?	(心配かも)	だいじょうぶ
電源容量は不足していないか?	(心配かも)	だいじょうぶ

図4 「失敗」を予防する事前チェックシート

3.4. 「失敗」を修正する原因究明チェックシート

また、作成した電子工作が動かない場合、自分が行った作業を評価して、修正すべき問題点を推測することを目的とした原因究明チェックシートも作成した(図5)。

動かない電子工作が「なぜ動かないのか」考えてみよう

次の項目を確認してみよう。心当たりはあるかな?

- 配線の準備はないか?

順番に繋がらぬか?	(抜けているかも)	だいじょうぶ
折り曲げ配線はないか?	(曲げすぎたかも)	だいじょうぶ
末端熱型ムラ(ヒゲ)はないか?	(ヒゲがあるかも)	だいじょうぶ
- 部品の取り付けに準備はないか?

他の部品と接触していないか?	(接触しているかも)	だいじょうぶ
部品の脚をギリギリで曲げすぎているか?	(曲げすぎたかも)	だいじょうぶ
部品の向きが間違っていないか?	(間違ったかも)	だいじょうぶ
部品の種類を間違っていないか?	(間違ったかも)	だいじょうぶ
電源の向きは間違っていないか?	(間違ったかも)	だいじょうぶ
電源の電圧は大丈夫か?	(古い電圧かも)	だいじょうぶ
- ハンダ付けに準備はないか?

パターンを汚れを落としたりしたか?	(落とさなかったかも)	だいじょうぶ
部品の脚を曲げたか?	(曲がなかったかも)	だいじょうぶ
ハンダがはうまく流れて銀色に光る? (シフになっているか)	(シフになっているかも)	だいじょうぶ
ハンダがはみだして部品がショートしていないか? (あるかも)	(あるかも)	だいじょうぶ
ハンダで隣のパターンと繋がっていないか? (あるかも)	(あるかも)	だいじょうぶ
加熱しすぎてパターンが割れているか? (割れているかも)	(割れているかも)	だいじょうぶ
ハンダ付けした部品がガラガラしていないか? (動くかも)	(動くかも)	だいじょうぶ
線とランドはハンダでつながっているか? (つながっていないかも)	(つながっていないかも)	だいじょうぶ

図5 「失敗」を修正する原因究明チェックシート

業は正しい」ことが評価されるに従って自信に繋がった。また、正常動作しない場合もいきあたりばったりで対策を行うのではなく、まずは正常な部分を切り分けて失敗をあぶり出すという系統的なアプローチによって不具合を見つけ出していく方法を体験的に理解してもらうツールとして有効といえよう。

4.2. 電子工作キットを組み立てる

この作業者は、自閉傾向のある中学校1年生の男子(と保護者)である。作業者は、報告者が主催するmicro:bitのプログラミングワークショップに参加し、自分でプログラムを組んで動かすことの成功体験からプログラミングに興味を持ち、自分が自由に使えるパソコンを入手したいと言う意識が芽生えた。そこでシングルボードパソコンの「IchigoDake」のキットを作成して、そこで動作するゲームプログラミングを行うワークショップに参加した。

まず特性要因図を使ってメンターから作業者に対し、工作を始める前に失敗しそうな作業を説明し、特性要因図で作業を確認・評価しながら進めることを了承してもらった。また、作業者にとってハンダ付けは初体験のため、ハンダ付けでは高温になるので火傷に注意すること、煙を吸い込まないことなど、作業上の危険性と注意点を説明した。

メンターは部品の取り付け方法を説明するが、実際の部品の取り付けやハンダ付けなどの作業はすべて学習者が行った。おおよそ40分程度で組み立ては終了したが、工程ごとに作業者とメンターで特性要因図で作業の状態を確認したため、動作テストを実施したところ1回で正常に動作させられた。

引き続きIchigoJam BASICを使用したゲームプログラミングを行った。作業者は、MakeCodeのJavaScript Blocks Editorの使用経験しかなかったが、JavaScript Blocks Editorに用意された「プログラムをJavaScriptに変換する」機能により、ブロックで組んだプログラムをJavaScriptで見せていたので、言語を使用するプログラミングに拒否感はなく、むしろ興味を感じていたのである。今回教材として使用したIchigoJamのゲームプログラムは「すべてのこどもたちにプログラミングの機会を提供する」を理念するPCNが提供する、BASIC言語で書かれた縦スクロールで迫る障害物を操作して避けるというものである。指導方法は、メンターが新しいBASICコマンドの機能を説明して、学習者はゲームプログラムのリストを参考にして該当部分を入力し、動作を確認

するという流れで行われた。これにより、作業者は自身の理解度に合わせたペースで入力を進めることができると同時に、またサンプルプログラムを一度に入力するのではなく、個々のコマンドの機能を理解しながら段階的に拡張するよう構成されており、プログラミング経験の少ない作業者でも成功体験を積み重ねることができるようになっている。また、作業者がキャラクターのデザインやスピードをカスタマイズでき、達成感を得られるようになっている。

作業者も、自分で組み立てたパソコンで自分が入力したプログラムが動く体験に興奮し、わざと難易度の高い状態に設定して操作するなど、メンターに頼ること無く自身で時間いっぱい工夫を続けていた。

このように、電子工作でも段階的にゴールを設定することで、学習者は自己達成感を得やすくなる。また、事前チェックシートにより想定される失敗が予防でき、失敗しても特性要因図が想定した既知の不具合であると理解することにより、作業者は安心して電子工作に取り組むことができたといえよう。

5. まとめ

雑誌記事やマニュアルは「成功」状態のシステムが掲載提示されているが、完成に至る手順の説明が不十分だったり、失敗の例示が少なく、不具合を解消する方法が説明不足なことも多く、それが作業者が失敗のままに終わらせることに繋がっている。

STEAM教育では、想定通り機能しない「失敗」したシステムをデバッグして不具合を解消することを学ぶことこそ重要ではないだろうか。学習者が「失敗」の原因を特定し有効な対策を実施して失敗を解消する、成功体験を積み上げて問題を解決する、あきらめない学び得ることを支援するのが必要だと思う。

今回開発した失敗の特性要因図とチェックシートは、電子工作の失敗を系統的に整理して可視化し評価することを目的としたが、これによって学習者は、自身の失敗がどこに位置づけられるのか俯瞰して客観的に眺めることができるようになり、自らの学びを実現していく手助けとなることを期待している。

6. 参考文献

Mitchel Resnick (2001) 非集中システム, コロナ社, 東京