

小学校と短期大学で開発する プログラミング学習コースの開発と評価

野津 伸治 (Shinji NOTSU)

【目 的】

ここ数年 IoT (Internet of Things) が世界的に進展している。そのことも踏まえて文部科学省は 2020 年度から全国の小学校でもプログラミング教育を始める。湯梨浜町立東郷小学校でも高学年でのプログラミング・クラブでこれへの準備として 2017 年度 1 学期から研究開発を始められる。鳥取短期大学生生活学科情報・経営専攻でも 2009 年度からタブレット/スマートフォンを授業と学生生活で活用している。その中では電子書籍 (ePub) の作成、アプリケーション開発、ネットワークセキュリティ、組込系開発などを鳥取短期大学学生が行っている。両校の教員と学生・児童が両校の授業で利用するプログラミング学習コースの開発と評価を行った。

○共同研究者・協力者 田中 靖浩 (湯梨浜町立東郷小学校 教諭)

【活動 (研究) の概要】

2017 年 4 月から 7 月まで鳥取短期大学生生活学科情報・経営専攻 1 年生の「情報処理実務」で「ライントレースロボット」のマイコン (Arduino) によるハードウェア設計・ソフトウェアによる赤外線センサと DC モータ制御プログラミング (processing) ・オシロスコープによる解析・設計した回路図の CAD での製図などをグループワークで行い、設定コースの走行性能を課題解決型学習目標に設定して実施した。

2017 年 9 月 16 日に中部こども科学まつりで「残像表示装置のハードウェア製作と表示文字列のプログラミング」を小学生 20 名に対して行った。

2017 年 12 月 16 日に東郷小学校で「残像表示装置のハードウェア製作と表示文字列のプログラミング」を小学生 20 名に対して行った。

2017 年 12 月 26 日に鳥取短期大学で行われた「鳥取県メディア教育研究会」の第 4 回研修会で県内外の小学校・中学校・高等学校・企業の 25 名に対して「ロボットプログラミングのワークショップ」でマイコンによる赤外線センサと DC モータの PWM 制御による自立走行のプログラミングを processing と Scratch で今後児童・生徒に指導する立場で基本概念を体験する機会を提供した。

2017 年 11 月 6 日に東郷小学校で児童 20 名にこちらが組み立てたマイコン制御の DC モータカーを持ち込み、processing による直進・後退・左折・右折の実装を行った。

2017 年 11 月 20 日に東郷小学校で児童 20 名に、各種ライントレースカーや 4 足歩行・2 足歩行ロボットを持参し、動作原理や利活用について説明をした。

2017 年 12 月 4 日に東郷小学校で児童 20 名に、ハードウェアパーツの個体差等での動作の補正(期待する動作と実際の動作の差) をプログラミングで行うため、動作の計測とプログラミングを行った。

2017 年 12 月 18 日に東郷小学校で児童 20 名に、赤外線 LED と受光器がセットになったフォトリ

フレクタを左右のセンサーユニットとして自作・持参して、動作環境での白黒判定の閾値を検出してプログラムに反映させた。

2018年1月22日に東郷小学校で児童20名に、まとめと発表の準備指導を行った。

2018年2月19日に東郷小学校で児童20名は5グループに分かれてまとめの発表を行い、指導助言を行った。

なお、表1の部品を使い、図1のように設計を行い、図2のように実証した。また表2のソフトウェアを利用して図3のようにprocessing言語で制御を行った。さらに教員と企業人向けにマイコンによる赤外線センサによるDCモータ制御のワークショップも行った(図4、図5、図6)。

表1 設計したライトレースロボットのハードウェア一覧

部品名	個数	単価 (円)	金額 (円)	備考
シャーシ (DCモータ、ギアボックス、タイヤ)	1	1,490	1,490	
Arduino UNO R3	1	435	435	要 CH340 ドライバ
モータドライバ(TA7291P)	2	150	300	
ブレッドボード	2	200	400	
電池ボックス(単 3x4 本、リード線付き)	2	60	120	
フォトフレクタ (LBR-127HLD)	2	50	100	
抵抗(220Ω)	2	1	2	
抵抗(1kΩ)	2	1	2	
ジャンパーワイヤー(オス-オス)	1	100	100	
合計			2,949	

