

# 換気を促すための二酸化炭素濃度表示装置の試作

曾根直人\*

COVID-19 ウィルスが世界的に流行しており、国内でも感染例の報告が増えている。感染を防ぐためのワクチンもまだ十分に流通しておらず、予防には手指消毒やマスクの着用が勧められている。COVID-19 の感染は飛沫を吸い込むことにより発生すると言われており、部屋の換気が重要になる。しかし大学の教室では、空調が行なわれていることや授業中の音を防ぐためにも常に窓を開放しての換気は行われていないことが多く、換気が気になった人が適宜窓を開けるという対応が行なわれている。

そこで、より合理的な換気を行うために、部屋の CO2 濃度を計測し表示するシステムを開発し運用した。客観的な指標を表示することで、適切な換気を促すことが可能になると考えている。

[キーワード: COVID-19 対策, CO2, IoT, Raspberry Pi, MH-Z19]

## 1. はじめに

COVID-19 ウィルス感染対策の一つとして、密閉空間を避けるために換気の徹底が必要とされている。しかし、適切な換気が実施できているかを判断することは難しく、文部科学省の「学校における新型コロナウイルス感染症に関する衛生管理マニュアル～「学校の新しい生活様式」～(2020. 12. 3 Ver. 5)」[1]では、二酸化炭素(CO2)濃度の計測による客観的な指標の利用が示されている。

ハ) 機器による二酸化炭素濃度の計測十分な換気ができているか心配な場合には、換気の指標として、学校薬剤師の支援を得つつ、CO2 モニターにより二酸化炭素濃度を計測することが可能です。学校環境衛生基準では、1500ppm を基準としています。政府の新型コロナウイルス感染症対策分科会では、マスクを伴わない飲食を前提としている飲食店等の場合には、1000ppm 以下が望ましいとされており、昼食時には換気を強化するなど、児童生徒の活動の態様に応じた換気をしてください。

そこで、本研究では、換気の目安となる CO2 濃度を計測し、利用者へ換気を促すフィードバックを行う機能や実際に部屋の利用により、どの程度 CO2 濃度が変化するかを時系列グラフで表示するためのシステム(図 1)を構築した。

## 2. 二酸化炭素濃度表示システム

CO2 濃度を計測し表示するシステムの開発を迅速に行なうため、本研究では既存の機材を利用した。そのため、異なる機材やセンサの組み合わせによる複数のシステムを構築した。ここではそれぞれのシステムについて詳細を述べる。

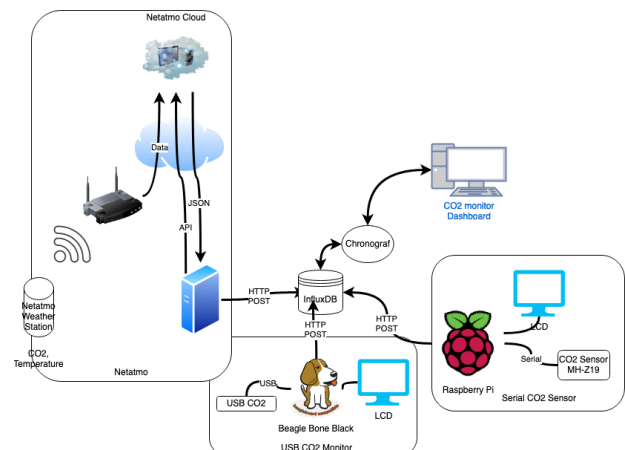


図 1 二酸化炭素濃度表示システム

### 2.1. Netatmo

Netatmo ウェザーステーション(図 2)は市販の環境観測機器で気温、湿度、気圧、騒音、CO2 濃度の計測が可能である。計測した結果はクラウドサービスへ送信されており、PC やスマートフォンから測定値の確認やグラフを参照できるようになっている。クラウドサービスには API [2]が用意されており、計測した値を独自に処理することも可能である。

Netatmo を利用すれば簡単に CO2 濃度の計測が可能

\* 鳴門教育大学大学院 高度学校教育実践専攻 自然・生活系教科実践高度化コース(技術・工業・情報科教育実践分野)



図 2 Netatmo ウェザーステーション

であるが、複数の部屋を監視しようとするコストが高くなる。また Wi-Fi 接続が必要になるが、Web 認証や 802.1x 認証は利用できないため、情報基盤センターの提供する Wi-Fi では利用できない。そのため専用の AP を用意する必要があり、導入は 1 台のみとした。また後述する他システムと同じように測定値を扱うため、API を利用して測定値を取り出し、時系列 DB へ格納するようにした。

Netatmo の API を利用するには、

1. Client ID, Client Secret の取得(初回)
  2. access\_token の取得(180 分毎)
  3. access\_token を利用した API 呼出(10 分毎)
- の手順が必要になる。Web ブラウザで 1. の情報を取得し、2., 3. の処理は shell script で記述 crontab により定期的に行っている。API で呼び出した値は、shell script により influxDB へ格納している。3. の API 呼出は curl コマンドで以下のように行ない、

```
curl -s -proxy ${PROXY} -X GET \
https://api.netatmo.com/api/getstationsdata?access_token=${TOKEN}
```

