A low-cost, open source, self-contained bacterial EVolutionary biorEactor (EVE)

Vishhvaan Gopalakrishnan[1,2,*], Nikhil P. Krishnan[2], Erin McClure[3,+], Julia Pelesko[4], Dena Crozier[1,2], Drew F.K. Williamson[3,†], Nathan Webster[3], Daniel Ecker[5], Daniel Nichol[6], and Jacob G, Scott[3,4,7,8,*]

[1]Lerner College of Medicine, Cleveland Clinic, Cleveland, Ohio, United States of America
[2]Case Western Reserve University School of Medicine, Cleveland, Ohio, United States of America
[3]Department of Translational Hematology and Oncology Research, Cleveland Clinic, Cleveland, Ohio, United States of America
[4]Department of Physics, Case Western Reserve University, Cleveland, Ohio, United States of America
[5]Hawken School, Gates Mills, Ohio, United States of America
[6]Centre for Evolution and Cancer, The Institute of Cancer Research, London, United Kingdom
[7]Department of Radiation Oncology, Cleveland Clinic, Cleveland, Ohio, United States of America
[8]Center for Proteomics and Bioinformatics, Case Western Reserve University School of Medicine, Cleveland, Ohio, United States of America
[*]corresponding.author@email.example
[+]Current Address: University of South Florida Morsani School of Medicine, Tampa, Florida, United States of America
[†]Current Address: Department of Pathology, Brigham and Women's Hospital, Boston, Massachusetts, United States of America

ABSTRACT

Recently, a concerted effort has been made to study the evolution of drug resistance in organisms at increasingly smaller time scales and in a high-throughput manner. One effective approach is through the use of a bioreactor, such as a morbidostat – a device that can continuously culture bacteria and monitor this growth in real time. These devices can be technically challenging and expensive to implement for scientists, let alone students or teachers who seek an innovative and intuitive way of studying evolution. We aim to provide a flexible and open source automated continuous culture device framework to study biological concepts such as population dynamics and evolution; a framework that is capable of replicating morbidostat, chemostat, and turbidostat functionality. Within the educational environment, our goal is to foster interaction and interest between the engineering and biological fields by allowing teachers and students to build their own systems and design experiments on the proposed open framework. We present a continuous culture device designed for bacterial culture that is easily and inexpensively constructed, lends itself to evolution experiments, and can be used both in the academic and educational environments.

Introduction

Continuous microbial culture is a powerful tool for observing and directing evolution in both research and industrial settings. However, commercially available continuous culture machines are often expensive, specialized to a field of research, and resistant to modifications. Recently a shift towards open-source science has brought about a new wave of Do-It-Yourself, customized continuous culture devices to accommodate a wide variety of experimental designs across several research areas [1–6].

A few such designs with detailed protocols have already been published in the literature; however, barriers in electrical, software, and biological expertise, or cost still leaves the assembly and operation of these machines out of reach of students or scientists in resource-poor settings [7–9]. In addition, increasing functionality of these machines while maintaining cost-effectiveness is an enduring challenge[10–13].

Here we present a protocol for the assembly and operation of a continuous culture device called the the EVolutionary biorEactor (EVE). It is a framework for a system which can support multiple individual bioreactors which we call culture units (CUs). Each of these CUs are capable of simultaneously running their own experiment with different optical density sampling rates, pump timings, optical density thresholds, antibiotics, and many other factors. While the EVE is able to replicate the morbidostat functionality defined in Toprak *et al.*, it is also able to replicate chemostat and turbidostat functionality[8]. To continuously monitor the cultures, the system proposed here utilizes low-cost consumer electronics, such as the Raspberry

Pi microcomputer, as well as cheap integrated circuits that communicate with the Pi. We have designed a printed circuit board (PCB) that can be assembled for faster setup but also support a breadboard implementation of the circuit for educational applications and to save cost. This system makes use of an open source programming language (Python) to power our software on the Pi. We have programmed a setup utility to allow users to quickly setup the software in the device. After the setup process, users can utilize a web interface to easily customize, start, and monitor experiments. Since the source code is readily available on GitHub (https://github.com/vishhvaan/eve-pi), users can implement quick modifications to the flexible framework. Modifiable plans for 3D printable parts for vial housings, housing bases, and circuit board files are freely available on GitHub. The 3D parts can all be printed in 24 hours without the need for special materials and at low cost.

Our device also features the capability of conducting multiple experiments (up to 16) simultaneously with one Raspberry Pi (referred to as multiplexing). The web interface is built to allow users to configure multiple experiments with a single interface. The live monitoring system automatically detects the number of systems running and optimizes the memory usage of the device to allow for monitoring all the experiments simultaneously.

Ease of use and extensibility were highly valued and prioritized when designing the system. We hope that the cost-effectiveness of the system and its ability to be setup by those with novice engineering and biological knowledge can lend to the system's utility both in the regular classroom environment and outreach evolution teaching efforts such as EvolvingSTEM[14]. Keeping in mind that users can design their own experiments, we demonstrate a few simple experiments that showcase the capability of the system that are ideal for users becoming familiar with the system, as well as students learning about evolutionary biology in an interactive way.

Description of the EVE

The physical functionality of the EVE is centered around a culture vial and is inspired by prominent bioreactors featured in literature[7–9,15,16]. For optimal growth conditions, the culture vial is stored in an incubator which is kept at 37C for the duration of the experiment. Within the incubator, the vial is on top of a holder which incorporates a fan with magnets glued to its top surface. When the fan is powered, magnets rotate to spin a magnetic stir-bar at the bottom of the culture vial. The stir bar has two functions – to facilitate diffusion of oxygen into the culture media and to keep the culture's turbidity uniform across the culture vial. The vial holder also houses two diodes (one LED and one photodiode). Throughout the experiment, the LED shines a light through the culture. Depending on the growth and corresponding turbidity of the culture, light is variably scattered through the culture, changing the amount of light that reaches the photodiode (PD). The LED and PD are connected to a circuit board (either a breadboard or PCB) that powers the devices and converts the analog information from the PD into digital information. The circuit board is connected to the Raspberry Pi with a ribbon cable, enabling the two boards to interact digitally. The culture vial is also connected to three pumps with three plastic tubes. Each pump in turn connects to either a media reservoir, a media reservoir containing cytotoxic drugs (antibiotics in this paper), or a waste reservoir. When powered by the circuit board, a specific pump activates which pumps either media or media mixed with drug into the vial. The waste pump also activates whenever the media pumps activate to prevent culture overflow. The waste pump can also provide the opportunity for easy temporal sampling. The software on the Raspberry Pi is responsible for triggering these pumps at specific points during the culture to achieve the (user defined) goal of the experiment. A schematic of the setup can be seen in Fig 1.

