

IoTを用いた多機能な消毒装置の開発

情報班: 梶 圭介、地主 恭悟、中西 翔哉、川口 大翔、
山澤 一颯、前畑 拓、金 度晃

要約

本研究では、COVID-19の感染拡大により使用頻度が格段に増えた消毒をより意味のあるものにするために、追加機能をつけることにより多機能かつ効率的な消毒装置を作った。消毒時にログの送信をすることはできたが、体温を測ることはできなかった。

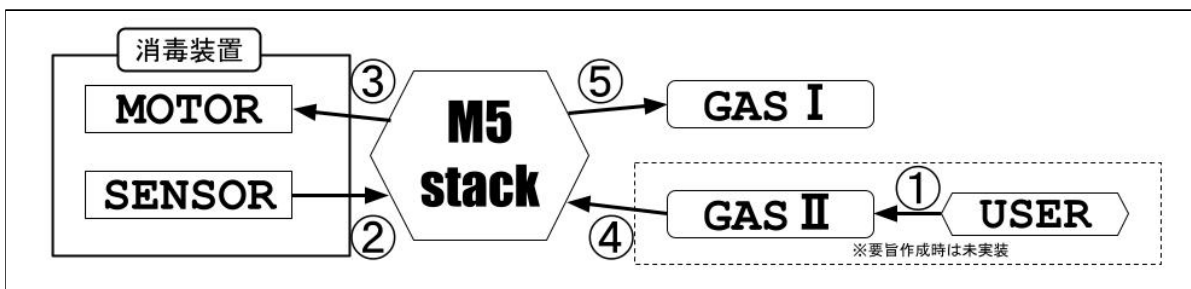
1. はじめに

昨今の新型コロナウイルスの感染拡大に伴い、様々な場面で消毒をすることが必須となっている。本研究では、消毒の時間を効率的に使うために、様々な機能を搭載した消毒装置があればよいと考えた。そこで、学校での使用を想定した、体温測定や消毒ログの記録などの機能を搭載した消毒装置の開発を行った。

2. 研究手法

(1)～(3)の部門に分かれ、以下のようなシステムの開発を目指した(図1)。

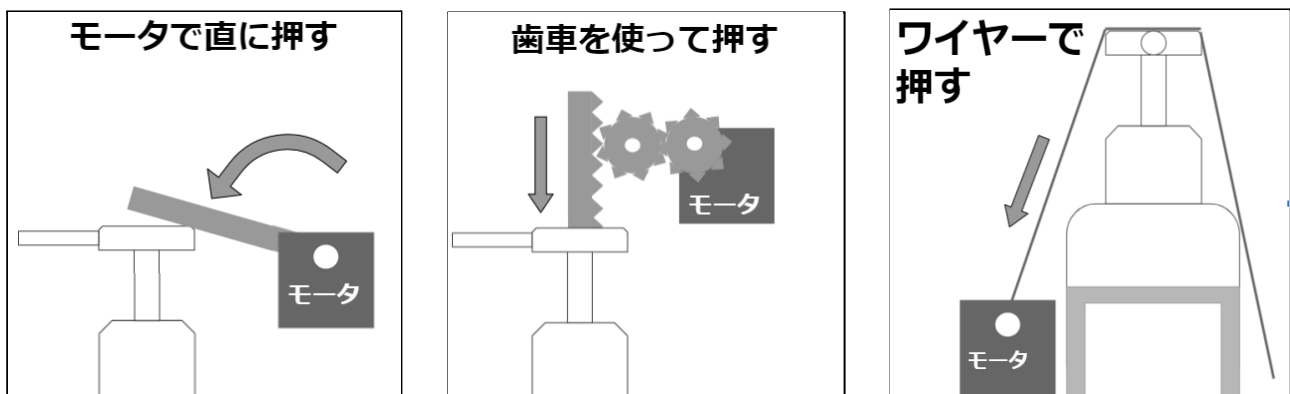
- ① 使用者が各自のスマホでを使って、IDを入力し、それをGoogle Apps Script(以下GAS)Ⅱに送信する。
- ② 使用者が手をかざしたことをセンサーで読み取り、M5stackに伝える。
- ③ M5stackからモータに信号を送り、消毒装置を作用させる。
- ④ ③と同時にM5stackがGASⅡからのIDを読み取る。
- ⑤ M5stackがGASⅠにIDなどのログを残す。



(図1:マイコンを中心としたシステム図)

(1)本体の作成

まず、本体の作成にあたり消毒液を噴霧する際に、消毒液上部をどのように押すのか検討した。モータで直に押す、ギヤを使って間接的に押す、ワイヤーを用いて押す、の3つの案から検討し、実現性の観点から今回はワイヤーを用いる方法を採用した(図2)。



