

Raspberry Piを用いたプログラミング教育用教材の開発

福井 真二* 藤澤 悠貴**

* 情報教育講座

** 卒業生

Development of Teaching Material for Programming Education Using Raspberry Pi

Shinji FUKUI* and Yuki FUJISAWA**

**Department of Information Sciences, Aichi University of Education, Kariya 448-8542, Japan*

***Graduate, Aichi University of Education*

1. はじめに

近年、小学校から高等学校までの初等中等教育において情報教育が充実されてきている。今後の情報教育では、プログラミング教育が導入・拡充されることが予定されている。平成29年3月告示の学習指導要領では「情報活用能力」を「学習の基盤となる資質・能力」と位置づけ、教科横断的に育成する旨を明記するとともに、小・中・高等学校を通じてプログラミング教育を充実させることが示されている[1]。2020年度には小学校でのプログラミング教育が必修化され、2022年度には高等学校の教科「情報」の新たなカリキュラム「情報I」においてプログラミングが必修化される。このように、プログラミング教育は非常に重要視されてきている。

本研究では、ロボットカーを用いた初等教育に適したプログラミング教材を開発する。ロボットを用いたプログラミング教材はいくつも市販されているが、本研究では、安価に作成でき、かつ、複数人が無理なく共有することのできるロボットカーを開発する。本教材では、プログラミング環境としてScratch、ロボットカーを制御するシングルボードコンピュータとしてRaspberry Piを利用する。Scratchを利用することにより、まだキーボード入力及びプログラミングに不慣れな児童であっても、コードをキーボードから入力する必要がなく、用意されているブロックを自由に組み合わせるだけでプログラムを簡単に作成できる。Raspberry Piを利用することで安価にロボットカーを作成でき、かつ、Scratchから無線LAN経由で命令を送信することで無理なく複数人でロボットカーを共有することができる。また、ロボットカーを児童生徒が自作することで、その製作過程を通して児童生徒が

楽しみながら電子回路に触れることもできる。

児童生徒自身が自分で作成したプログラムにより実際にものを動かしたという体験をすることにより、プログラミング学習に対する興味・関心などを育む教材を開発することが本研究の目的である。

2. Scratch

本研究では、プログラミング環境としてScratch [2]を使用する。Scratchはマサチューセッツ工科大学で開発されたビジュアルプログラミング環境である。Scratchのプログラミング環境を図1に示す。図に示されているように、Scratchではブロックを組み合わせることでプログラムを作成する。そのため、コードをキーボードで打ち込む必要がない。小学生はまだキーボード入力が不慣れなため、Scratchは小学校においてプログラミング教育をする上でとても扱いやすいプログラミング学習環境としてよく利用されている。小学校におけるプログラミング教育のためのポー

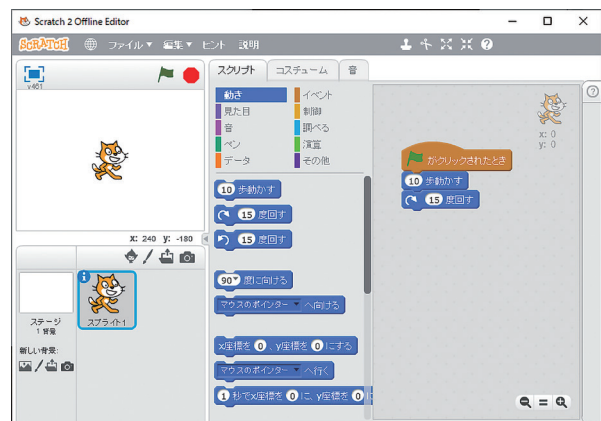


図1 Scratchによるプログラミング

タルサイト [3] においても Scratch を利用した授業実践例が多く紹介されている。

本稿執筆時において Scratch の最新バージョンは 3 である。本研究では、日本語のドキュメントが豊富で機能拡張のしやすいバージョン 2 オフライン版（以下 Scratch 2 という）を利用する。Scratch 2 では、Scratch 2.0 Extension を作成することでブロックの追加と外部デバイスの制御が可能となる [4]。Extension ファイル内に追加するブロックに関して記述する。追加されたブロックが実行された際、helper app と呼ばれる外部プログラムと HTTP による通信を行うことができる。helper app を作成することにより、外部デバイスを制御することが可能になる。helper app は HTTP サーバーとして動作し、Scratch 2 から GET メソッドにより送信されたデータを基に外部デバイスを制御する。