Of note, absorbance or the optical density of the culture is not directly reported to the Pi. Rather, the voltage of a resistor connected to the PD is measured. The voltage of this resistor increases as more light strikes the PD and therefore increases as more light is scattered - a surrogate of the absorbance of the culture. Since the voltage of the resistor is related to the optical density of the solution, we refer to the voltage value as the optical density of the culture in this article. This voltage value should not be interpreted as absorbance since they are numerically different. In many scenarios, the voltage value provides a clear picture of the state of the culture. However, if a true absorbance is needed, a simple calibration curve correlating the voltage value with an optical density can be performed.

Assembly

Assembly of the EVE can be separated into hardware and software tasks. We recommend beginning with hardware assembly as it requires more time to complete. Once the hardware is assembled, software can be installed on the Raspberry Pi and then seamlessly incorporated into the hardware as the last step of the assembly.

Hardware

The vial housing complex, fan with attached magnets, PD, and LED for each CU are housed in the incubator. The positive and negative (ground) wires that supply power to the LED, PD, and fan connect to the circuit board outside the incubator and extend into the incubator.

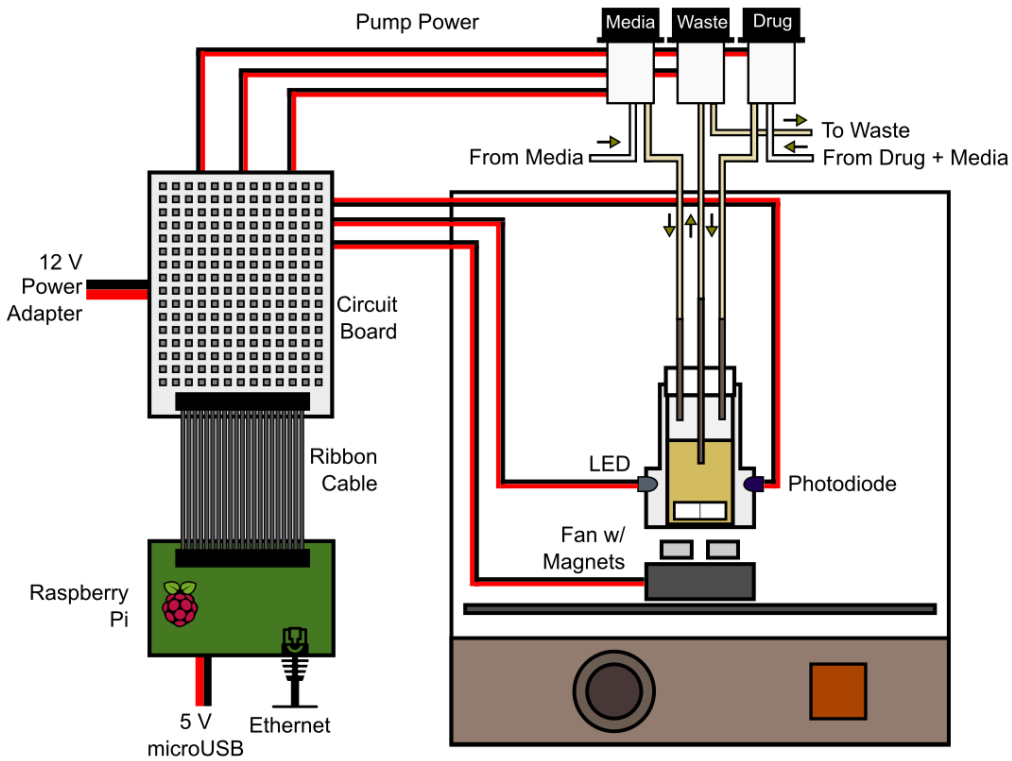


Figure 1. Hardware schematic of the EVE system. The culture vial is located inside the incubator. As the culture grows, an LED shines a light through the culture. Light that is reflected by the culture is received by the photodiode. A circuit board interprets and conveys photodiode readings to the Pi which runs control software. Based on user-configured directions, and optical density readings, the Pi uses the circuit board to power pumps connected to the vial. This strategy enables precise and dynamic control of the culture.

The 3D printable designs (.stl and .sldpart files) of the vial holder, base, and diode caps can be found on the device's GitHub repository. We used an Ultimaker S5 to print the components used in the experiments detailed below - though any 3D printer could be used. The vial holder is built to contain the experimental vial as well as the LED and PD. The diodes fit in openings on the side of the vial holder, which was adapted from Wong *et al.*[9]. The base of the vial holder (a separate component) houses the fan affixed with magnets and the vial holder. The vial holder base also raises the fan away from the metallic base of the incubator so there is no interaction between the base of the incubator and the solenoids of the fan.

If an incubator other than the one listed in the parts list is used for our experimental setup is to be used, the base file can be modified to ensure a fit with the incubator using any CAD software of choice . Finally, the caps fit on the diode openings to push the diodes into the openings and ensure a secure fit. These caps minimize diode positioning differences between each CU, therefore improve replicability of experiments.

Electrical connections between components can be made with solder or with heat shrink tubing. The components needed to solder (soldering iron, solder, etc.) are not included in the parts list can be easily found online or at a hardware store. Heat shrink tubing requires a heat gun to shrink the tubing. While the experiments below utilized soldered components, heat shrink tubing and a heat gun were used to test connections made with this method. Hardware assembly for a single CU system are as follows. These steps are influenced in many aspects by the work done by Toprak *et al.*[8]. For an experiment done in triplicate, repeat the steps detailed three times.

Pump Assembly

1. Label a set of three pumps with "Media", "Drugs", and "Waste" (one label per pump)
2. Cut six pieces of 5 cm large silicone tubing.
3. Cut small silicone tubing into six 45 cm long pieces. Ensure that the length of the small tubing is able to reach the culture vial or the media reservoirs from the pumps.

4. Attach two pieces of large silicone tubing to the two tubes protruding from each pump.
5. Fit a male leur lock on the free end the large silicon tubes.
6. Screw a female leur onto each of the male leur locks
7. Fit a small silicone tubing onto the female leur locks.
8. With the pumps facing up with the large silicone tubes at the top, label the large silicone tube on the left as the input.
9. Fit female leur locks at the end of the small silicone tubes attached to the "input" large silicone tubes.
10. Obtain three 60 cm red hook-up wire and three 60 cm black hook-up wire.
11. Curl one end of a red hook up wire around the positive terminal of a pump. Repeat for all three pumps.
12. Curl one end of a black hook up wire around the negative terminal of a pump. Repeat for all three pumps.
13. Use solder or heat shrink tubing to securely attach the wires to the pump.
14. Label the end of the wires with the pumps function (e.g. media, drug, or waste).

Diode Assembly

1. Obtain a LED and photodiode.
2. Obtain two 60 cm red hook-up wire and two 60 cm black hook-up wire.
3. To each diode, curl the red wire around the positive terminal and curl the black wire around the negative terminal.
4. Solder or use heat-shrink tubing to securely fasten the four wires to the terminal it is curled around.
5. Label the end of the wires with the diode function (e.g. LED, PD).