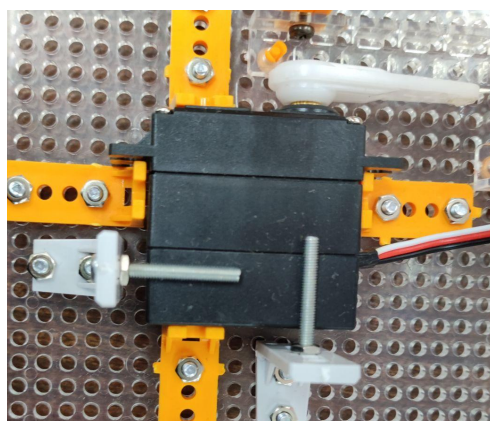
(図2)

<用いたもの>

- ・市販の工作キット(タミヤ 楽しい工作シリーズ)
〔透明ユニバーサルプレートL / 透明ユニバーサルプレート / ユニバーサルアームセット /
ロングユニバーサルアームセット / ラック&ピニオン ギヤセット / 工作ギヤセット / プーリーセット〕
- ・マイコン(M5Stack) (図3-1)
- ・サーボモータ(GWS サーボ SO3T/2BBMG/F) (図3-2)
- ・赤外線センサ(図3-3)
- ・体温センサ(M5Stack用ミニサーマルカメラキット MLX90640)



(図3-1)



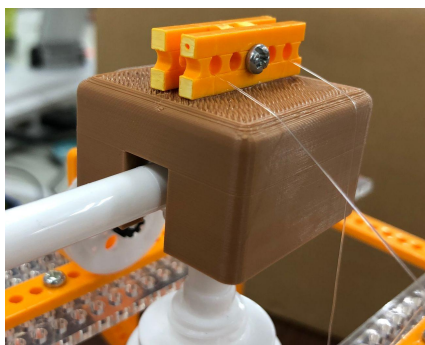
(図3-2)



(図3-3)

<作成手順>

- ① 工作キットを用いて本体を作成した。
- ② モータ、センサを取り付けた。
- ③ 消毒液上部の押す部分とワイヤーの接続のために、3Dプリンタで追加パーツを作成した(図4)。



(図4)

(2)マイコンとの連携

マイコンにはM5stackを用いた。UIFlowを用いてblocklyでM5stackからモータ、センサへの命令を行い、以下の目標の実現を図った。

<目標>

- ① 赤外線センサで手が近づくと、モータを動かす。
- ② 体温センサから使用者の体温を測定し、マイコンの小型モニターに表示する。
- ③ 消毒記録、温度データなどをGASに送信する。

<手順>

まずセンサーが手を感知したときのみプログラムが作用するようにし、ログをGoogle Spread Sheetに送り、消毒液の噴出が行われるようにモーターの角度を変更し、使用後に「THANK YOU」とM5stackのモニター上に表示されるようなブロックの配置を考えた(図5)。



(図5: UIFlow上のblockly)

(3)クラウドとの連携

M5Stackとクラウドとの連携にはGAS (Google Apps Script) を用いた。GASとはGoogle 社が提供するクラウド上のソフトウェア実行プラットフォームであり、プログラミング言語には JavaScript を改良した ECMAScript を用いる。

<目標>

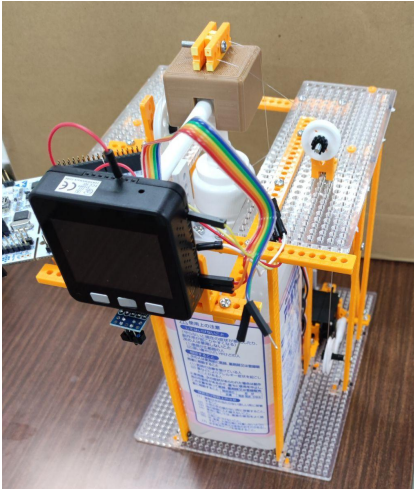
- ① M5stackから各個人の個人番号、体温や消毒した時間のデータをスプレッドシート上に表示する。
- ② 体温が一定の値を上回った時に赤くスプレッドシート上に表示する。
- ③ ホームページから個人の認証を可能にする。

3. 結果

本体を作成し、赤外線センサが反応するとサーボモータが動き、消毒液の噴出するシステムをつくることができた(図6)。しかし、体温センサとM5stackの接続が出来ず、体温データの取得はできなかった。