表2 利用したソフトウェア一覧

ソフトウェア名	バージョン	価格 (円)	備考
Arduino IDE	1.8.5	0	processing 言語統合開発環境
CH340 ドライバ	3.4	0	USB シリアルドライバ
S4A	1.6	0	Scratch-processing 変換環境

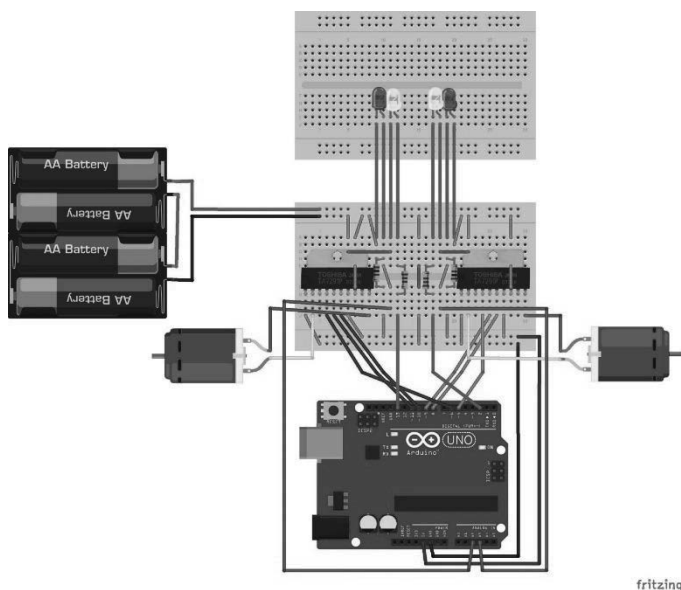


図1 設計したライトレースロボットの実際配線図

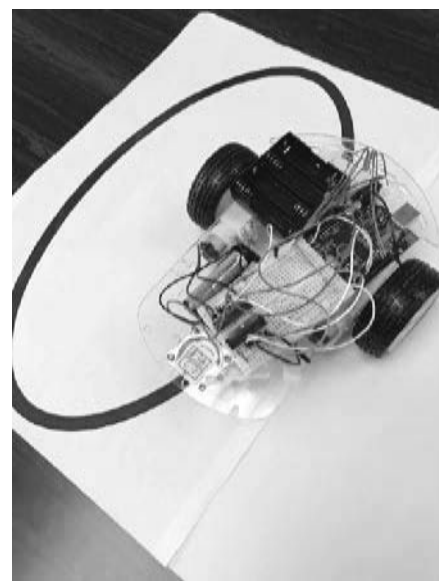


図2 設計したライトレースロボットの実際

```

> 0 |10 |20 |30 |40 |50 |60 |70
1  /*
2  const int lPin4 = 3; // 左モータ (Vref) ↑
3  const int lPin5 = 2; // 左モータ (IN1) ↑
4  const int lPin6 = 1; // 左モータ (IN2) ↑
5  const int rPin4 = 6; // 右モータ (Vref) ↑
6  const int rPin5 = 5; // 右モータ (IN1) ↑
7  const int rPin6 = 4; // 右モータ (IN2) ↑
8  const int lLED = 13; // 左LED ↑
9  const int rLED = 12; // 右LED ↑
10 /*
11 int lVal = 0; // 左センサの取得値の初期値 ↑
12 int rVal = 0; // 右センサの取得値の初期値 ↑
13 /*
14 int lAlpha = 110; // 左センサの値がこの値以下なら白と判定する閾値 ↑
15 int rAlpha = 110; // 右センサの値がこの値以下なら白と判定する閾値 ↑
16 /*
17 int forward(int pin3, int analog, int pin1, int pin2); // 車体を前進させる関数 ↑
18 int stop(int pin1, int pin2); // 車体を停止させる関数 ↑
19 ↑
20 void setup() { ↑
21 /*
22 pinMode(lPin4, OUTPUT); // 左モータ (Vref) 用に出カとして設定 ↑
23 pinMode(lPin5, OUTPUT); // 左モータ (IN1) 用に出カとして設定 ↑
24 pinMode(lPin6, OUTPUT); // 左モータ (IN2) 用に出カとして設定 ↑
25 pinMode(rPin4, OUTPUT); // 右モータ (Vref) 用に出カとして設定 ↑
26 pinMode(rPin5, OUTPUT); // 右モータ (IN1) 用に出カとして設定 ↑
27 pinMode(rPin6, OUTPUT); // 右モータ (IN2) 用に出カとして設定 ↑
28 pinMode(lLED, OUTPUT); // 左LED用に出カとして設定 ↑
29 pinMode(rLED, OUTPUT); // 右LED用に出カとして設定 ↑
30 /*
31 Serial.begin(9600); // シリアル通信速度を9600に設定 ↑
32 } ↑
33 ↑
34 void loop() { ↑
35 /*
36 digitalWrite(lLED, HIGH); // 左LEDを点灯 ↑
37 digitalWrite(rLED, HIGH); // 右LEDを点灯 ↑
38 /*