Fan Assembly

1. Place two spots of superglue across from each other on the axel of fan.
2. Place two 3D printed fan spacers on the superglue and apply pressure so that the spacers are flat.
3. Place a spots of superglue on each of the spacers.
4. Place two magnets on the superglue of each spacer and apply pressure so that the magnets are flat.
5. Use solder or heat shrink tubing to securely attach the long black and red wires to the short black and red wires protruding from the fan, respectively.
6. Label the end of the wires with "fan".
7. Allow the fan to dry for three hours before use.

Combine the prepared fan with the 3D printed components.

1. 3D print the vial holder, the base, and the caps (Fig 2h).
    - 3d printer settings:
        (a) Material: Black PLA
        (b) Layer height: 0.1
        (c) Print speed: 50 mm/s
        (d) Build plate temperature: 60C
        (e) Printing Temperature: 195C
        (f) 30% infill

(g) No support with components oriented upwards

(h) No adhesion

2. Place the base inside the incubator.
3. Place the fan inside the incubator and route the cables through the hole at the top of the incubator.
4. Fit the fan on the base by sliding the fan through the four columns on the edges of the base.
5. Fit the vial holder on the base so that it rests above the fan on the columns.
6. Place the LED and PD in the incubator and route the wires through the hole at the top of the incubator.
7. Fit the LED into the hole in the vial holder marked LED.
8. Fit the photodiode into the hole in the vial holder marked PD.
9. Ensure that the diode terminals are bent and protrude from the notches on the diode holes.
10. Cover the diode holes with the 3D printed caps. Ensure a secure fit by pressing down the cap firmly (Fig 2h).

The vials that are used during experiments need to be modified so that they can accept media and the drug solution and eject waste during the experiment. There are four tubes connected to each vial. Each tube is meant for a specific task and are distinguished from each other by their heights as seen in Fig 2e.

This is crucial, as there are specific reasons each tube has a different height. Below the cap, the tube meant to remove waste is longer so that it is able to come in contact with the surface of the culture media. The tubes for media and drug injection are shorter so that they do not touch the surface of the culture to prevent contamination. The tube connected to the air filter is the shortest so that it does not interact with the solutions pumped into the vial. Steps to modify the vials for experiments are as follows:

Vial Assembly

1. Cut a 7.5 cm piece of the small silicone tubing.
2. Cut four pieces of PEEK tubing, each 7.5 cm long.
3. Remove the cap of the vial and penetrate the cap with the PEEK tubing so that the insertion points are equidistant from each other.
4. Ensure that the PEEK tubes are at the appropriate height:

   (a) Two of the PEEK tubes should extend 1.25 cm below the cap and extend 3.8 cm above the cap.

   (b) One PEEK tube should extend 2.5 cm below the cap and 6.5 cm above the cap.

   (c) One PEEK tube should extend a 1 cm below the cap and 2.5 cm above the cap.

5. Apply the silicone sealant to both sides of the cap and around the protruding PEEK tubes to make the cap water and air tight.
6. Cut a 10 cm long piece of small silicone tubing.
7. Fit one end of the tubing to PEEK tube that extends 2.5 cm above the cap.
8. Fit the other end of the tubing to a female leur.
9. Push the female leur over the tapered end of a syringe filter.

Reservoirs of media and other liquids (for sanitation of the silicone tubes) also need to be made. These need to interface with the tubes from the pumps. Perform the following steps to create one reservoir:

Reservoir Assembly

Figure 2. Hardware components of the EVE. (a) Square fan with magnets attached to a spacer. (b) Three pumps for three functions (pumping in media, pumping in the solution of media and drug, and pumping out waste). All three are connected to the circuit board. (c) Silicone tunes connected to the pumps displaying terminal connections. (d) Pump in the 3D printed pump holder. (e) Three culture vials with 12mL of media and stirbars. The longest PEEK tube touches the culture. The other PEEK tubes meant to introduce media solutions to the vial do not touch the surface of the media. (f) Media reservoir with three small silicone tubes to transport media to the vials. (g) Terminal connections of the media reservoir. These connect to the terminal connections of the pump shown in (c). (h) 3D printed vial holder with vial inside. Diodes are held in place with 3D printed caps. (i) Top of the reservoir displaying the small silicone tubes attached with the silicone sealant (j) 70% Ethanol reservoir used for sterilization of the small silicone tubes prior to the experiment. Terminal connections and the attached air filter of the reservoir is displayed. (k) An incubator with three Culture Units. (l) The vial holder and fan with magnets fit into the base. Diodes are attached to the vial holder. Together, these components comprise a Culture Unit.

1. Obtain a 500 mL beaker with cap
2. Remove the cap from the beaker and drill 4 holes with a diameter of 3 mm in the cap.
3. Cut three pieces of 45 cm small silicone tube.
4. Cut one piece of 15 cm small silicone tube.
5. Insert one 45 cm silicone tube into a hole in the cap so that tube can touch the bottom of the beaker when the cap is placed on top of the beaker.
6. Apply silicone sealant to both sides of the cap to secure the silicone tube (Fig 2i).
7. Repeat the previous two steps for each of the 45 cm silicone tubes.

8. Insert the 15 cm silicone tube into the last hole so that only 2.5 cm is below the cap.
9. Apply silicone sealant to both sides of the cap to secure the 15 cm tube.
10. Insert male leur locks at the end of the three 45 cm silicone tubes above the cap.
11. Insert a female leur lock at the end of the 15 cm silicone tube above the cap.
12. Push the female leur over the tapered end of a syringe filter .

Circuit Board

Schematic and PCB files can be found on the project's GitHub (https://github.com/vishhvaan/eve-pi). Surface-mounted components including integrated circuits, resistors, diodes, and others can be found in the electronics section of the supplementary parts list . PCBs can be printed from the PCB file and the necessary circuit components soldered to board to create a functional circuit board. Four simultaneous evolution experiments can be conducted from one circuit board connected to one Raspberry Pi.

Steps to setup the circuit board are as follows:

1. Connect the circuit board to the 12-volt power adapter. Ensure that the LED on the board is on after it has been plugged in.
2. Connect the circuit board to the Raspberry Pi with the ribbon cable.
3. Ensure that the top of the Raspberry Pi's GPIO ('General purpose input/output pins' located in an array-like configuration on one side of the Pi) is connected to the top of the GPIO port on the circuit board.
4. Collect the labeled wires of the diodes, pumps, and fan.
5. Push the wire into its appropriate position on the labeled Phoenix block terminal connector (red wires to the positive terminal and black wires in the negative terminal)

If assembling the circuit board is not desired, a breadboard can also be used. The PCB schematics can be utilized to build a circuit on the breadboard. When purchasing resistors, diodes, and integrated circuits to build a breadboard circuit, through-hole components should be bought instead of their surface-mounted counterparts. A Raspberry Pi GPIO breakout board included in the parts list as an optional component recommended when building the circuit with a breadboard . In tests of building the machine, given the necessary components, an undergraduate engineering student (NW) was able to assemble the circuit in the course of five hours.

