

Raspberry Pi 4 を使った集中講義の効果

土肥 紳一¹⁾, 齊藤 剛¹⁾

1) 東京電機大学 システムデザイン工学部

dohi@mail.dendai.ac.jp

The effect of intensive lecture using Raspberry Pi 4

Shinichi Dohi¹⁾, Tuyoshi Saitoh¹⁾

1) School of System Design and Technology, Tokyo Denki Univ.

概要

2020年1月15日(水)にRaspberry Pi 4を使い、LEDを点滅させる集中講義を開催した。受講対象者はシステムデザイン工学部デザイン工学科の1年生から3年生である。受講定員は20名で、主な内容はOS(Raspbian:ラズビアン)のインストール、ネットサーフィン、C言語を使ってLEDを点滅させるプログラムの実行である。13:40から約3時間といった短時間での開催であったが、1つのLEDの点滅は全員達成できた。集中講義の最後に実施したアンケート調査結果から、その効果について述べる。

1 はじめに

切っ掛けは、昨年の夏にRaspberry Pi 3を手にしたことである[1]。マウス、キーボード、ディスプレイを接続しOSをインストールすると、デスクトップが表示され、一昔前のノートPCと同程度の性能を持ったコンピュータが手に入ることに驚愕した。しかも本体価格は数千円である。コンピュータリテラシーやコンピュータプログラミングの授業を担当する1人として、Raspberry Pi 3をこれらの授業で活用できる可能性を探り、AXIES2019で講演発表を行った[2]。発表会場で、年明け早々に、集中講義の開催を予定していることを述べた。本論文は、集中講義に至った経緯、実施内容、アンケート調査による受講者のモチベーションの推移、受講者の声等を分析し、集中講義の効果を述べる。

デザイン工学部に改編され、エクステンションプログラムは中断した。

2.2 委員会での発言

昨年の9月下旬に開催された情報戦略会議で、今後の情報システムの導入について、本学らしさは何かと聞かれ、Raspberry Pi 3を紹介した。これはノートPCにGPIOが付いた様なコンピュータでプログラムによってハードウェアを容易に制御でき、プログラミング教育の視野が広がることを述べた。他の委員の中にも、その使用経験者がおり、Raspberry Pi 3を並べた教室があつても良いのではとの話で盛り上がった。図1はPC教室のシンクライアントをRaspberry Pi 3に置き換えたイメージである。その後、システムデザイン工学部デザイン工学科を対象に集中講義を開催することを決め、物品の手配と講義内容の検討に入った。

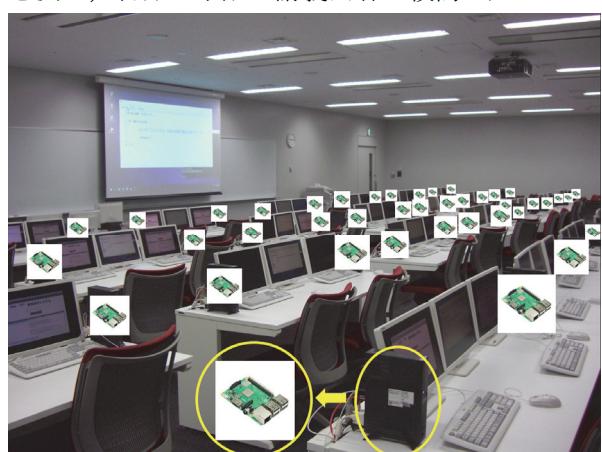


図1 シンクライアントをRaspberry Pi 3に

2 実施に至った経緯と準備

2.1 エクステンションプログラムでの実績

今回の集中講義を開催する前に、情報環境学部の時代に、2014年から2016年にかけて「コンピュータプログラミングA」を教わった後のハードウェア入門の集中講義を、エクステンションプログラムとして開催した経験があり、AXIES2014で講演発表を行った[3]。当時はArduinoを使用した[4]。その後、情報環境学部は2017年にシステム

2.3 使用した主なパーツ等

当初は Raspberry Pi 3 のスターターキットを購入予定であった。これは品薄状態が続いている最中に、タイミング良く Raspberry Pi 4 が販売され、単品で準備することに切り替えた。使用した主なパーツ等を表 1 と図 2 に示す。