GASを用いて、M5stackから受け取った、消毒した日付、時間をGoogleスプレッドシートに表示することができた。また、実際に使用はできなかったが、体温が37.5度以上の場合スプレッドシート上で赤く表示し、指定したアドレスにメールを送るプログラムを作成することができた。

(1) 本体作成の結果



(図6)

(2) マイコン(M5stack)との連携結果

赤外線センサが物体の存在を感知すると、マイコンの小型モニターの色を変化させ、使用者が認知できるようにした。また、マイコンとモータを連携することにより手をかざした際にモータのアームが任意の角度まで動き、適量の消毒液を出すようにした。センサが反応してからモータが動くまでに一定時間のラグが見られた。

体温測定は、用意した体温センサが既に用いていたUIFlowのVer.では使用することができず、研究時間の制限もあり実現することができなかった。

(3) クラウドとの連携結果

GASでコードを作成し、M5stackとコードとの連携には成功した。(図4)消毒装置が作動すると、消毒した日付と時間をスプレッドシート上に記録することができた。すなわち、出席確認をすることができた。体温が測定できなかったため、代わりに一定の範囲内で無作為に数値が代入されるランダム関数を用いた。体温が37.5℃以上だった場合、スプレッドシート上で背景を赤くすることに成功した。(図5)また、この場合、万が一に備えて特定の教職員にGmailで通達できるようにできた。ホームページの作成はできたが、個人を特定してスプレッドシート上に個人番号のデータを送るといったホームページとクラウドとの連携ができなかった。そのため、個人番号も体温と同様に、代わりにランダム関数を用いた。

```
1 function doGet(e) {
2   //SpreadsheetApp.getActiveSheet().appendRow([new Date(), e.parameters["id"],e.parameters["tempra"]]);
3   Logger.log(e);
4   const tempra = Number(e.parameter["tempra"]);
5   const s = SpreadsheetApp
6     .getActiveSheet()
7     //idは個人番号 例)高津高校2年8組22番の場合 2822
8     //tempraは体温
9     .appendRow([new Date(), e.parameter["id"], tempra]);
10  const row = s.getDataRange().getLastRow();
11  if (tempra >= 37.5) {
12    s.getRange(row,1).setBackground("red");
13    s.getRange(row,2).setBackground("red");
14    s.getRange(row,3).setBackground("red");
15    MailApp.sendEmail({
16      to: "メールアドレス",
17      subject: "高熱者来校!",
18      body: "高熱の人が来ました。詳細はこちら! https://docs.google.com/spreadsheets/d/1eBiSYgqQVsEYrgK13ZpJGgBX-P0xGKYA8SdwsMRGBNM/edit#gid=0"
19    });
20  };
21 }
22 return HtmlService.createTemplateFromFile("result").evaluate();
23 }
```

(図7: GASで作成したコード)

1	日付、時間	ID	体温
2	2022/01/26 15:18:42	1313	38.1
3	2022/01/26 15:18:50	2509	38.8
4	2022/01/26 15:18:57	3214	38.6
5	2022/01/26 15:19:20	2237	36.4
6	2022/01/26 15:19:59	3624	36.5
7	2022/01/26 15:20:06	1828	38.4
8	2022/01/26 15:20:14	1602	36.4
9	2022/01/26 15:36:22	1731	37.9
10	2022/01/26 15:36:49	1904	35.3
11	2022/01/26 15:36:58	3219	36.7

(図8:データを受信したスプレッドシート) ※IDは個人番号のこと

4. 考察

赤外線センサからUIFlowを通してサーボモータを動かし消毒を行い、ログをとるところまでは実現することができた。しかし、研究時間の関係から、個人情報入力や実用化はすることはできなかった。そのため今後の展望としては消毒時の出席確認のシステム構築や、実用化が挙げられる。

5. 結論

昨今の新型コロナウイルスの感染拡大から、今や当然となっている消毒の時間の効率化のために、基本となる非接触の消毒器の開発とともに、体温測定や消毒のログ記録、消毒時の出席確認を目指し本体作成からプログラミングまで研究を行った。非接触消毒器の作成やログ記録などは実現完了したが、出席確認の一般化や電気消費効率の改善など、実現が間に合わなかったことも多い。

6. 参考文献ならびに参考Webページ

<https://blogdaichan.hatenablog.com/entry/2020/06/03/194105> [ブログだいちゃん, 2020]

<https://contents.zaikostore.com/semiconductor/5994/> [Core Contents,2021]

7. 謝辞

本研究を進めるにあたり、大阪工業大学の小林裕之教授にご指導いただきました。お礼を申し上げます。