Software

The software in this device is built on the Python programming language and can be run from a single Raspberry Pi. Users interact with a simple web interface that is capable of controlling the hardware, running experiments, and editing configuration files, as shown in Fig 3. The web interface used in this project is forked from another open source script server (https://github.com/bugy/script-server) and modified to integrate with EVE control applications. This web interface can be utilized by multiple users connected to the local network, allowing for remote monitoring and control of experiments. This network capability of the device also allows it to save data to network locations, preventing lack of local space on the Pi from limiting experiment duration. The software allows data to be saved to a USB attached to the Pi or to network locations mounted to the Pi's file system.

Interaction and direct modification of the underlying Raspbian operating system or the core programs is unnecessary for most users who are unfamiliar with Linux or Python. However, since all the open-source software is contained in a clone of the GitHub repository, Python-savvy users can still be able to access the location of the files (/eve) and modify the programs to their specific needs.

When users run an experiment or control the hardware through the web interface, the executed programs rely on a configuration file that provides information about the hardware configuration. These parameters include the hardware address of the pumps, LED, and fan, duration of the experiment, time between pumps, thresholds for introduction of drug and media, and many others. This configuration file can be edited through a built-in configuration file editor that can be accessed through the web interface.

Since the circuit board has a pre-wired set of hardware addresses for the four culture units, a pre-populated configuration file for the circuit board is included in the software. For those who wish to build a breadboard version of the circuit, a sample configuration file is also included. Users can replace keys with their own parameters depending on the setup of their breadboard.

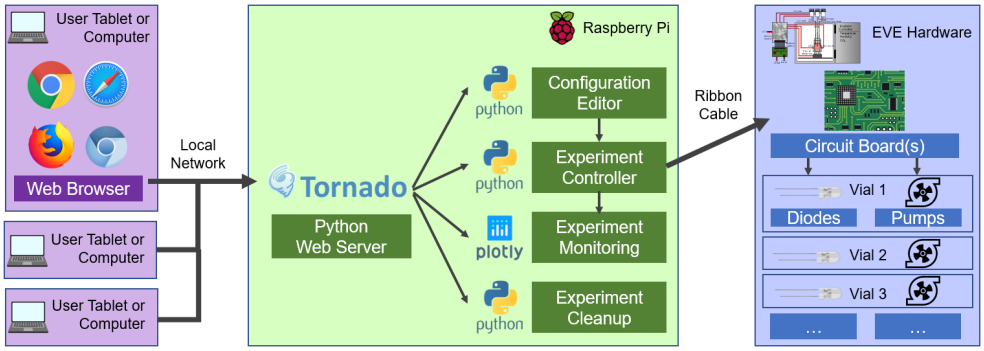


Figure 3. Software schematic of the EVE system. The device's software resides on the Raspberry Pi and center around a Python-based web-server which serves as the GUI for experimental design and control. This implementation offers built-in remote monitoring and multi-user control of the device. More EVE-centric Python-based applications are also installed Pi, accessible via the web interface. These applications are able to share data, and perform experiments based user defined configurations. The experiment controller application directly interfaces with the circuit board. All of these applications are open source and are built with extensibility in mind. Users have access to all the underlying programs and code. We encourage them to fork our GitHub repository or download programs to their devices and make modifications as they see fit. See Fig 1 for the hardware schematic of the EVE system.

The software offers experiment monitoring in two ways: through Slack and through a web server that is deployed automatically when the experiment begins. The web server, based on Dash by Plotly, allows any user on the local network access to real-time progress of the experiment through a web browser[17]. With both Slack and this web server, users can monitor:

1. Voltage of a resistor connected to the photodiode - a surrogate of the optical density of the solution inside culture vial(s). This voltage value can optionally be correlated to the OD600 of the solution through a calibration curve.
2. Time and duration of specific pump activation.
3. Calculated drug concentration inside the culture vial(s). These calculations do not account for drug consumed inside the vial. It is based on the concentration of the drug solution reservoir and the duration of pump activity.
4. Thresholds of optical density before media or drug pumps are activated.
5. Temperature of the incubator.
6. Concurrent threads of activity in the program during the experiment.

Installation

The software was tested on the Raspbian Buster operating system installed on a Raspberry Pi model 3B. The software instructions are written for this combination of software. However, these instructions should also work on any Debian-based operating system installed on versions of the Raspberry Pi 3 or 4. To maintain ease of use, the software installer can be run on an empty installation of Raspbian which will download and setup all software requirements. It will also ensure that the web interface launches every time the Raspberry Pi is restarted. Alternatively, a Docker image compiled on the ARM architecture is available on the GitHub repo. This image and associated Dockerfile can streamline the installation process and also allow for deployment across many ARM devices. The instructions below detail the setup process with the installer from the GitHub. Instructions for the installation of the Docker image can be found on the image's Docker Hub.

Steps to install the software on the Raspberry Pi are as follows:

1. Download and flash the Raspbian Buster OS on an SD card.
2. Insert the SD Card into the Pi and connect the Pi to a display, keyboard, and the local area network.
3. Note the IP address of the device with the command `ifconfig`.
4. Enter super user mode with the command: `sudo su`.
5. Run the setup script by entering the command: `bash <(curl -s https://raw.githubusercontent.com/vishhvaan/eve-pi/master/st_eve.sh)`

6. Follow the on-screen instructions to install the software.

7. If the Pi has a browser, navigate to the web interface by accessing: `http://localhost`.

8. On a browser on the network, navigate to the web interface by accessing the IP address or hostname of the device (e.g.: `http://eve.local_domain.net`).

Usage

The EVE web interface has several tabs that perform different functions. The functions of the tabs are as follows:

1. Pump control – This tab allows users to turn on specific pumps of specific CUs connected to the circuit board. The 'CU #' field allows users can select multiple devices simultaneously by entering a comma separated list (i.e. 1,2,3) or using the text "all" to select all the enabled CUs. Users are able to input time and the pump type (e.g.: media and drugs alone). The program relies on the configuration file to find the hardware addresses of the CUs. If the CU is disabled in the configuration file, the program will not activate the pumps for the system. This is a safety mechanism that prevents activation of mis-configured pumps.

2. One of the main functions of this program, in the context of the EVE, is sterilization of the silicone tubes prior to the experiment. When reservoirs of 10% bleach and 70% ethanol are connected to the pumps, this program can be run to pump fluid through the tubes, cleaning it prior to the exposure to media during the experiment. Instructions for these steps are detailed below.