3. Raspberry Pi

Raspberry Pi は イ ギ リ ス の Raspberry Pi Foundation により開発されているシングルボードコンピュータである。教育での利用を想定して作成されており、非常に安価に入手することができる。Linux ベースの OS を動作させることができ、Raspberry Pi でサーバーを運用することも可能である。

Raspberry Pi には複数のモデルが存在する。本研究では、比較的高機能な Raspberry Pi 3 Model B と、性能は低いものの安価に入手可能な Raspberry Pi Zero WH を使用する。研究の初期段階では Raspberry Pi 3 Model B を利用していたが、より安価にロボットカーを導入できるようにするため、途中から Raspberry Pi Zero WH を利用するようにした。Raspberry Pi 3 Model B は CPU と し て ARM Cortex-A53（クアッドコア、クロック数 1.2GHz）、メモリを 1 G Byte 搭載しており、\$35 程度で入手可能である。それに対して、Raspberry Pi Zero WH は CPU と し て ARM1176JZF-S（シングルコア、クロック数 1GHz）、メモリを 512M Byte 搭載しており、\$14 程度

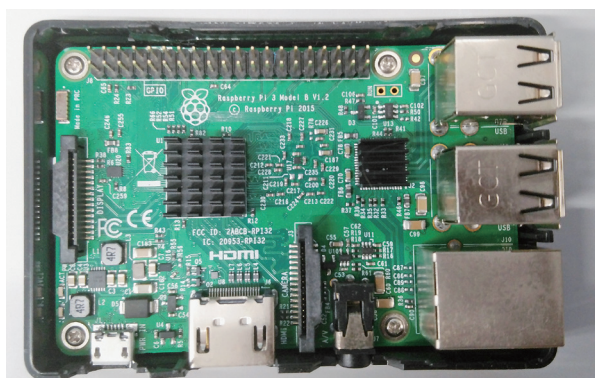


図 2 Raspberry Pi 3 Model B

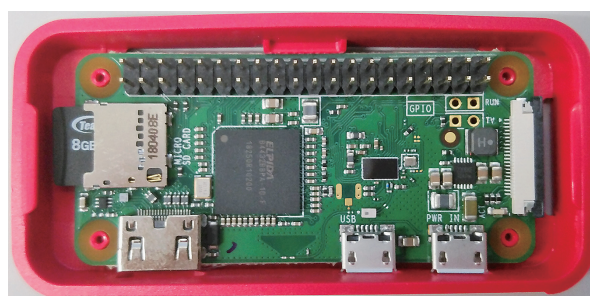


図 3 Raspberry Pi Zero WH

5 V	5 V	G N D	G P I O 14	G P I O 15	G P I O 16	G N D	G P I O 23	G P I O 24	G N D	G P I O 25	G P I O 8	G P I O 7	D N C	G N D	G P I O 12	G N D	G P I O 16	G P I O 20	G P I O 21
2	4	6	8	10	12	14	16	18	20	22	24	26	28	30	32	34	36	38	40
1	3	5	7	9	11	13	15	17	19	21	23	25	27	29	31	33	35	37	39
3.3 V	G P I O 2	G P I O 3	G P I O 4	G N D	G P I O 17	G P I O 27	G P I O 22	3.3 V	G P I O 10	G P I O 9	G P I O 11	G N D	D N C	G P I O 5	G P I O 6	G P I O 13	G P I O 19	G P I O 26	G N D

図 4 Raspberry Pi Zero WH のピン配置

で入手できる。また、両方とも無線 LAN や Bluetooth も搭載しており、他の機器と通信ができる。本研究で使用した Raspberry Pi 3 Model B を 図 2 に、Raspberry Pi Zero WH を 図 3 に示す。

Raspberry Pi 3 Model B 及び Raspberry Pi Zero WH には、外部デバイスと接続するためのピンが 40 ピンあり、それらと外部デバイスを接続して外部デバイスを制御することが可能である。両 Raspberry Pi のピン配置と各ピンの役割を 図 4 に示す。図中の中央の数字はピン番号を示している。また、GPIO は General Purpose Input / Output を省略したものであり、GPIO となっているピンは汎用的に利用できるピンである。