表 1 使用した主なパーツ等

No	品名
1	Raspberry Pi 4 Model B 4GB(Element14 製)
2	microSD(32GB)
3	AC アダプター 5.1V/3.0A USB Type-C
4	ヒートシンク(Pimoroni COM1203)
5	テキスト、たのしくできる Raspberry Pi とブレッドボードで電子工作[5]
6	ジャンパー線 15cm(黒、赤、青、白、黄)
7	赤色 LED(OSR5JA5E34B)
8	黄緑色 LED(OSG8HA5E34B)
9	黄色 LED(OSY5JA5E34B)
10	カーボン抵抗(炭素皮膜抵抗) 1/4W 220Ω
11	ブレッドボード 6 穴版(EIC-3901)
12	HDMI マイクロ変換アダプタ(タイプ D)
13	HDMI-DVI 変換ケーブル(DH-HTD10BK)
14	パーツケース(M43-10)



図 2 使用した主なパーツ等

受講定員は 20 名とし、20 セットを準備した。集中講義の会場は、総合メディアセンターが管理する東京千住キャンパスの PC 教室 4 を利用した。理由は、コスト削減のためにキーボード、マウス、ディスプレイを借用すること、そして図 1 のイメージを実現することである。先行して 1 セット分のパーツ等を購入し、PC 教室 4 で接続テストを行い、問題ないことを確認してから残りのセット数のパーツを購入した。パーツケースは丁度良い

大きさのものを探すのに時間を要した。これも 1 つ先行手配し、上手く収まることを確認してから残りのセット数を購入し、集中講義開催前日に届くなど、何とか間に合った状況である。

2.4 パーツ等の調達で分かったこと

パーツ等の調達で分かったことが、いくつかある。Raspberry Pi 3 と Raspberry Pi 4 では AC アダプターの形状が異なり、使い回しができない。ジャンパー線はオスメスが必要であり、Arduino のオスオスは兼用できない。Raspberry Pi 4 は発熱量が多いため、ヒートシンクは必須である。

Arduino には AD コンバータが付いていたが、Raspberry Pi には無い。1 つの GPIO を使って、1 ビットの変化は検出できるが、アナログのデータを直接扱うことはできない。これを実現するためには、別途、ハードウェアを組む必要がある。

3 実施内容

3.1 受講者の様子

受講学年の内訳を表 2 に示す。受講希望者は合計 21 名であった。定員は 20 名のため、1 セットは 2 人で担当してもらった。学年の割合は 3 年生の受講が最も多く 57.1%，2 年生は 28.6%，1 年生は 14.3% であった。

表 2 受講者の学年の内訳

学年	人数	%
3 年生	12	57.1
2 年生	6	28.6
1 年生	3	14.3
合計	21	100.0

3.2 主な内容(3 時限)

3 時限(13:40～15:20)は、OS のインストールである。主な内容を表 3 に示す。インストーラ(NOOBS)のダウンロードのサイズは、約 2GB ある。

表 3 主な内容(3 時限)

No	主な内容
1	配布物の確認および注意事項
2	インストーラ(NOOBS)のダウンロード
3	Raspbian のインストール用 microSD の準備
4	キーボード、マウス、ディスプレイの接続
5	Raspbian のインストール
6	Raspbian のデスクトップのセットアップ
7	Raspbian のデスクトップの利用

事前にダウンロードをためしたところ、2時間程度かかった。集中講義では時間的に体験が難しいため、事前にダウンロードしたファイルをmicroSDに入れ配布した。

15:00頃には、全員がOSのインストールを終え、デスクトップが利用できるようになった。webブラウザやMathematica等のアプリケーションが利用できることをためした。Officeに類似したアプリケーションもあるが、操作性がかなり異なることも分かった。1年前期で学んだコンピュータリテラシーのいくつかが、Raspberry Pi 4で利用できることを体験した。今回の集中講義では時間の関係で割愛したが、Processingも利用でき、コンピュータプログラミングIの授業内容にも対応できる。

3.3 主な内容(4時間)

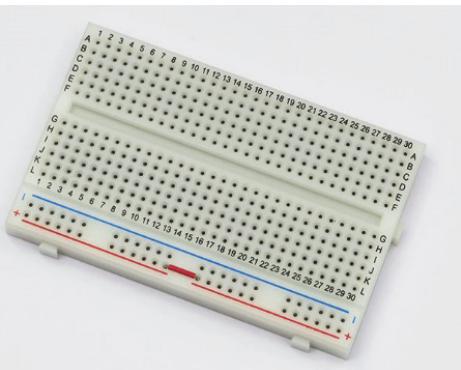
4時間(15:30~17:10)は、LEDの点滅を行うプログラムの実行である。主な内容を表4に示す。全員がクリヤする目標は1つのLEDの点滅とした。

表4 主な内容(4時間)