3. Fan Control – This tab allows users to toggle power to specific or multiple fans.

4. Run Experiment – This tab runs the experiment on the EVE based on the settings in the configuration file.

5. Configuration Editor – This tab runs the editor for the configuration file. Users can view the configuration file and edit specific keys. This program offers an alternative for users who are not familiar with the Linux terminal and don't want to use SSH to edit the configuration file.

6. PD Test – This tab tests the functionality of the LED and photodiodes of specific or multiple CUs. Users have the ability to modify the time between queries and number of times each PD is queried. This program can be used to test the circuit and debug the setup.

7. Pi Power Control – This tab allows users to restart or power off the Pi.

8. Pumps and LED Off – This tab turns off all the pumps and LED in all the CUs connected to the EVE. This program provides a method to stop pumps or LED activity before the original program finishes running. In the event of an error, this program can also be used to an efficient way to halt the pumps the LED.

9. Resource Viewer – This tab displays system statistics for the Pi, including real-time CPU and memory usage.

After the experiment is begun, the live plotting interface is automatically deployed after the first set of data is written to the file (approximately 1 minute after the experiment starts). To visit the live plotting interface, users edit the web address based on the address of the web interface by adding the port 8050. For instance, if the users accessed http://localhost for the web interface, the live plotting interface can be found at `http://localhost:8050`. If the users accessed the device by the hostname or the IP address, the same rule applies (e.g.: `http://eve.local_domain.net:8050` if the web interface is found at `http://eve.local_domain.net`)

The live plotting interface, shown in Fig 4, can be navigated with the tabs at the top of the website. Every time the web page is refreshed, the Dash server processes the data from the output file and displays the most recent data. This *ad hoc* method of processing data for graphing proves to be less of a computational burden on the Pi compared to a strategy where data is processed on a regular basis. The first tab provides an overview of all the data while the other tabs (Pumps, Drug Concentration, Threads, and others) provide focused comparisons of specific data types. Users can select specific culture units to compare in the second row of tabs.

Experimental Use Cases

Prominent bioreactors featured in literature are very versatile and can be used to perform a variety of experiments. The EVE aims to follow suit by being capable of performing a variety of experiments. In this section, two basic biological experiments that can be carried out by the EVE are described.

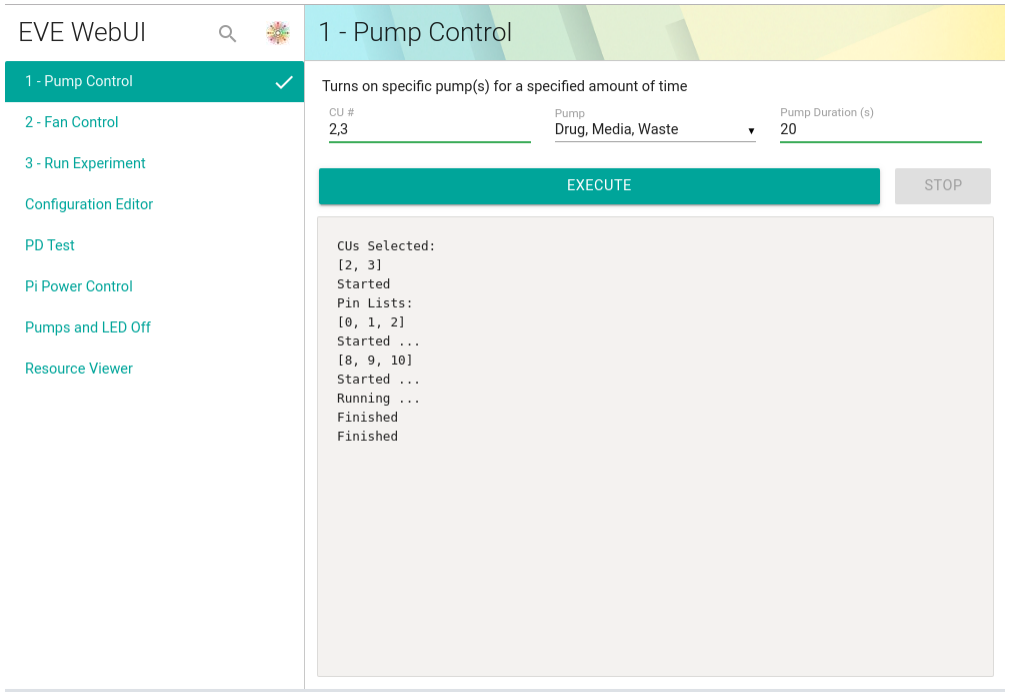


Figure 4. Web interface of the EVE running on the Pi. A variety of tabs are available to users to configure and run experiments, selectively turn on the pumps (useful for sterilization prior to the experiment), and control the Pi directly. This server is run from the Pi itself and is automatically installed after the user runs the setup script. This interface is accessible from any device connected to the same network the EVE resides on. It was built to allow ease of use and particularly, to provide non-programmers with a simple way of accessing and using the device. In this screen shot, specific pumps in multiple CUs have been turned on for twenty seconds.

Bacterial Growth

In this experiment, the real time bacterial growth of common K12 *Escerichia coli* can be followed by tracking optical density. This experiment can be especially enticing for students or educators since it requires less hardware to be setup and can provide a cross-disciplinary approach to biology. For this experiment the following list of parts are needed for single vial:

1. An incubator that can reach 37°C.
2. 3D printed vial holder, base, and caps.
3. 12V computer fan and magnets.
4. LED and photodiode as setup above.
5. A breadboard circuit or a circuit board.
6. Soldering or heat-shrink technology.
7. K12 E. Coli.
8. Broth to culture bacteria
    - If an autoclave is accessible:
        (a) Mueller Hinton (MH) Broth powder.
        (b) Distilled Water.
    - If an autoclave is not accessible:
        (a) Sterile pre-prepared broth

9. Culture Vials modified to only have 1 PEEK tube for air flow.

10. Magnetic stir-bar.

For a triplicate experiment, repeat the above steps three times. Note that no pumps or tubes are needed for this experiment. Once the system has been setup, as described above, users setup the vial to support bacterial growth. To do this, Sigma Aldrich MH Broth is first made by dissolving 21 g of powder in 1 liter of deionized or distilled water. For other brands of MH broth, brand-specific preparation instructions should be followed. It is always recommended to autoclave media after it is created to prevent contamination of the media itself. However, if an autoclave is not available, there are several alternatives proposed by the WHO in their "Methods of Sterilization"(https://apps.who.int/phint/pdf/b/7.5.9.5.8-Methods-of-sterilization.pdf). Sterilize the bench area and gloves with 70% ethanol. Work near a lit Bunsen burner to maintain a sterile field. Then, 12 mL of the broth is transferred to the culture vial with stir-bar inside. The vial is then inoculated with 0.5 g of the K12 *E. coli* and is placed in the holder. The experiment can then be started from the web interface.

To build a system that can multiplex multiple vials simultaneously, multiple 3D printed parts, computer fans, diodes, and vials are needed. The rest of the parts do not have to be multiplied.

