

# アクアポニックスによる植物生産工程の自動化

—リビングに置ける生態系をハック可能に—

## 1. 背景

歴史上、自然や生態系は人間にとって制御不可能なものであった。とりわけ農業においては、「米」という漢字は「八十八」を組み合わせてできたものであり、これには米の生育に88もの手間がかかるという意味が込められているものであるが、実際これだけ人間が手間暇をかけたとしても育たないこともある。技術が進歩した現在でも農業は定量化できない、神頼みのものであるという意識は未だ根強い。

## 2. 目的

本プロジェクトでは、生態系の育成環境を人間の恣意的制御に落とし込み、農業や自然は神頼みで、人間の手では自然の力には遠く及ばない、逆らえない、というイメージを払拭し、自分たちの手で完全にコントロール可能、ハック可能なものであるというイメージや体験の普及を目指した。

そのため本プロジェクトでは、アクアポニックスと呼ばれる栽培手法を利用した、生態系のパラメータを人間が手軽に、そして自由自在に干渉できるようにするデバイスとインタフェースを開発することにより、人間が“生態系をハックする”という、新しい概念とそのシステム“Hackaponics”を実現することを目的とした。

## 3. 開発の内容

アクアポニックスとは、魚、植物、微生物のみで閉鎖的な生態系を形成させる、地球のエコシステムをミニマム化したような栽培手法である。魚は生きていく中で排泄物としてアンモニアを出す。これは魚にとって有害であるが、このアンモニアは微生物によって硝酸塩や亜硝酸塩へ分解する。この硝酸塩は植物が吸収し生育の助けとなる。そして、これらのアンモニアや硝酸塩が吸収され魚には綺麗にフィルタリングされた水が戻る。この循環がアクアポニックスの核となる窒素循環となり、物質はこの生態系内で基本的に循環し続ける。この系の中での唯一の外部からのインプットは魚の餌であり、それがこの経路で循環して植物に届き、植物をアウトプットとして収穫するという事で均衡が保たれる仕組みになっている。

本プロジェクトではこのアクアポニックスの栽培手法をベースとした、外部から隔離されたミニマムな生態系を形成するハードウェアと、それを制御、ハックするためのインタフェースであるソフトウェアを開発した。ハードウェアは魚、植物、微生物を育成するための重要なパラメータである気温、水温、栄養等を調節するためのセンサやアクチュエータを備えつつも、六畳間程度のリビングでも設置ができるコンパクトさであるデバイスを開発した。ソフトウェアでは、ユーザがWebやモバイルアプリからそれらのパラメータの値を指定するだけで、センサやアクチュエータ

が適切に制御され、その環境を作り出すインタフェースと制御系、及び詳細な制御を可能にするためのSDKを開発した。

### 3.1. ハードウェア

ハードウェアの外観を図1に、その構造を表した模式図を図2に示す。本ハードウェアは閉じた系となっており、上に植物を育てる空間、下に魚を育てる水槽を配置し、両者はポンプで接続され、アクアポニックスに於ける水の循環を実現している。本ハードウェアは温度、湿度、水温、電気伝導率、pH等の環境値を取得するためのセンサと、それらのパラメータに干渉するためのヒーター、クーラー等の装置やpH調整剤、魚の餌等の物質を投入するための装置であるアクチュエータを備えており（図3）、これらのセンサ・アクチュエータはArduinoとRaspberry Piに接続されて管理、制御されている。制御系と後述するインタフェースとは、Amazon Web Servicesの一つであるAWS IoTを介して通信を行っている（図4）。