39 lVal = analogRead(1); // 左センサをアナログ1で入力 ↑
40 rVal = analogRead(2); // 右センサをアナログ2で入力 ↑
41 Serial.print("左="); // 左の取得値を表示 ↑
42 Serial.print(lVal); // 左の取得値を表示 ↑
43 Serial.print("右="); // 右の取得値を表示 ↑
44 Serial.println(rVal); // 右の取得値を表示 ↑
45 /*
46 if (lVal < lAlpha && rVal < rAlpha) | // 左センサ=白 かつ 右センサ=白 ↑
47 forward(lPin4, 255, lPin5, lPin6); // 左→を正転(出力調整: 素=0~255=最大) ↑
48 forward(rPin4, 255, rPin5, rPin6); // 右→を正転(出力調整: 素=0~255=最大) ↑
49 delay(40); // この状態を40ms間維持 ↑
50 } ↑
51 /*
52 else if (lVal > lAlpha && rVal < rAlpha) | // 左センサ=黒 かつ 右センサ=白 ↑
53 stop(lPin5, lPin6); // 左→を停止 ↑
54 forward(rPin4, 255, rPin5, rPin6); // 右→を正転(出力調整: 素=0~255=最大) ↑
55 delay(30); // この状態を30ms間維持 ↑
56 } ↑
57 /*
58 else if (lVal < lAlpha && rVal > rAlpha) | // 左センサ=白 かつ 右センサ=黒 ↑
59 forward(lPin4, 255, lPin5, lPin6); // 左→を正転(出力調整: 素=0~255=最大) ↑
60 stop(rPin5, rPin6); // 右→を停止 ↑
61 delay(30); // この状態を30ms間維持 ↑
62 } ↑
63 /*
64 else if (lVal > lAlpha && rVal > rAlpha) | // 左センサ=黒 かつ 右センサ=黒 ↑
65 forward(lPin4, 255, lPin5, lPin6); // 左→を正転(出力調整: 素=0~255=最大) ↑
66 forward(rPin4, 255, rPin5, rPin6); // 右→を正転(出力調整: 素=0~255=最大) ↑
67 delay(40); // この状態を40ms間維持 ↑
68 } ↑
69 /*
70 stop(lPin5, lPin6); // 左モータを停止 ↑
71 stop(rPin5, rPin6); // 右モータを停止 ↑
72 delay(250); // この状態を250ms間維持 ↑
73 } ↑
74 ↑
75 /*
76 int forward(int pin3, int analog, int pin1, int pin2) | // 車体を前進させる関数 ↑
77 analogWrite(pin3, analog); ↑
78 digitalWrite(pin1, HIGH); ↑
79 digitalWrite(pin2, LOW); ↑
80 } ↑
81 /*
82 int stop(int pin1, int pin2) | // 車体を停止させる関数 ↑
83 digitalWrite(pin1, LOW); ↑
84 digitalWrite(pin2, LOW); ↑
85 } ↑
86 ↑
87 [EOF]