As the experiment progresses, users can monitor the bacteria's growth via the live web interface or through the included Slack integration. At the end of the experiment, users can parse the recorded data and visualize the data using their own methods. Graphs of a trial we performed are included in Fig 5.

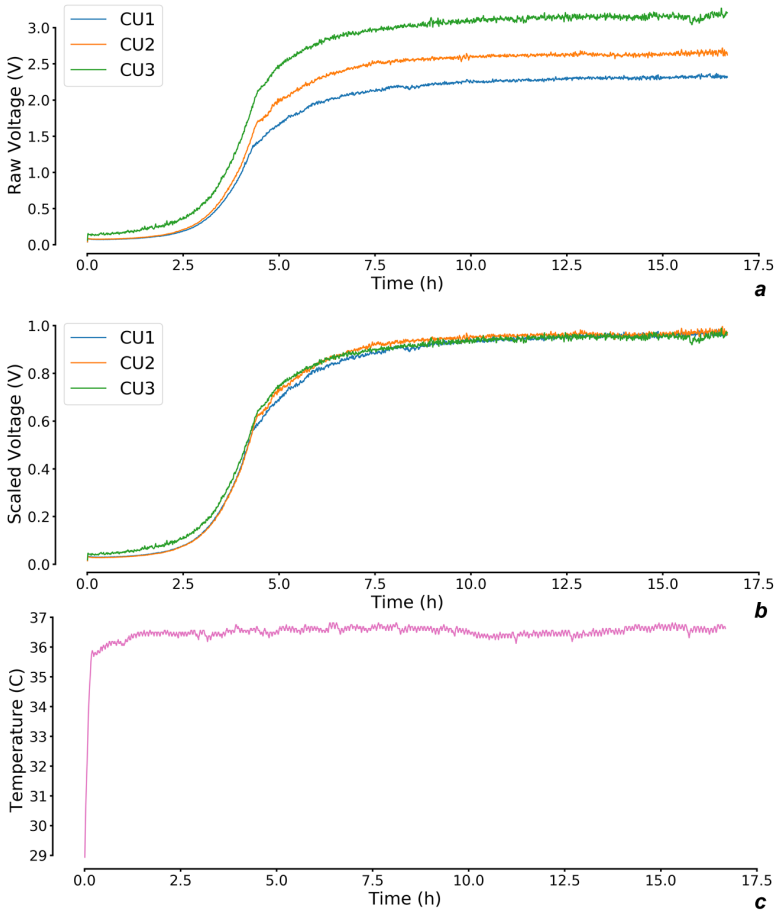


Figure 5. Sample graphs for a triplicate bacterial growth experiment. Triplicate bacterial growth experiments sharing the same axis. (a) Technical replicates (individual CUs) show different voltage readings over time despite being connected to the same circuit board and made identically. This variation is due to subtle differences in the positioning of the diodes in the holder. (b) When the readings are scaled by the maximum voltage reading, all curves overlap, exemplifying the triplicate nature of the experiment. Diode differences are constant over time, therefore raw voltage values can optionally be correlated with OD600 values. For simple experiments, this step is unnecessary. In general, this growth experiment can be readily performed by scientists to test their system or by students and teachers in scientific classes.

Evolution of Antibiotic Resistance

The second experiment demonstrates the EVE's full functionality by applying a selective pressure to bacteria as it grows. An antibiotic, in this case ampicillin, is introduced to the culture vial as the bacteria grows by passaging the drug mixed with media. Additionally, the culture is kept in the log-phase of growth during the duration of the experiment by passing fresh media into the vial while removing waste simultaneously. This experiment builds upon the setup of the simpler bacterial growth experiments. Pumps and silicone tubing would need to be setup for this phase of the experiment. In addition to the parts described in the bacterial growth experiment the following parts would be needed for a single vial antibiotic evolution experiment:

1. 3 pumps setup as described above.
2. Vials modified as described in the Assembly section.
3. Reservoir of MH broth (prepared as explained above).
4. Reservoir of MH broth with added antibiotics.
5. Reservoir of 10% bleach.

6. Reservoir of 70% ethanol.
7. Reservoir of distilled water.

To create a reservoir of MH broth with antibiotics, antibiotics are simply added to a reservoir of MH broth. The concentration of antibiotics in the reservoir is important. Toprak *et al.* found that 10 times the MIC inhibited growth sufficiently [8]. Since the duration of the activation of the drug pump is modifiable in the software, this is also a factor when calculating the dose of antibiotics in the reservoir. The shorter the pump time, higher concentrations of antibiotics are needed to produce the same inhibitory effect on the culture. Generally, with the default pump time of 0.75 seconds, we found that 10 times the MIC of ampicillin inhibited growth in our cultures. Users can optimize these parameters as needed depending on the antibiotics they use.

Again, for a triplicate experiment, three times the pumps and vials are needed and the steps below also need to be repeated thrice. The same reservoirs can be used for multiple replicates. When running an experiment on the EVE, especially one that will last several days, it is important to maintain a sterile environment – especially the silicone tubes, which are exposed to the media. During long-term (on the order of weeks) experiments, some groups have been known to clean the tubes between pump activations or pause the software. To sterilize the tubing:

1. Spray all luer connectors with 70% ethanol.
2. Attach the three outputs of the 10% bleach reservoir to the inputs of the media, bleach, and waste pumps.
3. Run all three pumps for 60 seconds.
4. Disconnect the 10% bleach reservoir from all three pumps.
5. Attach the three outputs of the 70% ethanol reservoir to the inputs of the media, bleach, and waste pumps.
6. Run all three pumps for 60 seconds.
7. Disconnect the 70% ethanol reservoir from all three pumps.
8. Attach the three outputs of the distilled water reservoir to the inputs of the media, bleach, and waste pumps.
9. Run all three pumps for 60 seconds.
10. Disconnect the distilled reservoir from all three pumps.
11. Attach one output of the media reservoir to the input of the media pump.
12. Run the media pump for 20 seconds to preload media in the tube.
13. Attach one output of the media reservoir to the input of the drug pump.
14. Run the drug pump for 20 seconds to preload drug solution in the tube.

After this step, the vial is ready to be placed in the holder and attached to the tubes for the media, drug solution, and waste:

1. Obtain a 2-liter Erlenmeyer flask and label it as waste.
2. In a sterile field, transfer 12 mL of MH broth to the culture vial with a stir-bar inside.
3. Inoculate the vial with 0.5 g of the K12 E. coli.
4. Close the vial with the cap.
5. Place the vial on the holder.
6. Attach one output of the media reservoir to the input of the media pump.
7. Attach one output of the media reservoir to the input of the drug pump.
8. Insert the input tube of the waste pump into the incubator through the opening at the top.
9. Connect the tube to the longest PEEK tube of the vial.

10. Insert the output of the media pump and the output of the drug pump into the incubator through the opening at the top.