ユーザはこのハードウェアに種もしくはは苗と、魚をセットするだけで、制御は全て後述のインタフェースを介して行う。



図1: ハードウェア外観

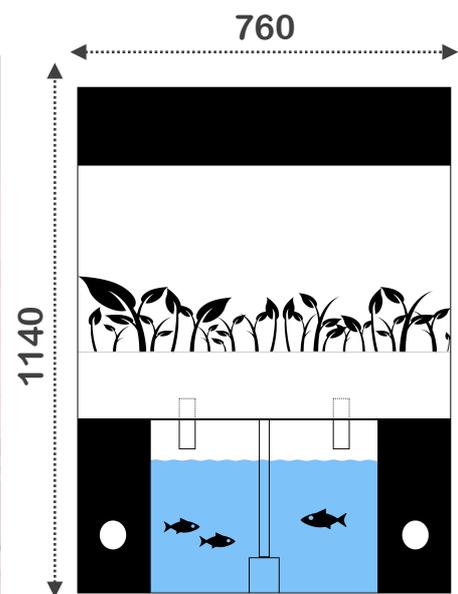


図2: 筐体の模式図（単位はmm）

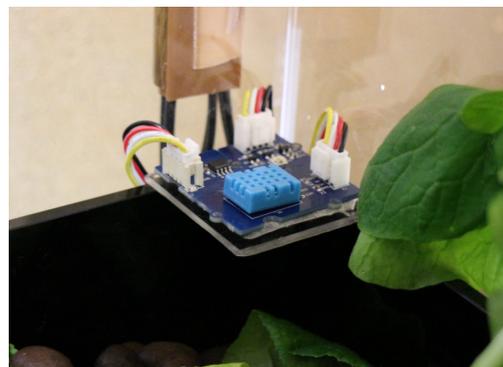


図3: センサ/アクチュエータの例

左：湿度、気温、照度を取得するセンサ/右：植物部のヒーター、クーラー及びLED



### 3.2. ソフトウェア

ユーザは図5に示すWebインタフェース（UI）から生態系のパラメーターを設定することができる。このUIからは各パラメーターの一日単位の値を設定することが可能だが、例えば一日の中で温度差を設定するようなより細かい単位の制御をする場合には、開発したHackaponics SDKを使用して生態系をハックするコードを記述する（図6）。ハードウェアの様子は、図7に示すUIを通して、インターネットに接続していさえすれば任意の場所で閲覧可能である。

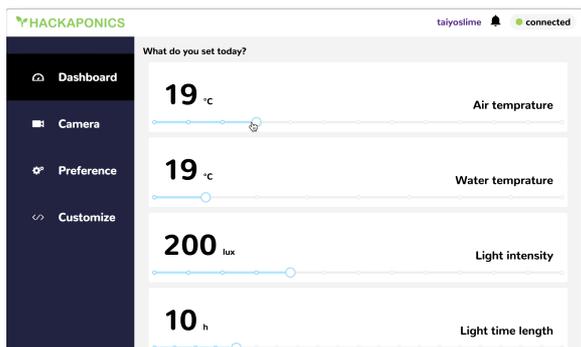


図5: 生態系のパラメーターを設定するためのWebインタフェース

```
const payload = {
  state: {
    actuator: {
      air_temperature: [
        17, 18, 18, 19, 20, 20, 22, 20, 19, 18, 18, 17, 17
      ]
    }
  }
};
```

図6: Hackaponics SDKの使用例  
1日の中で気温の変化を作り出すコード

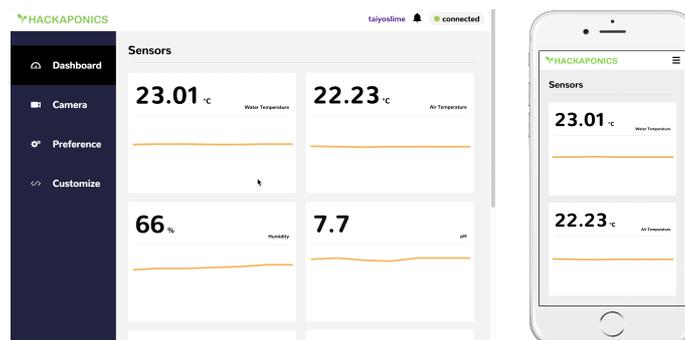


図7: ハードウェアの様子を閲覧するUI  
Webブラウザだけでなくモバイルアプリにも対応

さらに、ハードウェア内の生態系が栽培開始から終了まで、パラメータをどのように制御し、環境値がどう変化したかの一連の制御記録データを、別のハードウェアに適用できるようにフォーマット化し、それをユーザ間で共有できるプラットフォーム

フォームを構築した。Hackaponicsではその制御記録データをレシピと呼ぶ。つまりHackaponicsはこのレシピを入力とすることにより、一度実行した生態系を簡単に再現することが可能となっている。これにより例えば「制御することすら面倒なので、全て自動で運用してほしい」というユーザの要求があったとしても、ユーザコミュニティで共有されたある植物、魚で上手く育った「レシピ」を適用するだけでそれに応えることが可能となる（図8）。