4. ロボットカー

本研究では、安価な部品を使ったロボットカー [5] を作成し、その動作を Raspberry Pi により制御する。ロボットカーを作成するための部品の一覧を表 1 に示す。これらの部品はすべて安価なものであり、Raspberry Pi Zero WH を利用すれば 1 台 7000 円程度でロボットカーを作成可能である。作成したロボットカーを 図 5 に示す。

Raspberry Pi によりロボットカーの動作を制御するため、ロボットカーと Raspberry Pi を接続する必要がある。配線図を 図 6 に示す。ロボットカーは 2 つのモーターを搭載しており、それぞれに順回転、逆回転、停止の命令を送ることができる。そのため、モータードライバとして TA2791P を使用し、Raspberry Pi と 2 つの TA2791P を接続することで、2 つのモーターをそれぞれ独立して制御することができる。

表 1 ロボットカーの部品

Raspberry Pi Zero WH
8GB microSD カード
ブレットボード
ジャンパーワイヤー
モバイルバッテリー
モータードライバ TA7291P
楽しい工作シリーズ No.100
楽しい工作シリーズ No.157
楽しい工作シリーズ No.164
楽しい工作シリーズ No.168

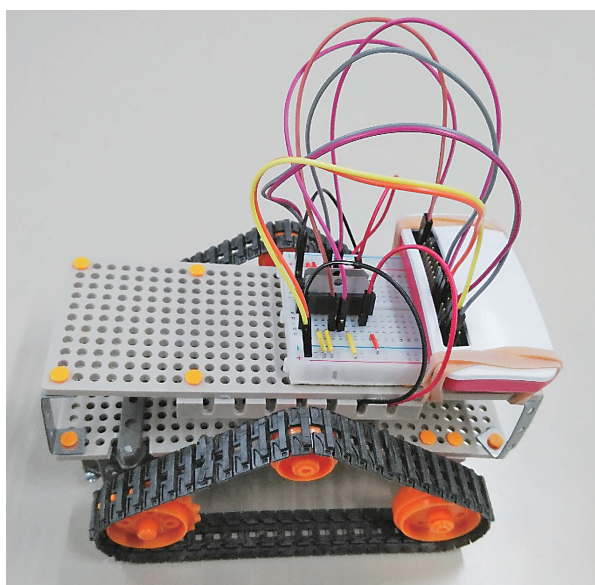


図 5 ロボットカー

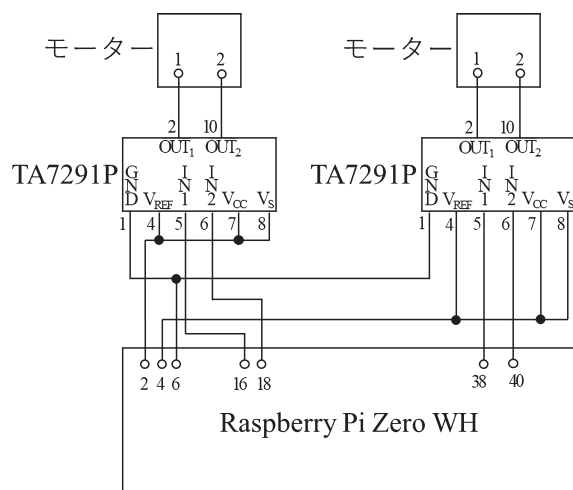


図 6 ロボットカー配線図

小学校段階ではロボットカーの作成を児童に行わせることは難しいと考えられるため、予め教員がロボットカーを作成しておくことが望ましい。中学校、あるいは高等学校で本ロボットカーをプログラミング教育に取り入れることを考える場合には、ロボットカーの

作成を生徒に行わせることも想定できる。生徒にロボットカーを作成させることにより、楽しみながら電子回路に触れることもでき、電子回路に対する知識を身に付けることもできる。

5. 教材概要

本研究で作成する教材の概要を図 7 に示す。本教材では、児童が操作するパソコンに Scratch 2 及び Python をインストールし、Scratch 2 にはラジコンカーを動作させるためのブロックを追加する Extension を読み込ませる。なお、今回追加するブロックは、forward (前進)、back (後退)、right (右回転)、left (左回転) の 4 つである。