No	主な内容
1	C言語プログラムの入力、コンパイル、実行
2	1つのLEDの点滅(全員がクリヤする目標)
3	2つのLEDの点滅
4	3つのLEDの点滅
5	練習問題(3つのLEDを使用)
6	アンケート調査
7	キーボード、マウス、ディスプレイの後始末
8	配布物の回収

3.4 1つのLEDの点滅に使用したパーツ等

1つのLEDの点滅に使用したパーツ等は、図3から図5に示す。配線例は、表5に示す。もちろん、これ以外の配線方法もある。なお、ジャンパー線は、オスメスが必要である。



<https://akizukidenshi.com/catalog/>より引用

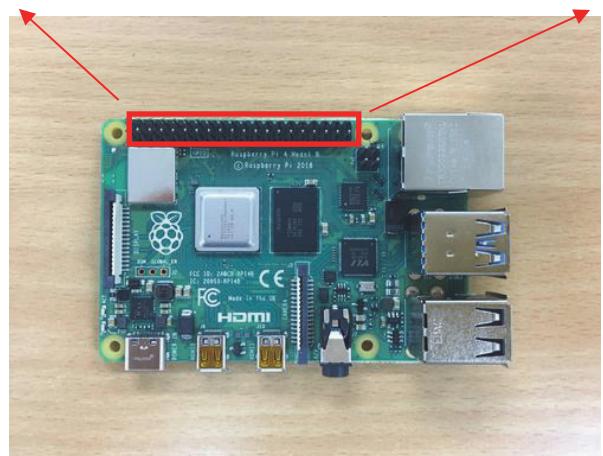
図3 ブレッドボード



<https://akizukidenshi.com/catalog/>より引用

図4 1つのLEDの点滅に使用したパーツ

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



<https://www.switch-science.com/catalog/5680/>より引用

図5 Raspberry Pi 4のピン配置

表5 1つのLEDの点滅の配線例

GPIOとジャンパー線	ブレッドボード側
GPIO10(19番) 赤色 GND(6番) 黒色	L21 LEDのカソード L22 LEDのアノード K22とK29の間に220Ω G29 赤色のジャンパー線 G21 黒色のジャンパー線

3.5 プログラムの入力、コンパイル、実行

最初は、「Hello TDU!」の文字列を表示するプログラムをためした。プログラムの入力、コンパイル、実行を行う。この様子を図6に示す。要領がつかめたところで、ハードウェアの接続を行い、これを制御するプログラムをためした。受講者の様子を図7に示す。集中講義の資料はwebで公開した[6]。最近の学生はタブレット等の情報端末を別途準備し、これを資料の閲覧用として利用する人が増えている。

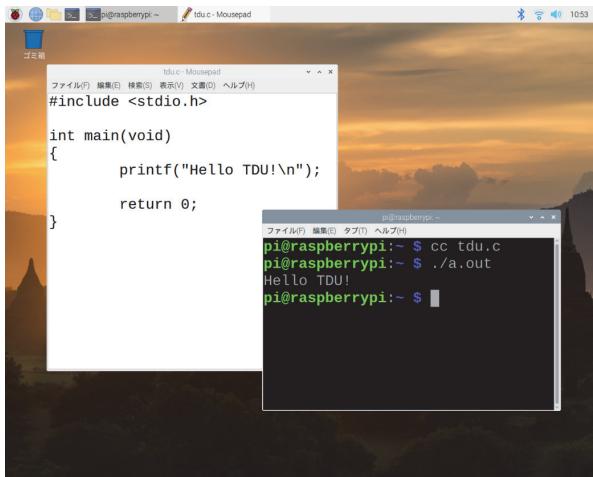


図 6 プログラムの入力とコンパイル実行の様子

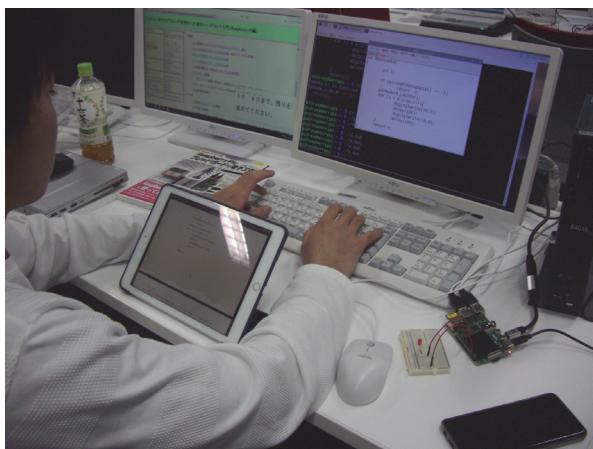


図 7 1つのLEDの点滅に取組む受講者

受講者はデジタルネイティブな世代であり、このような利用は、ごく当たり前の様である。筆者は、1つのデスクトップ上に複数のウインドウを表示し、適宜、切り替えながら使うことが多いが、世代の違いによる情報端末の扱い方の差が現れた。

