

# Blockly × Electronで作るプログラミング環境の概要

竹本 義孝



図1 第3部でやること…マイコン制御用C++コードを生成するブロック・プログラミング環境を作る

第3部 第4章で作成したアプリケーションを実行している様子。ブロックからマイコン制御用のC++コードを直接生成するので、デバイスを実作するような低レイヤな処理は格段に作りやすい

第1部と第2部で紹介した内容は、いずれもScratchをベースとしたブロック・プログラミング環境でした。マイコン自体のプログラムを直接作る訳ではないので、デバイスを実作するような低レイヤな処理の作り込みには工夫が必要でした。

第3部では、ブロックからC++コードを直接生成するアプリケーションを作成します(図1)。デバイスを操作するような処理は格段に作りやすくなります。また、インターネットから切り離れたアプリケーションを作成することも可能なので、工場内のような閉じた環境での使用も可能です(図2)。第4章では実際にSTM32マイコン用のC++プログラムを生成します。第3部で紹介するサンプル・コード

は、次のウェブ・ページから入手できます。内容ごとにディレクトリが分かれています。対応するディレクトリ名は本文に記載しています。

<https://www.cqpub.co.jp/interface/download/contents2022.htm>

(編集部)

## ● 組み込みにもブロック・プログラミングを!

Scratchのようなビジュアル・プログラミングは、今や子ども向けプログラミング環境の枠を超えて、MakeCode(マイクロソフト)やUIFlow(M5Stack)など、組み込みの世界にも波及してきました。

筆者もCoderDojoで子どもたちと一緒にScratchに触ってきました。1時間もあれば何となく子どもたちも触れるようになるビジュアル・プログラミングのしきいの低さに驚くばかりです。

もしScratchのような環境でマイコンや開発した製品を操作できるようになれば、学習のしきいは大きく下がるはずです。今まで開発担当者でなければ検証できなかったことが、個々人で簡単に操作できるようになるかもしれません。

## ● 第3部でやること…「Blocky」をベースにマイコン・プログラミング環境を作る

### ▶ ScratchやMakeCodeのベースにも使われている

Scratch, MakeCode, UIFlowなどの有名なビジュアル・プログラミングの環境では、実はバックグラウンドにブロックを描画するライブラリとして、ゲートルが開発した「Blocky」が使われています<sup>注1</sup>(図3)。

### ▶ ブロックから直接コードを生成できる

ScratchとBlockyのブロックには大きな違いがあります。Scratchのブロックは既に完成された機能が呼び出されるのに対し、BlockyのブロックはC++コードを生成します。生成されたC++コードを評価したり、コンパイルしたりすることで初めて機能として動作します。

注1: 正確には、Scratchの場合はBlockyからフォークしたscratch-blocks、MakeCodeの場合はBlockyからフォークしたPXT blockyを使っています。