ブロックが実行された際の通信を Python で作成された helper app で受信する。helper app を Python で作成する理由は、比較的理解しやすい言語であり、かつ、HTTP サーバーを簡単に構築することが可能であるためである。ブロックが実行された際、helper app に対してそれぞれ `http://127.0.0.1:12345/forward`、`http://127.0.0.1:12345/back`、`http://127.0.0.1:12345/right`、`http://127.0.0.1:12345/left` という HTTP 通信が発生するようにしている。なお、ポート番号 12345 は Extension ファイル内で設定されており、任意の値に変更することができる。

Scratch 2 から HTTP 通信を受信した helper app が受けた通信に応じた通信を Raspberry Pi 上で動作する HTTP サーバーに対して行う。Raspberry Pi 上の HTTP サーバーは、受信した内容に応じて Raspberry Pi の GPIO ピンを制御してロボットカーを動作させる。

Raspberry Pi 上の HTTP サーバーは Apache 等の一般的な Web サーバーではなく、Python で作成された簡易 HTTP サーバーを利用している。これは、Raspberry Pi は性能が低いため、なるべく負荷の少ないサーバーを利用したかったからである。

Raspberry Pi で動作する HTTP サーバーは、通信

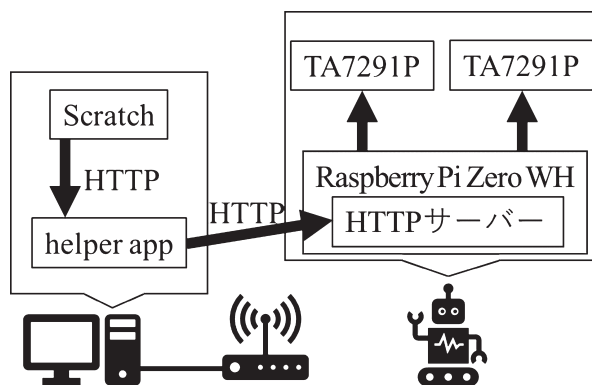


図 7 システム概要

表2 通信先アドレスと制御信号

通信先アドレス	制御信号	
	右側モーター	左側モーター
/forward	順回転	順回転
/back	逆回転	逆回転
/right	逆回転	順回転
/left	順回転	逆回転

先のアドレスに応じてモータードライバーへの制御信号をGPIOピンから送信する。通信先アドレスと制御信号の関係を表2に示す。

このような教材であれば、複数の端末から1台のロボットカーを動作させることができる。市販されている安価なロボット教材は1人1台が前提となっているものが多く、クラス全員分用意することが難しい。本教材は複数人で共有することを前提としており、導入しやすいと考えられる。

なお、今回作成したロボットカーは1台の端末から操作することを想定している。複数端末から操作する場合、命令の競合が起きることも想定する必要がある。命令の競合が起きないようにするには、先に命令を送信した端末のプログラムの処理が終了するまで他の端末からの命令を受け付けないようにする必要がある。これにはプログラムの終了ブロックを用意し、動作開始時からプログラム終了ブロックが実行されるまでロボットカーが他の命令を受け付けないようにすることで対応可能である。

6. 実験

提案したプログラミング用教材を用いて、教材の有用性を確認するための実験を行った。なお、本実験では、Raspberry Pi 3 Model Bを用いて実験を行っている。

まず、Scratch 2で作成したプログラムの通りにロボットカーが動作するか実験した。実験で作成したプログラムを図8に示す。なお、各ブロックが実行された際には、各動作を2秒間行うようにしている。この



図8 作成したプログラム

プログラムを実行したところ、ロボットカーは2秒間前に前進し、2秒間右回転するという動作を2回繰り返し、その後2秒間左回転し、そして2秒間後進した。作成したプログラムの通りの動作をロボットカーがすることを確認した。

次に、アンケートによる評価実験を行った。教材として使用することを想定し、本学の学生に提案教材を利用してもらった後評価してもらった。実験に参加した学生は本学の4年生12名であった。まず、事前準備として、Raspberry Piで簡易HTTPサーバーを起動する作業を行ってもらった。次に、Extensionを読み込み、helper appを起動する作業を行ってもらった。その後、プログラムを自由に作成し、実行する作業を行ってもらった。このような作業を行った後、アンケートによる教材の評価を行った。アンケートの主な設問を表3に示す。各設問は5段階で評価するようにしており、数字が大きいほどよい評価であったことを示している。それらの設問に対する集計結果を表4に示す。

