

実物のディスプレイに近い加法混色モデル教材の一提案

- マイコンボードによる加法混色の実験 -

天良 和男

東京都立小石川中等教育学校

tenra.kazuo@nifty.com

加法混色の仕組みを理解させる教材として、王・小山によって開発された装置^①などがある。この装置は3原色のLEDの発光強度をPC制御で自由に変えてさまざまな色を作ることができるが、電子回路を自作する必要がある。筆者は、フルカラーLEDが搭載されたマイコンボードとLEDを制御する命令ブロックを用いることによって、電子回路の製作を必要とせず王・小山によって開発された装置と同等の機能を有する教材を製作した。この装置は、実物のコンピュータディスプレイに近い加法混色モデル教材として活用することができる。

1. はじめに

使用したマイコンボードは、図1の(株)アーテックのStuduino:bitである。表面には個別に点灯できる25個のフルカラーLEDがマトリクス上に配置されており、これらは英数字、記号などを表示するディスプレイの役割をする。

フルカラーLEDは、赤、緑、青の3つのLEDが一つのパッケージに封印されたもので、3つのLEDの発光強度を変えることで加法混色により様々な色を表示することができる。

Studuino:bitを動かすソフトウェアには、Scratch3.0互換のStuduino:bit SoftwareやPython3互換のMicroPython、JavaScript互換のQuickJSなどがある。



図1 Studuino:bit

2. すべてのLEDを同じ色に発色させる場合

Studuino:bit上の25個のLEDを制御するためのStuduino:bit Softwareの命令ブロックは複数あり、そのうちの1つに図2のブロックAがある。

ブロックAは、「LEDを全て赤:Ⓡ 緑:Ⓞ 青:Ⓟで点灯」の形式で設定することで、全てのLEDを同じ色に発色させることができる。

Ⓡ, Ⓞ, Ⓟの値はそれぞれ赤、緑、青の発光強度を表し、0~100の範囲の整数で設定する。

GUI部品である図3のスライダーと連動させ

ることで、つまみをマウスで左右にドラッグしてⓇ, Ⓞ, Ⓟの値を変化させることができる。

ブロックAを使ってⓇ, Ⓞ, Ⓟの各値をすべて同じ値(すべて100)にしたとき、図4のようにすべてのLEDは白色に発色する。

この原稿がモノクロのため、図4、図7、図8には、図中に色名を付記している。

ブロックAの欠点として、フルカラーLEDは、赤、緑、青の3つのLEDが一つのパッケージに封印されているため、混色された後の光しか観察することができず、混色される前の各LEDから発光される赤、緑、青の光を直接観察することができない。加法混色の仕組みを理解させるには、混色前後の様子を観察することが必要である。



図2 ブロックA



図3 スライダー(図2や図5と連動させる)

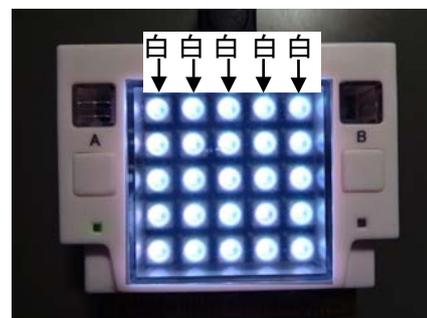


図4 ブロックAを使って発色させる場合(すべて白色に発色する)

3. LED を3つごとに同じ色に発色させる場合

LED を制御するためのもう一つのブロックとして、図5のブロック B がある。

ブロック B は「LED の X:ⓧ Y:Ⓨを赤:Ⓡ 緑:Ⓞ 青:Ⓟで点灯」の形式で設定することで、設定した位置の LED を設定した色に発色させることができる。ⓧ, Ⓨはそれぞれ LED の横位置, 縦位置を表す番号で、図6のように、それぞれ1～5の整数で設定する。また、Ⓡ, Ⓞ, Ⓟの値はそれぞれ赤, 緑, 青の発光強度を表し、0～100の範囲の整数で設定する。

図6は、Studuino:bit 上の25個のLEDをブロック B を使って発色させるパターンも表している。

図中の R, G, B と記した部分の LED は、それぞれ赤, 緑, 青に発色させることを示しており、「RGBRGB…」のように3つごとに同じ色に発色させる。

例えば、ⓧ=1, Ⓨ=1の部分の R と記した LED には、Ⓡのみを設定し（例えばⓇ=100）、Ⓞ, Ⓟを0にして赤色に発光させる。また、ⓧ=2, Ⓨ=1の部分の G と記した LED には、Ⓞのみを設定し（例えばⓄ=100）、Ⓡ, Ⓟを0にして緑色に発光させる。さらに、ⓧ=3, Ⓨ=1の部分の B と記した LED には、Ⓟのみを設定し（例えばⓅ=100）、Ⓡ, Ⓞを0にして青色に発光させる。それ以外の R, G, B と記した部分の LED も同様に発色させる。このように設定すると、図7のようになる。

実際の LED ディスプレイも隣り合う赤, 緑, 青の三色の副画素がまとまって一つの画素として機能しているため、ブロック A を使って発色させる場合（図4）よりもブロック B を使って発色させる場合（図7）のほうが、実物のコンピュータディスプレイに近いモデルを示しているといえる。

しかし、図7のように発色させても近くから見ると混色された色（この場合は白色）に見えない。これは、網膜上で赤, 緑, 青のエリアの結像位置が明らかに異なるため、別々の色として認識されるからである。遠くから見ると結像位置が重なり混色された色に見えるが、近くから見ても混色された色に見えるには、図8の右半分のようにトレーシングペーパーなどで LED の表面を覆うなどすればよい。これにより、混色前後の様子を観察できる。光はトレーシングペーパーを透過する際、その表面や内部にある微小な凹凸、繊維などが光を乱反射・乱透過させて広範囲に拡散するため、混色させることができる（この場合白色になる）。



図5 ブロック B

ⓧ	1	2	3	4	5	
Ⓨ	1	R	G	B	R	G
	2	G	B	R	G	B
	3	B	R	G	B	R
	4	R	G	B	R	G
	5	G	B	R	G	B

図6 LED の位置番号と発色パターン



図7 ブロック B を使って発色させる場合
（図6のパターンで発色させる）



図8 右半分をトレーシングペーパーで覆う場合
（左半分は図7と同様、右半分は白色に発色）

4. おわりに

Studuino:bit Software の代わりに Python と MicroPython を組み合わせることにより、上記と同等の機能を有する教材も作成することができた。

Studuino:bit[®]には、フルカラーLEDのほか、温度, 光, 加速度, 磁気などのセンサ, WiFi, 省電力 Bluetooth (BLE) などの通信機能が搭載されており、情報 I・II の様々な学習項目に対応できるので、各学校で活用されることを期待したい。

参考文献

- (1) 王 宇熙, 小山 英樹: 3 原色 LED を用いた加法混色学習システムの開発と授業実践, 日本産業技術教育学会近畿支部 第 33 回研究発表会 A-01, 講演論文集 pp.1-2(2016).
- (2) 三宅 丈夫: マイコンボードで 100 倍わかる高等学校情報 I・II (2022 発行), https://www.artec-kk.co.jp/special/school_programming/04case.php