テキスト p21 の「8-3 ソースファイルの作成」に記載されているプログラムを利用した。このプログラムは 100ms 毎に LED の点灯と消灯を 10 回繰り返す。テキストのプログラムは `pinMode(10, OUTPUT);` が抜けており訂正した。その他、筆者が気付いた部分を赤字で加筆修正した。この様子をプログラム 1 に、点滅の様子を図 8 に示す。

```
#include <wiringPi.h>

int main(void)
{
    int i; // 制御変数

    if (wiringPiSetupGpio() == -1) // GPIO の利用判定
        return -1;
    pinMode(10, OUTPUT); // GPIO10 を出力に設定
```

```
for (i = 0; i < 10; i++) {
    digitalWrite(10, 1); // GPIO10 をハイに
    delay(100); // 100ms の遅延
    digitalWrite(10, 0); // GPIO10 をローに
    delay(100); // 100ms の遅延
}
```

プログラム 1

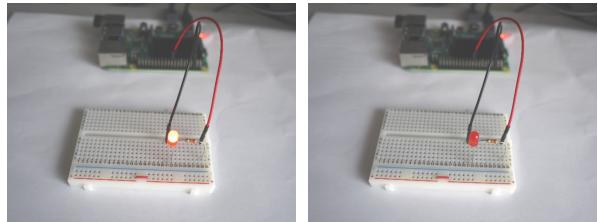


図 8 1つのLEDの点滅の様子

3.6 オブジェクト形式のマクロの活用

LED の個数が増えてくると、GPIO の番号をプログラム中に直接数字で記述することを続けると、混乱が生じる。これを避けるためにオブジェクト形式のマクロを定義し、LED との関係を分かり易く表現する工夫が効果的である。プログラムの可読性を向上するために、オブジェクト形式のマクロが効力を発揮する良い例である。要点はプログラム 2 に赤字で示した。

```
#include <wiringPi.h>

#define R_LED 10 // 赤色 LED
#define T 100 // 遅延時間(ms)
#define LOOP 10 // 繰り返し回数

int main(void)
{
    int i; // 制御変数

    if (wiringPiSetupGpio() == -1) // GPIO の利用判定
        return -1;
    pinMode(R_LED, OUTPUT); // 出力に設定
    for (i = 0; i < LOOP; i++) {
        digitalWrite(R_LED, HIGH); // 点灯
        delay(T); // 遅延
        digitalWrite(R_LED, LOW); // 消灯
        delay(T); // 遅延
    }
    return 0;
}
```

プログラム 2

3.7 3つのLEDの点滅の例

受講者の中には日常的に電子工作を行っている人も含まれている。初心者からベテランまで幅広

く満足してもらえるように、LED の個数を 2 つ、3 つと増やせるように配慮した。LED の数が増えてくると、配線が複雑になり、GPIO のピンも複数接続する必要がありジャンパー線が密集してくる。どのピンをどの LED に割り当てるかは、受講者に任されるが、配線の誤りの確認作業を軽減するために表 6 を示し、これに合わせてもらった。

表 6 3 つの LED の点滅の配線例

GPIO とジャンパー線	ブレッドボード側
GPIO10(19 番) 赤色 GND(6 番) 黒色	L21 赤 LED のカソード L22 赤 LED のアノード K22 と K29 の間に 220Ω G29 赤色のジャンパー線 G21 黒色のジャンパー線
GPIO9(21 番) 黄色 GND(14 番) 黒色	L11 黄 LED のカソード L12 黄 LED のアノード K12 と K19 の間に 220Ω G19 黄色のジャンパー線 G11 黒色のジャンパー線
GPIO11(23 番) 青色 GND(20 番) 黒色	L1 黄緑 LED のカソード L2 黄緑 LED のアノード K2 と K9 の間に 220Ω G9 青色のジャンパー線 G1 青色のジャンパー線

プログラムを実行すると、右端の赤色の LED、中央の黄色の LED、左端の黄緑色の LED が順に点滅し、これを 10 回繰り返す。この様子を図 9 に示す。LED の配置は交通信号機の順に合わせた。