JSON データを得る。得られた JSON は jq コマンドを利用してパースし、CO<sub>2</sub> 濃度、室温の情報を取り出している。

## 2.2. USB CO<sub>2</sub> モニター

USB 接続して利用する CO<sub>2</sub> 計測器 CO<sub>2</sub>-mini (図 3) が市販されている。本体の LCD に室温と CO<sub>2</sub> 濃度を表示する機能を有している。さらに非公式な利用ではあるが USB インタフェースから計測値を読み取ることも可能である。この計測器を扱うためのライブラリも Ruby や Python では公開されている。Python の

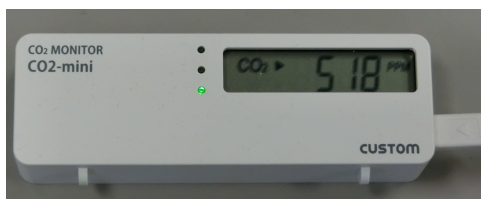


図 3 USB CO<sub>2</sub> モニター

ライブラリを活用した CO<sub>2</sub> 監視ソフトウェア co2meter[3] が github 上で公開されている。co2meter は測定値をファイルへ出力するだけでなく flask Web サーバを介して、CO<sub>2</sub> の状況を Web ブラウザへ表示することもできる。しかし今回の研究では、設置場所の問題からできるだけシステムを小型化したい希望があったため、パソコンではなく Single Board Computer (SBC) を採用したため、より表示に負荷の少ないソフトウェアを作成した。

開発したシステムでは SBC の一種である Beagle Bone Black (BBB) を用いて作成した。BBB は Raspberry Pi と同じく Linux をインストールし、co2meter をインストール、実行する。このままでも Web ブラウザから接続すれば、CO<sub>2</sub> 濃度を確認できるが、BBB 本体で Web ブラウザを実行し、表示するには CPU パワーが不足しているため難しい。より負荷を減らした可視化を行なう必要がある。そこで co2meter で計測した値を用いて python スクリプトにより、PNG ファイル(図 4)を生成し、fbi コマンドによりフレームバッファへ表示するようにした。これにより Web ブラウザも不要かつフレームバッファへ PNG を出力するだけですむため X Window System も不要となった。また、端末室は夜間には施錠しており、無人になるためディスプレイへの表示は無駄になる。節電のために /sys/class/graphics/fb0/blank へ crontab から値を書き込み夜間は HDMI の出力を off にしている。

このシステムも Netatmo システムと同様に計測値を時系列 DB である influxDB へ定期的格納している。

PNG で生成する画像には co2meter で利用している画像をそのまま流用した。この画像は 800ppm 未満ではグリーンアイコン、800ppm 以上 1200ppm 未満ではイエローアイコン、1200ppm 以上ではレッドのアイコンとなるように設定した。

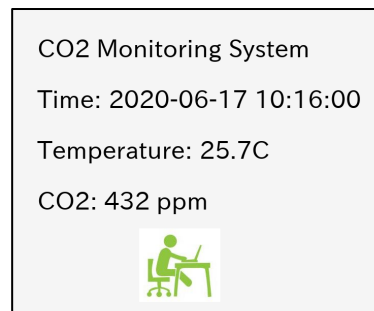


図 4 CO<sub>2</sub> 濃度表示画面

### 2.3. シリアル接続 CO2 センサ

安価な CO2 センサに MH-Z19 がある。このセンサは、シリアルポート(ピンヘッダ)接続で CO2 濃度の測定値を得ることができる。BBB や Raspberry Pi などピンヘッダのシリアルインターフェイスを持つ SBC であれば、Tx と Rx および電源をジャンパー線で接続すれば利用できる(図 5)。python には MH-Z19 を扱うためのモジュールが提供されており、

```
pip install mh-z19
```

により導入できる。このモジュールを利用することで、CO2 濃度や室温の測定値を json 形式で取得できる。json 形式で取得した測定値を jq コマンドによりパースし、csv 形式に変更してファイルへ保存している。以下に測定値を保存するための shell script を示す。

```
#!/bin/bash
date=`date --rfc-3339=seconds`
csv = `python3 -m mh_z19 --all| \
jq -r '[.co2,.temperature]| @csv'|`
echo $date,$csv
```

この script を実行することで、次のような出力を得られるので、ログファイルへ直接保存している。

```
2021-03-21 15:28:01+09:00,505,18
```

influxDB への保存は、crontab を利用して上記フォーマットのログファイルから最後尾に記録されている測定値を取り出し、influxDB へ格納するための LINE 形式[4]へ変更する。変換されたデータは curl コマンドを使って influxDB へ POST し、データベースへ格納している。

