

「プログラムによる計測と制御」を対象とした 「疑似体験型」プログラミング学習支援システム

Programming Learning Support System for the Education on “Computer-based Measurement and Control” through Simulated Experience

原 康仁, 稲垣 宏

Yasuhito HARA, Hiroshi INAGAKI

国立高等専門学校機構 豊田工業高等専門学校

National Institute of Technology, Toyota College

Email: inagaki@toyota-ct.ac.jp

あらまし：中学校の「技術・家庭科」の「情報に関する技術」分野において、「プログラムによる計測と制御」が必修となっている。教育現場では、教材用のロボットを用いた意欲的な授業が実施されている例がある一方で、学校側の予算不足や、教員のスキル不足により、効果的な授業を実現できていないことも多い。本研究では、「プログラムによる計測と制御」を対象に、十分な予算や、豊富な ICT 教育経験がなくとも効果的な授業を可能にする「疑似体験型」プログラミング学習支援システムを開発した。今後、効果の検証を行い、機能の追加や拡張を行うことで、より効果的な学習支援システムを目指す。

キーワード：技術・家庭科，プログラミング教育，ハードウェア制御，プログラミング学習支援システム

1. はじめに

中学校の「技術・家庭科」の「情報に関する技術」分野において、平成 24 年度より、「プログラムによる計測と制御」が必修化された。学習指導要領には、「実践的・体験的な学習活動を通して、プログラムにより機器等を制御したりする喜びを体験させる」、「情報処理の手順を考える際に、自分の考えを整理するとともに、よりよいアイデアが生み出せるよう、フローチャートなどを適切に用いることについて指導する」とある⁽¹⁾。

それに伴い、教材用ロボットを用いた意欲的な授業が実施されている一方で、効果的な授業が実現できていないことも多い。ロボットなどの実機を用いた「計測と制御」体験には、ハードウェア制御特有の難しさがあり、教員にとってハードルが高い。また、学校側の予算不足という問題もある。

そこで、豊富な ICT 教育経験や十分な予算がなくとも効果的な授業を可能にするプログラミング学習用の教材が必要となる。

本研究では、「プログラムによる計測と制御」を対象とした「疑似体験型」プログラミング学習支援システムの開発を目指す。

なお、本システムは、開発言語として C#を用い、Windows フォームアプリケーションとして開発した。

2. 開発するプログラミング学習支援システムの概要

本システムでは、ハードウェア制御の疑似体験によって、プログラムの制御構造の基本である「順次処理」、「条件分岐」、「繰り返し」の直観的な理解を促すことを目指す。

2.1 学習スタイル

「与えられたミッションを段階的にクリアしながら次のステップに進む」という学習スタイルを採用する。学習者は仮想的な色センサやタッチセンサを用いて、ミッションクリアを目指す。ミッションには、色センサを用いる「ライトレースをせよ」や、タッチセンサを用いる「壁に 10 回ぶつかれ」などがある。ミッション表示画面を図 1 に示す。

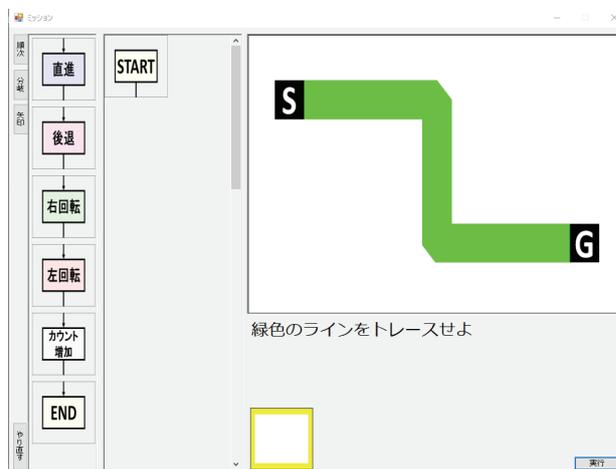


図 1 ミッション表示画面

2.2 プログラミング形式

本システムでは、ブロックプログラミング形式を採用する。ここでは、与えられたブロックを使って、フローチャートを作成する。実行時には、実行画面が開かれ、実行結果をアニメーションで確認することができる(図 2)。