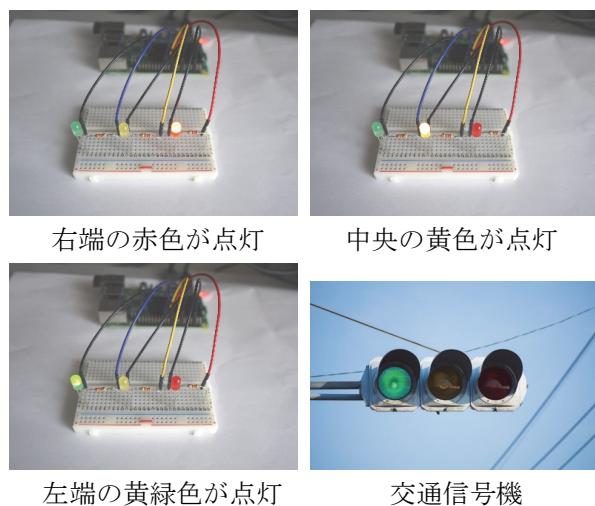


図 9 3 つの LED の点滅の様子と交通信号機

3.8 配列を活用した 3 つの LED の点滅

オブジェクト形式のマクロと配列を組み合わせ

ることによって、個々の LED を配列の要素の 1 つとして扱えるようになり、プログラムの可読性がさらに向上する。このような活用の効果は、ハードウェアを活用しない通常のプログラミングの授業では体験することができない。また、繰り返しの仕組みは LED の点滅が順に移動して行くことで理解させることができる。要点はプログラム 3 に赤字で示した。

```
#include <wiringPi.h>

#define R_LED 10          // 赤色 LED
#define Y_LED 9           // 黄色 LED
#define G_LED 11          // 黄緑色 LED
#define T 100              // 遅延時間(ms)
#define LOOP 10            // 繰返し回数
#define N 3                // LED の個数

int main(void)
{
    int i;                  // 制御変数
    int j;                  // 制御変数
    int led[] = {R_LED, Y_LED, G_LED}; // 配列活用

    if (wiringPiSetupGpio() == -1) // GPIO の利用判定
        return -1;
    for(i = 0; i < N; i++)      // N 回繰返し
        pinMode(led[i], OUTPUT); // 出力に設定
    for (i = 0; i < LOOP; i++) { // LOOP 回繰返し
        for (j = 0; j < N; j++) { // N 回繰返し
            digitalWrite(led[j], HIGH); // 点灯
            delay(T);               // 遅延
            digitalWrite(led[j], LOW); // 消灯
            delay(T);               // 遅延
        }
    }
    return 0;
}
```

プログラム 3

3.9.3 つの LED の振舞を変更

表示の順番を逆順にする場合、大きく 2 つの方法が存在する。一つは、ハードウェアの配線を変更する方法である。もう一つは、プログラムで表示の制御部分を変更する方法である。集中講義の狙いとしては、色々な対処方法があることを体験することもある。ハードウェアで解決すると振舞ごとに複数のハードウェアを作成することになり、生産性が低下する。同じハードウェアをソフトウェアで制御できることの重要性を体験できる。

作業の早い人には、3 色の LED を同時に点滅する振舞、交通信号機を模倣して赤色の LED が 3 秒、黄色の LED が 1 秒、黄緑色の LED が 5 秒点灯を 10 回繰り返す振舞を考えさせ、プログラム 3 を少し書き換えるだけで実現できることを体験した。

4 アンケート調査結果の分析

4.1 アンケート調査結果

集中講義の効果を分析するために、集中講義の最後に、アンケート調査を実施した。調査方法は、アンケート項目を記載した Excel のワークシートを web サイトで配布し、メールに添付する方法で回収した。理解度と経験等に関する設問項目は表 7 に、回答結果は図 10 に示す。

表 7 主なアンケート調査項目

No	設問内容
1	半田付けを行った経験はありますか。
2	電子工作を行った経験はありますか。
3	Raspberry Pi を知っていましたか。
4	インストーラ(NOOBS)のダウンロードは、理解できましたか。
5	OS のインストールは理解できましたか。
6	LED の電極(アノード、カソード)の見分け方は、理解できましたか。
7	ブレッドボードの仕組みは理解できましたか。
8	抵抗のカラーコードの読み方は、理解できましたか。
9	ジャンパー線は、理解できましたか。
10	プログラムの入力方法は理解できましたか。
11	コンパイル方法は、理解できましたか。
12	実行方法は、理解できましたか。
13	LED を点滅する仕組みは理解できましたか。
14	1 つの LED の点滅は達成できましたか。
15	2 つの LED の点滅は達成できましたか。
16	3 つの LED の点滅は達成できましたか。
17	コンピュータプログラミング I の知識は、本日の内容に役立ちましたか。
18	今後 Raspberry Pi を購入し、電子工作を行ってみたいと思いますか。