11. Connect these two tubes to the remaining two short PEEK tubes of the vial.

After these steps are completed, the experiment is ready to begin and can be started from the web interface. To keep track of the concentration inside the vial, the concentration of the drug inside the reservoir is modified in the configuration file before the experiment starts. Based on the amount of time each pump is run (another variable that can be modified in the configuration), the total amount of drug in the vial can be tracked. Sample graphs of this experiment are shown in Fig 6. This experiment demonstrates the selection of an *E. coli* culture with ampicillin in a CU. In this figure, there are small dilutional effects as a result of the media being added to the culture vial. The effect of the initial addition of the drug solution is significant – killing 33.5% of the population. According to the selection algorithm, the drug concentration is then lowered to allow for the recovery of the population. This is done by repetitively adding fresh media to the culture while removing waste. In the experiment, the bacterial population is able to recover in 12 hours. The drug solution is then re-introduced to the culture. After this second introduction, the population size decreases 15%, indicating development of tolerance/resistance to the drug. The population is again allowed to recover for a shorter period – around 6 hours. After this recovery period, the addition of more ampicillin does not kill the population. At the end of the experiment, the cells are effectively resistant to ampicillin. Resistance of the resultant population can be confirmed with an MIC assay[18]. The experiment can be performed with a variety of different drugs, however drug stability in cell culture should be considered. For instance, ampicillin is stable in culture for 3 days[19]. If the experiment lasts longer than the stability of the drug in culture, the drug solution can be refreshed by replacing the drug reservoir with one that is freshly made.

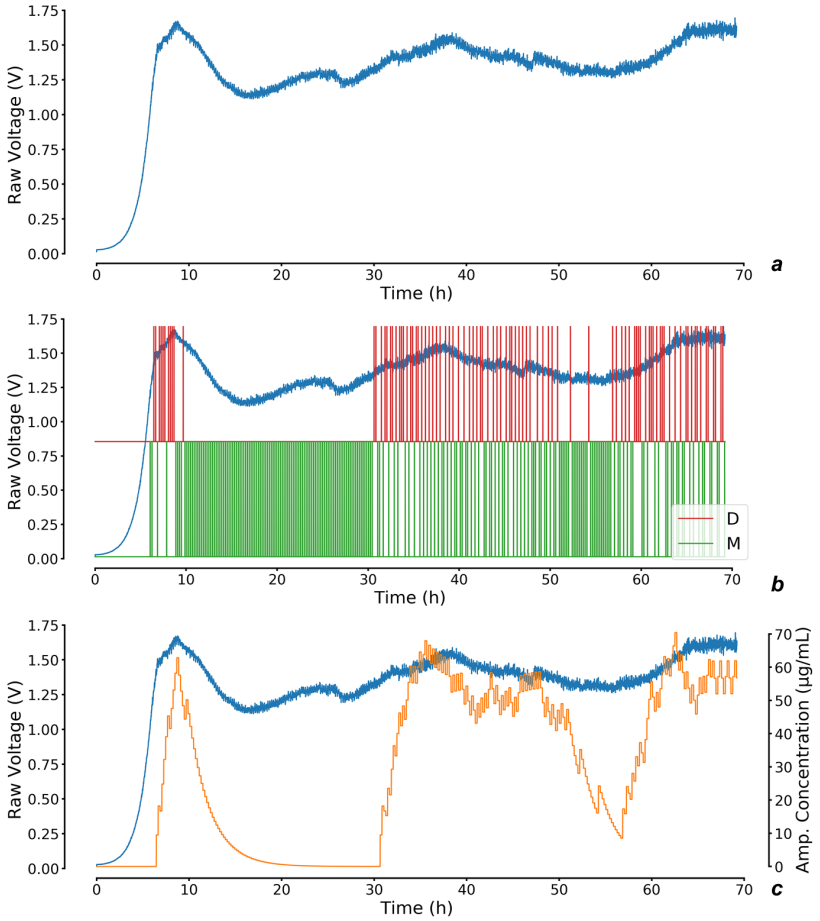


Figure 6. Sample bacterial growth experiment with an ampicillin selective pressure applied by the Toprak selection algorithm[8] All three graphs are generated in real-time during the experiment. Users can monitor the activity of a culture unit through the graphs. (a) Voltage over time, as measured from CU photodiode. These values can be optionally co-related to the calibration curves in the supplemental documents to achieve OD600 readings. (b) Pump activation shown over time. The red lines represent the activity of the pump that passages a solution of media and drugs into the vial. The green lines represent the activity of the pump that passages media into the vial. Whenever the drug and media pump activate, a waste pump also activates to maintain volume equilibrium. The activity of this pump is not shown in the figure as it matches the sum of the activity of the drug and media pumps. All pump activity is recorded in the software. (c) Calculated drug concentration plotted over time. Values are calculated from the concentration of the drug solution reservoir and the duration of pump activity. These values do not take into account consumption of the drug in the vial and only account for dilution effects.

Discussion

While extensibility was of prime importance, device affordability was also an important consideration during the design of the device. Based on the parts used in our experiment, an EVE build for simple growth experiments with triplicate CUs can be built for approximately $72, while an EVE build for directed growth experiments with selective agents will cost approximately $154. This cost is an estimate for all the parts excluding the incubator, 3D printer costs, and any glassware needed to prepare the solutions for the experiment. A system meant simply meant for growth experiments costs cheaper since it does not require pumps or tubing – parts which can add significantly to the total cost of the system. Vendor costs are also listed on the parts list, so costs can be driven down significantly for users who buy their parts from commercial vendors such as eBay or Amazon. This is particularly relevant for those who wish to implement the system in the educational setting. This low cost enables labs to explore concepts of evolution without investing much capital into tools. The cost also makes this project feasible for the educational environment, where cost is not a burden for educators to investigate the feasibility of this project in their classroom or extra-curricular setting. The two experiments described above serve as tests of the hardware and software and demonstrate the most essential components of the system. Successful completion of the first experiment confirms proper operation of the

LED sensors and the connections from the microcontroller. In addition, it provides ample opportunity to identify sources of contamination.

The second experiment demonstrates the pumps and the 'morbidostat' algorithm in action. The algorithm incorporated into the system is a modification from Toprak *et al.*[8] In its original conception, maximum and minimum OD thresholds are determined from pre-experiment bench tests and are dependent on a number of variables including the species of bacteria used. Different trials of the same bacteria might also yield different maximal optical density readings and the bacteria might enter the log phase at different optical densities owing to the stochasticity of evolution[20] as well as variation in ambient light and other experimental conditions. To account for this variation, the control program is capable of shifting the threshold automatically. It does this by estimating the first derivative and second derivative of the real-time optical density (real-time growth rate and real-time growth acceleration respectively). Based on these readings, it shifts the threshold to the optical density when the growth rate crosses a certain pre-specified threshold and the growth acceleration is approximately zero. In the results and conclusion presented here though 'optical density' is discussed, the true absorbance was not calculated or derived, with the data reported being raw voltage data from the photodiode. A simple calibration of this data using a standard spectrophotmeter is recommended to report data from the machine in terms of true optical density.