本システムで、実装したブロックの機能を以下に示す。

- (a) 直進：制御対象を指定ピクセルだけ前進させる。
- (b) 後退：制御対象を指定ピクセルだけ後退させる。
- (c) 右回転：制御対象を指定角度だけ時計回りに回転させる。
- (d) 左回転：制御対象を指定角度だけ反時計回りに回転させる。
- (e) カウント増加：内部のカウント変数の値を1増やす。
- (f) カウント値監視：カウント変数の値が指定した数と一致するかを条件として分岐する。
- (g) 色検知：取得した色が指定した色と一致するかを条件として分岐する。
- (h) 接触検知：取得した色が黒色と一致するかを条件として分岐する。
- (i) 分岐終点：分岐の終点を示す。
- (j) 繰り返し始点：繰り返しの始点を示す。
- (k) 繰り返し終点：繰り返しの終点を示す。
- (l) END：プログラムの終わりを示す。

実装済みのブロックを以下に示す(図3)。

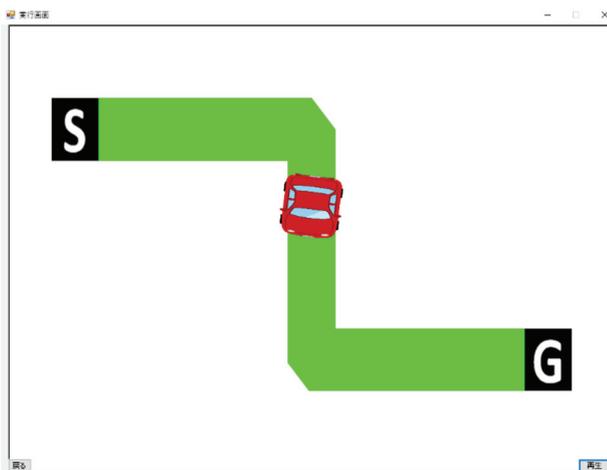


図2 実行画面

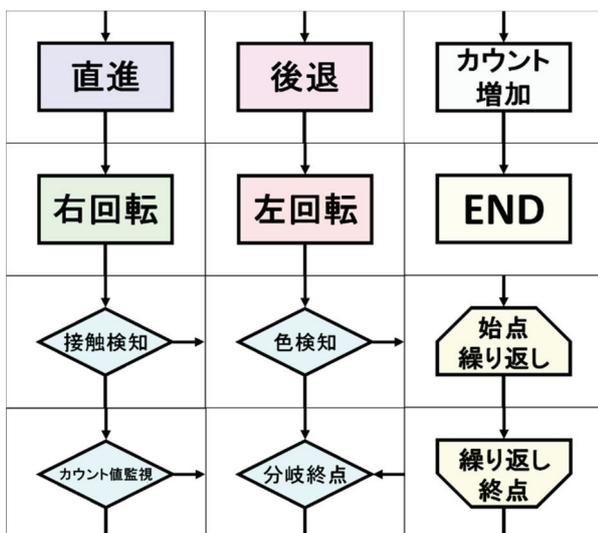


図3 ブロック一覧

2.3 制御する対象の設定

学習者が、画像データを変更することで、制御する対象を自由に変更が可能である。また、制御する対象のどこにどのセンサがあるのかを指定することができる(図4)。センサ座標は、設定画面で制御する対象のセンサ座標として扱いたい位置をクリックすることで設定する。



図4 センサ設定の例

3. システムの使用法

3.1 学習者側の操作

与えられたミッションのクリアを目指し、フローチャートを作成する。フローチャートは、ブロックを組み合わせることで作成する。使うブロックをクリックにより選択した後、配置したい場所をクリックすることでブロックが配置される。フローチャートが完成したら、実行ボタンを押し、実行結果を確認する。

3.2 教員側の操作

制御する対象の設定を行う。まず、制御する対象の画像を、制御したいものの画像へと変更する。その後、センサの種類を選択と、座標の設定を行う。

4. おわりに

「与えられたミッションを段階的にクリアしながら次のステップに進む」学習スタイルであること、マウス操作のみでフローチャートが作成できること、ハードウェア制御特有の難しさを考慮しなくてもよいこと、といった本システムの有する特長により、豊富なICT教育経験を持っている教員でなくても、「プログラムによる計測と制御」の授業を効率よく進めることができると期待している。

今後、実際のプログラミング学習用ロボットカーに合わせて、本システム内で制御する対象の画像やセンサを設定し、実機を用いたプログラミング実習へのスムーズな移行が可能であることを実証したい。そして、試行実験を繰り返すことで、使い勝手の向上や新しい機能の実装を進めていきたい。

参考文献

(1) 文部科学省：“中学校学習指導要領 技術・家庭科編”，pp.32-37, (2008)