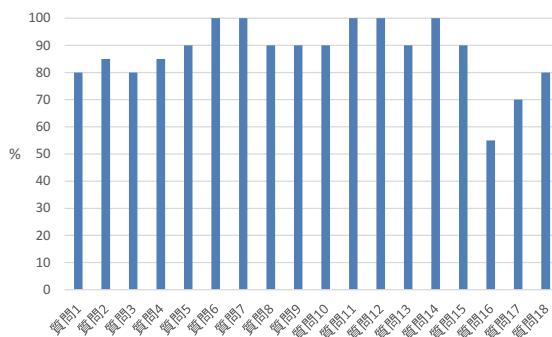


図 10 「はい」の割合

設問 1 の半田付けの経験は、80%が「はい」と

回答した。設問 2 の電子工作的経験は、85%が「はい」と回答した。3 年次生の参加が多かったため、既に演習系の授業で電子工作を体験し、これが高い回答率になった原因と考えられる。設問 3 のラズパイを知っていた人は、80%であった。設問 4 の OS ダウンロードは、体験はしていないものの 85%の理解が得られた。設問 5 の OS インストールは、90%の理解が得られた。設問 8 から 10 のカラーコードの読み方、ジャンパー線、プログラムの入力方法は、90%の理解が得られた。設問 13 の LED の点滅の仕組みは、90%の理解が得られた。

設問 14 から 16 の回答結果から、1 つの LED の点滅は全員が達成、2 つの LED の点滅は 90%が達成、3 つの LED の点滅は約 50%が達成できたことが分かり、作業の個人差が明らかになった。

設問 17 の「コンピュータプログラミング I の知識は、本日の内容に役立ちましたか」は、70%の回答となった。筆者としては 90%程度を期待していたが、これは残念な結果となった。

設問 18 の「今後 Raspberry Pi を購入し、電子工作を行ってみたいと思いますか」の設問に対して、80%が「はい」と回答した。集中講義の大きな成果であると考えている。

4.2 モチベーションの推移の分析

受講者のモチベーションの推移を分析した。モチベーションはプログラミング教育を対象に開発した SIEM(ジーム)アセスメント尺度をアレンジして利用した。この様子を表 8 に示す。重要度は「ハードウェアを学習することは重要だと思いますか」、現状認知度は「ハードウェアの知識・技術は身についていると思いますか」、期待度は「もっとハードウェアの知識や技術を高めたいと思いますか」の設問項目を設け、「1 まったくそう思わない」「2 あまりそう思わない」「3 どちらともいえない」「4 ややそう思う」「5 強くそう思う」の 5 段階のリッカート尺度で測定する。モチベーションは、重要度と期待度の積の平均値を求め、1 から 25 に定量化する。

表 8 モチベーション評価項目

No	設問内容
19	ハードウェアを学習することは重要だと思いますか(重要度)。
20	ハードウェアの知識・技術は身についていると思いますか(現状認知度)。
21	もっとハードウェアの知識や技術を高めたいと思いますか(期待度)。

集中講義を受ける前のモチベーションは 16.5、集中講義を受けた後のモチベーションは 20.8 と 4.3 向上していることが分かった。

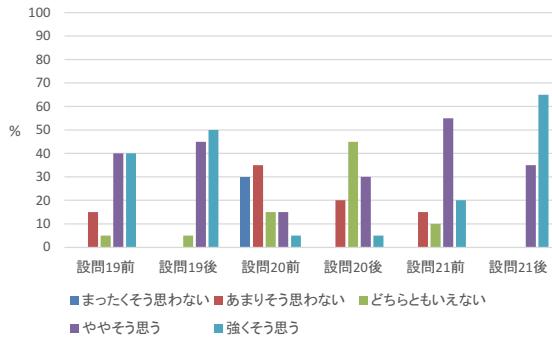


図 11 モチベーション評価項目の受講前後の変化

モチベーション評価項目の集中講義前後の推移は、図 11 に示す。重要度(設問 19 後)は「あまりそう思わない」が無くなった。現状認知度(設問 20 後)は「まったくそう思わない」が無くなかった。期待度(設問 21 後)は「ややそう思う」「強くそう思う」だけになり、「強くそう思う」が 60% を超え、モチベーションの向上に大きな影響を与えていたことが分かった。