設問1では、多くの学生が操作には問題ないと回答しており、使いやすい教材であることが分かる。また、設問2や設問4、設問6の結果から、楽しみながらプログラミングを学ぶことのできる教材であることも確認できた。設問3や設問5の結果から、本教材を授業に利用できそうであることも分かった。

一方、本教材を使用した後の感想を聞いていく中で、本教材の問題点も出てきた。Scratch 2の停止ボタンを押してもロボットカーがすぐに止まらない、動作が秒数（2秒）で制御されているのでわかりにくい、右回転、左回転は角度で制御できるようにするとわかりやすいといった意見があった。また、本教材を利用す

表3 アンケートの設問

設問No.	設問
1	Scratch 2の操作は問題ないか
2	子どもの興味をひきそうか
3	プログラミング教育に利用できる教材になりうと思うか
4	(子ども目線で) 分かりやすかったか
5	授業で利用してみたいか
6	楽しかったか

表4 アンケート結果

設問No.	回答人数				
	1	2	3	4	5
1	0	0	1	2	9
2	0	0	0	3	9
3	1	0	1	7	3
4	0	2	0	7	3
5	1	0	0	8	3
6	0	0	0	3	9

るためのマニュアルや本教材を扱う際に起こるトラブルを解決するためのマニュアルが必要であるという意見も多かった。

これらの結果を受け、より活用しやすいように教材の改良を行った。実験ではRaspberry Pi 3 Model Bを利用していたが、Raspberry Pi Zero WHを使用するようにした。これにより、より安価にロボットカーを作成することができるようになった。Raspberry Pi Zero WHはRaspberry Pi 3 Model Bと比較すると非力で、GUIによる操作ができないため、極力Raspberry Pi上で作業をしなくてもよいように工夫もした。具体的には、Raspberry Pi Zero WHを起動した際に簡易HTTPサーバーが自動起動するようにした。また、児童の作業をなるべく減らすようにもした。事前準備としてExtensionを読み込ませた状態でプログラムを保存しておけば、それをScratch 2から開けばExtensionが読み込まれた状態で開くことができるため、Extensionを読み込んだ状態で保存されたファイルを使用するようにした。helper appを簡単に起動できるようにバッチファイルも用意した。更に、1つのブロックを実行する際に2秒間動作させるのではなく、動作する秒数を指定できるようにした。これらの改良により、より簡単に本教材を利用できるようになった。

7. おわりに

本研究では、Scratch 2から操作できるロボットカーを用いたプログラミング教育用教材の開発を行った。Scratch 2からリモートのロボットカーを動作させることができるよう、ブロックを追加し、helper appを作成した。また、Raspberry Pi上で動作する簡易HTTPサーバーを作成し、アクセス先に応じてGPIOの制御を行うことでロボットカーを動作させた。このようなロボットカーを用いることで、一つのロボットカーを複数の児童が共用できるようになり、より安価に導入できる教材を開発できた。

実践の結果より、目新しさや面白さなどの観点で子どもの興味を惹く教材であることが分かった。また、授業をスムーズに行えるようにするための工夫として、教師自身が教材についての知識を十分に備えられるようにすることや、マニュアルなどを用意することが必要とされることも分かった。

今後の課題として、プログラムの実行途中でロボットカーの動作を停止できるようにする、左右の旋回動作を角度で指定できるようにする、マニュアルを作成する、などが挙げられる。また、今回作成したロボットカーにはセンサーを全く付けていない。センサーを付けることにより、より高度なプログラミングを行うことが可能となるため、センサーの付いたロボット

カーによるプログラミング環境の構築も今後の課題である。これらの課題を解決し、より有用な教材にしていきたい。

参考文献

- [1] 新学習指導要領のポイント（情報教育・ICT 活用関係）：http://www.mext.go.jp/component/a_menu/education/micro_detail/_icsFiles/afiedfile/2018/07/20/1407394_2_1.pdf
- [2] Scratch: <https://scratch.mit.edu/>
- [3] 小学校を中心としたプログラミング教育ポータル: <https://miraino-manabi.jp/>
- [4] Creating Scratch 2.0 Extensions: <https://en.scratch-wiki.info/w/images/ExtensionsDoc.HTTP-9-11.pdf>
- [5] ラズパイで作ろう！ ゼロから学ぶロボット製作教室:
<http://itpro.nikkeibp.co.jp/atcl/column/15/040800081/>

(2019 年 9 月 17 日受理)