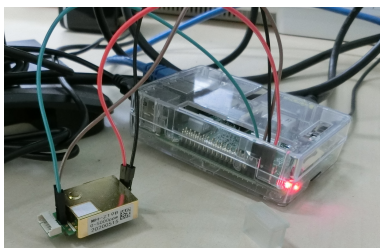


図 5 CO2 センサ MH-Z19 と Raspberry Pi2

### 3. CO2 監視ダッシュボード

CO2 の測定値は、全て時系列データベース influxDB へ格納している。influxDB に格納することで、可視化ツール Chronograf を利用でき、CO2 濃度の変化データを示すグラフや直近の CO2 濃度を計測器のアナログメータのように表示する gauge を使った CO2 濃度監視ダッシュボード(図 6)を作成した。

現在 CO2 を計測しているのは、

1. 教育用端末室(Netatmo)
2. マルチメディア教育実習室(USB CO2+BBB)
3. 人文棟特殊端末室(MH-Z19+RaspberryPi 2)
4. 自然棟特殊端末室(MH-Z19+RaspberryPi)

の 4 箇所であり、ダッシュボード上では、4 つの部屋の CO2 濃度、室温の時系列グラフ、直近の測定値を表示するメータ及び本部棟の屋上に設置している太陽光発電設備が測定している外気温と発電量を時系列のグラフとして表示している。

CO2 濃度のグラフ(図 7)に注目すると、換気が不十分な部屋で人が活動することで濃度が上昇することが確認できる。また、窓を開けるなど換気を実施することで、CO2 の濃度を下げられることも確認できる。

CO2 監視ダッシュボードは Web ブラウザでアクセスし、目視で監視する必要があるが、常に監視することは困難である。そこで Chronograf の Alerting 機能を利用し、CO2 濃度が設定したしきい値を越えた場合に警告メールが届くようにした。具体的には、直近 15 分の CO2 濃度の平均が 1000ppm を越えた場合には指定したメールアドレスへメールが届くようにした。メールを受信した場合には、部屋の窓を開けるなど換気を促すことを予定している。



図 6 CO2 濃度監視ダッシュボード

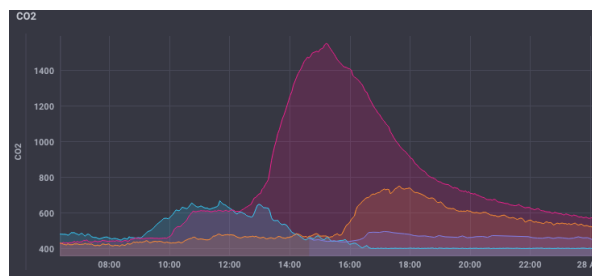


図 7 CO2 濃度グラフ

### 4. まとめ

部屋の換気を促すために、部屋の CO2 濃度を測定し表示するシステムを構築した。既存の機材も活用したため、複数の測定機器から構成されるシステム

となったが、測定値は全て influxDB に格納することにより、統一のダッシュボードで結果を確認できるようになった。

本来なら部屋の利用者が測定値の表示を見て自主的に換気を行い、十分に換気ができれば窓を閉めるといった行動を取って欲しいと考えているが、現時点ではシステムの周知が不十分なためかディスプレイの表示を参考に利用者が窓を開けて換気を行うといった行動の変化を見ることはできていない。行動を変えるためには、さらなる利用者へのフィードバックが必要ではないかと推測している。例えば SBC で表示する画面にもリアルタイムの CO2 濃度だけでなく、時系列のグラフを表示することを検討している。時系列の CO2 濃度を表示することで、自身が窓を開けるといった行動により、CO2 の濃度がどの程度下がるのかをより具体的に把握することができると、利用者へ提供する価値があるのではないかと考えている。

部屋の換気を行うための目安として CO2 濃度の測定は有効であると考えられ、市販の CO2 測定器も増えてきた。しかし、リアルタイムの測定値だけでなく、時系列のグラフを表示したい場合は今回のよう

なシステムを作成する必要がある。SBC と MH-Z19 センサの組み合わせであれば比較的 low 価格で構築することが可能であり、学校などでの大量導入も可能ではないかと考える。

## 参考文献

- [1] 文部科学省(2020) 新型コロナウイルス感染症に関する衛生管理マニュアル～「学校の新しい生活様式」～(2020.12.3 Ver.5), [https://www.mext.go.jp/content/20201203-mxt\\_kouhou01-000004520\\_01.pdf](https://www.mext.go.jp/content/20201203-mxt_kouhou01-000004520_01.pdf) (最終アクセス日:2021年1月7日)
- [2] NETATMO, Weather API Documentation, <https://dev.netatmo.com/apidocumentation/weather> (最終アクセス日:2021年1月7日)
- [3] co2meter, <https://github.com/vfilimonov/co2meter> (最終アクセス日:2021年1月7日)
- [4] InfluxDB line protocol tutorial, [https://docs.influxdata.com/influxdb/v1.8/write\\_protocols/line\\_protocol\\_tutorial/](https://docs.influxdata.com/influxdb/v1.8/write_protocols/line_protocol_tutorial/) (最終アクセス日:2021年1月7日)