4.3 受講者の声

アンケート調査に記載された自由記述の一部を以下に示す。なお自由記述には軽微な誤植等があったため、著者にて一部の文言を修正している。

- ・ RaspberryPi は聞いたことあった程度だったので触ることができて良かった。プログラミングが得意ではないがシンプルで分かりやすく、自分にもできてうれしかった。発展させていけば様々なことに使えるそうなので、卒業研究などでぜひ活用したいと思った。そう思えるきっかけをつくりていただきありがとうございました。第2回第3回とレベルアップしていく講義があったらぜひ参加したい。
- ・ RaspberryPi を全く分からなかったため参加するか迷っていましたが、ゆっくり進めて下さったためついていくことが出来、参加して良かったです。ありがとうございました。LIFI 興味が湧いたので調べてみようと思いました。
- ・ この機会に Raspberry Pi に触れることができてよかったです。セットアップから学ぶことができたので、自主的に触ってみたいと思う。もっと高度なことができるといいなと思った。(疑似的な人感センサーを作成するなど)
- ・ セットアップに時間がかかり、L チカで講義が終わってしまったのがもったいないと感じましたが、楽しかったです。
- ・ ラズパイに興味がありずっと始めたいと思っていたのですが、初めの環境を整えるところで挫折する可能性がある事とお金がかかる事でハードルが高く始められずにいたところ今回の話が出たのでありがたかったです。Processing は、コンパイルする必要がなくインストールするだけで環境も整える必要がないのでプログラミングの本質的な部分に集

中できたので良かったのですが、ほかの言語を独学で始める際、環境を整えるところに時間を取られ挫折してしまうのでぜひ prosessing 以外の言語の入門的な立ち位置の講習会も開催していただきたいです。今日はありがとうございました。

- ・ ラズベリーパイの概要がわかり、知見を広めるのにいい体験となった。事前に C 言語を少しでもさわっておけばもっと活用できたと思う。Java で動かせるハードウェアがあれば知りたい。
- ・ 初めて Raspberry Pi を使ってみてハードウェアを使ったプログラミング自体はやったことがなかったので楽しい講義でした。3つのダイオードを点滅までいかなかったことは心残りではあります。
- ・ 進みがゆっくりで理解できた。
- ・ 丁寧に指導してくださったので、ハードウェアについて無知な自分でもついていくことができ、楽しかった。
- ・ 普段体験できないことができて楽しかった。

集中講義を通じて Raspberry Pi が何なのかは理解できたようである。また、ハードウェアの敷居が高いと感じていた人にとって良い体験になったことがうかがえる。今後はレベルアップした講義への期待も寄せられ、自分でやってみようと思っていることがうかがえる結果となった。

5 今後

5.1 繼続的な開催

今後も、初めて体験する人たちを対象とした集中講義の開催が継続的に必要であると考えている。さらに、これに続く高度な内容につなげる必要がある。以下に動作確認を行った例を紹介する。

5.2 光スイッチ

1つは、光スイッチを使った LED の点灯である。CdS(硫化カドミウム)セルは光が当たると抵抗値が低下する。この性質を利用すると、暗くなったら LED を点灯する光スイッチとして活用できる。この様子を図 12 に示す。ペンのキャップで CdS セルを覆い隠すと、暗くなり LED が点灯する。

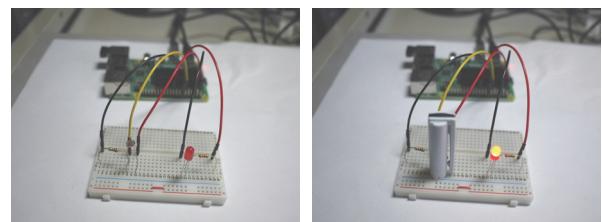


図 12 暗くなったら LED を点灯する例

5.3 サーボモータと RGBLED アレイ

次の例は、サーボモータ(SG90)の回転である。繰返しの仕組みをサーボモータの回転に置き換えることができる。さらに、RGBLED アレイ

(OSX10201-LRPB2)である。1つのセグメントの中にRGBの3色のLEDが入っており、3色の点滅の仕方を工夫すると、3色以外の色に見せかけることができる。セグメントの数が増えると、配線が複雑になる。これらの様子を図13に示す。