```

図3 processing 言語によるライトレースロボットの制御プログラム

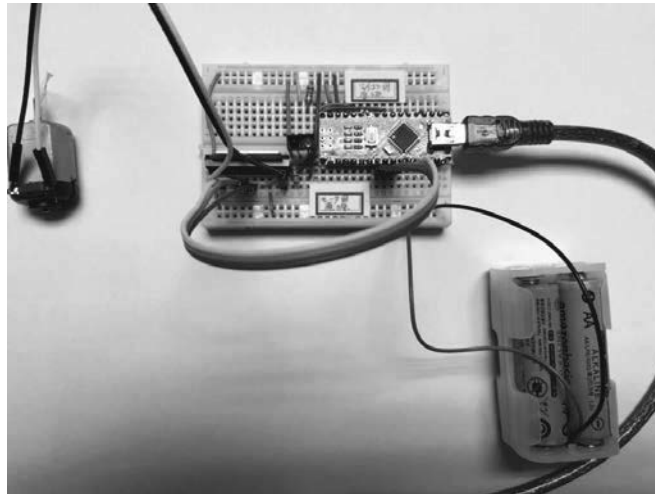


図4 教員と企業人向けワークショップ（フォトリフレクタと DC モータを Arduino で制御）

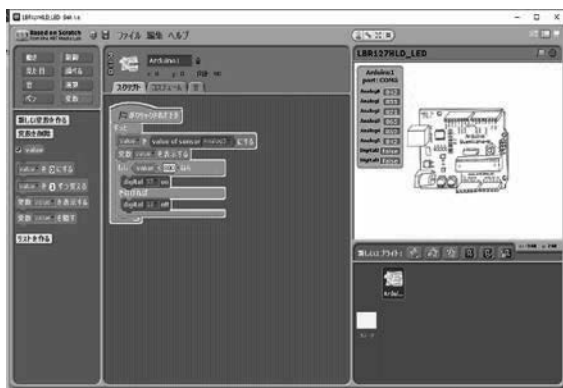


図5 フォトリフレクタ LBR-127HLD を Scratch で検出するコード

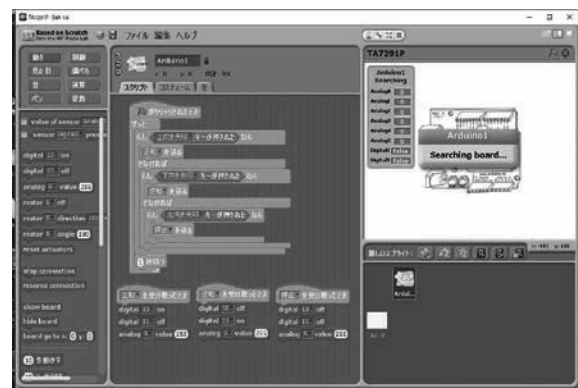


図6 DC モータをモータドライバTA7291P 経由で制御する Scratch のコード

【成果】

IoT (Internet of Things) として製品群はハードウェアとソフトウェアが統合されたものである。このことの理解として、鳥取短期大学生生活学科情報・経営専攻学生1年生42名は90分15回の授業の中でハードウェア部品の電気的特性、回路の設計、動作の計測を行った。小学生4年生から6年生20人は7回45分のクラブ活動として概念の理解にとどめ、著者が完成させてハードウェアを提供した。またデジタル機器の設計手法としてハードウェアの個体差をソフトウェアで吸収する開発方針を取り、鳥取短期大学生生活学科情報・経営専攻学生には5グループでの課題解決学習として左右の赤外線センサの白黒の検出値の閾値を計測させてから、左右のDCモータの出力をPWM (Pulse Width Modulation) 値でprocessingプログラムの実装を通して調整させた。小学生の5グループでは、著者のprocessingプログラムのテンプレートにおいて左右の赤外線センサの白黒の閾値を計測してから定義する手順を実行させたのち、DCモータの出力調整値PWMの調整を評価コース（白地に幅1cmの黒線を直線、S字、楕円、真円）で行った。指示内容が明確ならば45分の限られた時間で各グループが到達することができた。

2020年開始の小学校でのプログラミング教育の必修化でビジュアル・プログラミングとしてScratchが注目されているが、コマンドベースのprocessingでも可能なことが検証された。また県内外の教員と企業人25名参加の120分でのマイコンを使った赤外線センサによるDCモータ制御のワークショップを行い、processingとScratchの比較も行った。さらに競合する製品群と比較してもコストが1/5から1/3であるメリットも紹介し納得していただいた。教材として提供する企業にも海外からの部品調達から設計・プログラミングの技術アドバイスも行った。