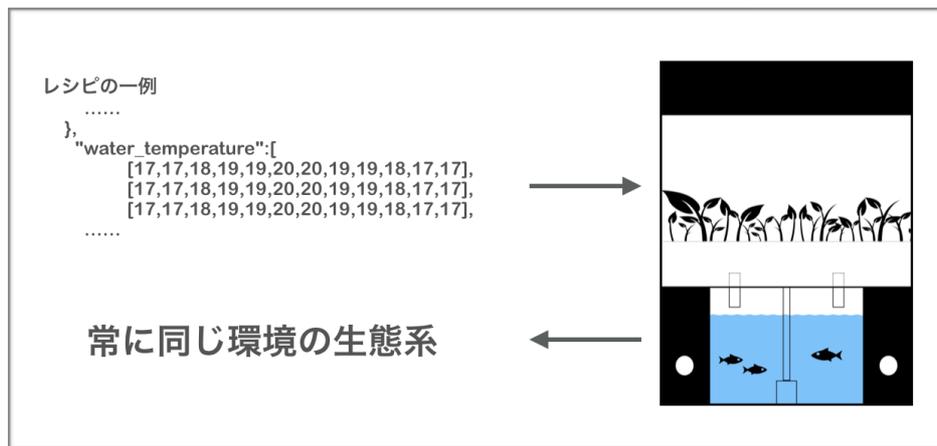


図8: レシピシステム

本プロジェクトでは、以上の開発したHackaponicsを使用して、グッピー及びいくつかのハーブの育成に成功した（図9）。

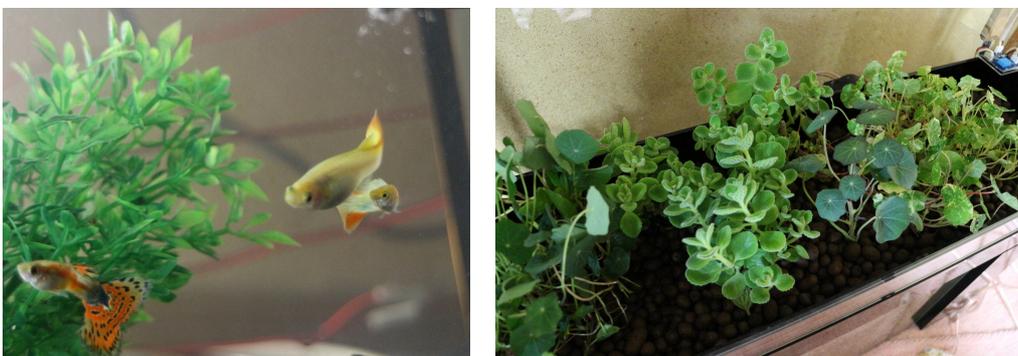


図9: 育成に成功したグッピー（左）とハーブ（右）  
ハーブは主にナスチウムとアロマティカスと呼ばれる品種

#### 4. 従来の技術（または機能）との相違

アクアポニックスを利用した製品はいくつか存在するが、これらはユーザが制御できる要素がそもそも存在せず、魚の餌をあげれば後は勝手に循環していくだけのものが殆どである。こういった製品は「手間がかからない」ことを売りにしているが、実際にいくつか入手し運用してみた結果、栄養不足に陥ったり、pH値の問題が発生したり、不適切な温度管理をされたりなどの要因で、その生態系が崩壊しやすいという問題が発生した。これらはHackaponicsではユーザが適切にハックをすることで制御、解決できる問題である。

研究的な競合システムもいくつかあるが、それらは研究施設内での運用を想定しており、ハードウェアが大きすぎる上にデザインが無機質であることが多いため、Hackaponicsで実現したリビングに置ける手軽さは持ち合わせていない。

## 5. 期待される効果

Hackaponicsを通して、身近なリビングに我々が存在している世界とは異なる生態系が存在し、かつその生態系をハックできるという新しいユーザ体験を提供できるだけでなく、生態系の行く末を握っている緊張感をユーザに与えることによる、従来のアクアポニックスシステムでは生み出せないより高次元の教育効果も期待できる。また、教育への応用については、対象は生態系に関するそれに留まらず、Hackaponics SDKを使用することにより、プログラミング教育をも対象とし、その具体的なアウトプットの一例、とりわけIoTシステムの具体例として活用されることも期待できる。

## 6. 普及（または活用）の見通し

今後も動作テストと改良を繰り返していき、Hackaponicsという概念を表す実用的なデバイスとして家庭や教育現場への普及、およびそのためのプロモーション活動を図っていく。また、Hackaponicsで蓄積された生態系の制御データであるレシピを活用し、農業・水産業の発展に寄与することを目指すことも進めていく。

## 7. クリエータ名（所属）

水橋 大瑠（松本秀峰中等教育学校）

長谷川 陸央（フリーランス）