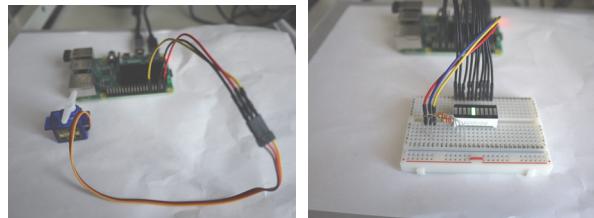


図 13 サーボモータ(左)とRGBLED アレイ(右)

5.4 超音波距離センサー

最後の例は超音波距離センサー(HC-SR04)である。これは2cmから400cmまで測定でき、この様子を図14に示す。LEDの点滅と組み合わせると、距離が接近すると点灯させることも可能となる。

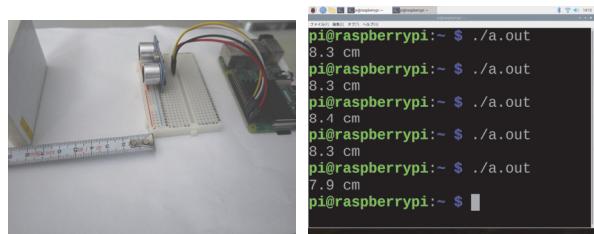


図 14 超音波距離センサーを使った計測の様子

5.5 電子教科書への掲載

集中講義の準備と並行して、「ためしながら学ぶC言語」の書籍化が進行していた。2018年と2019年はプリント本を作成し、「12章 組込み型プログラミングの体験」にArduinoを対象にした内容を掲載していた。2020年の書籍化にともないRaspberry Piの内容に書き換え、先に述べたCdSセル、サーボモータ、RGBLEDアレイ、超音波距離センサーの例を12章に記載した[7]。今回の集中講義では、LEDの点滅の部分を取り上げた。

5.6 マウス、キーボード、ディスプレイの手配

2020年4月から総合メディアセンターのPC教室4は、アクティブラーニング用の教室に改修され、残念ながらマウス、キーボード、ディスプレイが利用できなくなった。この対策として、ノートPCからLANを経由してリモート接続する方法が現実的な方法であると考えている。ただし、OSのインストールはマウス、キーボード、ディスプレイが必須であり、これらの確保が必要である。

6まとめ

ハードウェアに興味はあるものの、一步を踏み出せない受講者にとって、良い切っ掛けになったことが分かった。さらに、受講者のモチベーションは、集中講義の前後を比較すると、集中講義後は4.3の向上があり、その効果を示せた。今後Raspberry Piを購入し、電子工作を行ってみたいと思う人が80%居たことも大きな成果と考えている。

集中講義ではタイミング良くRaspberry Pi 4を入手できたためこれを使ったが、Raspberry Pi 3でも同様の効果が期待できる。最近、Raspberry Pi 4は主記憶が8GBのものも販売されるようになり、今後はサーバとしての活用も期待できそうである。

その他、集中講義を実施して受講者の様子から分かったことは、GUIに慣れている世代であり、Unixのコマンドをほとんど知らないことである。他の学科でC言語の授業を担当しているが、gccを使うために必要なコマンド程度しか紹介していない。今後は、コマンドの教育もバランスよく取り入れていくことが必要である。

最後に、集中講義の開催によって図1に示したイメージはほぼ実現できたと考えている。コストを考えるとノートPCを持たせるBYODからBring Your Own microSDへの可能性もありそうである。

集中講義の開催に当たり、総合メディアセンターのご支援をいただいたことに感謝いたします。

参考文献

- [1] Raspberry Pi, <https://www.raspberrypi.org/>, 2020年10月2日閲覧
- [2] 土肥紳一, Raspberry Pi 3を活用した情報教育の可能性, AXIES, FD1-2, pp.155-162, 2019.
- [3] 土肥紳一, プログラミング入門教育を教わった後のArduinoの活用効果について, AXIES, F2E-4, p1-p8, 2014.
- [4] Arduino, <https://www.arduino.cc/>, 2020年10月2日閲覧
- [5] 加藤 芳夫, たのしくできる Raspberry Pi とブレッドボードで電子工作, 東京電機大学出版局, 2016.
- [6] コンピュータプログラミングIを教わった後のハードウェア入門 (Raspberry Pi編), <https://dohi.chiba.dendai.ac.jp/~dohi/ext-raspberrypi/>, 2020年10月2日閲覧
- [7] 土肥 紳一他, ためしながら学ぶC言語(電子教科書), 東京電機大学出版局, <https://coop-ebook.jp/asp>ShowSeriesDetail.do?seriesId=MBJS-382324>, 2020年10月2日閲覧