A further, novel application of this system is the ability to manage multiple protocols at once. The capability of the EVE system to manage multiple protocols in this manner makes it the only system of its kind that is fully open source and of low-cost. Such versatility allows this system to be used as a comprehensive automated evolution experiment management tool for numerous diverse experiments in a self-contained integrated manner. In the experiment above, the default algorithm used is identical to that used in the original paper by Toprak and colleagues (Figure 1 in that paper). In this algorithm, the culture is allowed to recover by sequentially passing fresh media after exposure to the drug. However, there is an alternative control algorithm that can be used during the experiment. Users can choose to maintain a constant concentration of drug once the bacterial population is sufficiently healthy inside the tube. Biologically, since the selective pressures of the different populations were applied differently, the way resistance is reached between the two populations is different.

The maximum theoretical drug concentration inside the vial is equal to the concentration of the drug inside the drug solution reservoir. Maximizing this theoretical drug concentration, however, is at odds with slowly raising the concentration of the drug inside the vial. If the reservoir is too concentrated and if the drug solution pump replaces 1/12th of the solution, it is likely that the drug concentration will be extremely high inside the vial and kill all the bacteria. A solution to this problem is to allow the drug solution pump to run for different lengths of time depending on the stage of the experiment. If instead of replacing 1/12th of the solution at the beginning, if the pumps run for a shorter time, a lower concentration of drug can be achieved inside the vial. Later in the experiment, the drug pumps can be run for longer, ensuring that the population is exposed to the maximal possible drug concentration. A maximum ΔV (such as the 1mL proposed by Toprak and colleagues) can be specified so that the culture population does not get washed away by the drug solution. This way, development of resistance is assured and the population faces strong selection pressures as the experiment progresses.

The EVE also easily lends itself to fluorescence experiments where LED and PDs corresponding to excitation and emission frequencies can be incorporated into the design. By using fluorescence instead of optical densities, growth of several populations can be distinguished. In the context of varying selective pressures (provided by the antibiotics), interactions of different populations of bacteria can be studied to further elucidate the effect of these interactions on the generation of drug resistance. Game theory and other methods of mathematical modeling can be used to draw conclusions about the interactions of the different species/types of bacteria[21,22].

Limitations and Future Work

There are several limitations of the device. Naturally, since all the software is executed from the Raspberry Pi, memory and processing power is limited during the experiment. Many of the programs, especially the master program that interfaces with the circuit board during the experiment, have been streamlined to utilize as little of the memory and processing power as possible. A queuing system was implemented to prevent many threads that utilize high processing power from executing at the same time. Furthermore, as more culture units and corresponding sensors are configured, an increasingly lower resolution list of the data is generated in addition to the high-resolution optical density data. This lower-resolution data is used for the live graphing to prevent the greater processing power that's required to graph high resolution data.

To circumvent processing power and memory constraints from preventing users to implement more CUs, future work on a server-based version will be implemented to take advantage of remote processing power while allowing the Raspberry Pi simply interface with the circuit board and relay information to the server. In this scenario, the web interface and all the data are managed by the server. Docker integration in this step will be crucial to streamline the process and create a simplistic but powerful and robust way to deploy the EVE system across the local network.

Another limitation of the current design is that the circuit board only supports 4 simultaneous culture units. In this scenario, the hardware addresses are "hard-coded" into the circuit board design. An alternative, but more complicated design, would

allow for jumpers on the circuit board, allowing users to easily select four different hardware addresses. The user is then able to attach circuit board together with one circuit board interfacing with the Raspberry Pi. Ultimately, for users who would like to increase the level of multiplexing, with 4 circuit boards connected to each other, a maximum of 16 CUs will be able to be controlled simultaneously. An advantage of this design is its modular capabilities, allowing users to only print as many circuit boards as they need and limit the number of components needed to make a functional circuit board and reduce overall system cost. A future version of the circuit board will implement this design improvement.

Another future direction is to allow the machine to sequentially inject multiple drugs autonomously. The current version of the proposed system only passages one drug solution and therefore can only explore the pathways to specific drug resistance. However, there are many biologically and clinically relevant questions that can be asked when different drugs are given sequentially or together. Concepts like collateral sensitivity and cross-resistance can be effectively studied – especially when combined with frequent sampling of the bacterial population[20,23]. Tracking the concentrations of multiple drugs in the vial and creating algorithms for drug scheduling can be added in future updates or merged to the main repository from community-driven efforts. In this scenario, a microfluidic solution such as that implemented by Wong *et al.* would be useful[9].

Conclusion

We present a low-cost, open-source, extensible, and functionally-flexible bioreactor design. The utility of this machine is not just limited to the lab environment where the evolution of drug resistance can be studied with high temporal resolution, but extends to the educational environment where students and educators can engage in an interdisciplinary project to study biological concepts through technology.

The proposed framework can have many applications, including many not discussed here. Indeed, use of this device may be used to study a wide variety of biological concepts – even outside the field of evolutionary biology. The open source nature of the project welcomes modification to the hardware and the software for more specialized applications of the technology. We hope the versatile and accessible nature of this device makes it useful for scientists and educators from all areas.

Supporting information

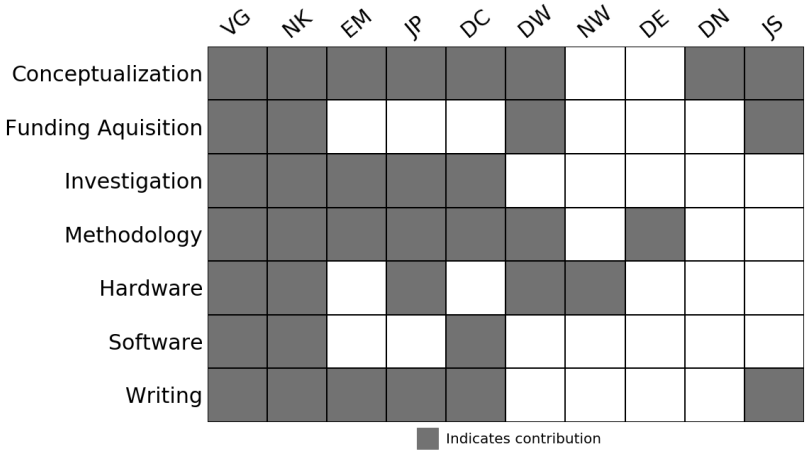
S1 Appendix. Parts list. List of all parts that comprise the EVE system with estimated price: https://github.com/vishhvaan/eve-pi/raw/master/PartsList.xlsx.

Acknowledgments

Rick, our 3D printer. We love you mate. In addition, we'd like to thank VeloSano for their generous support of this work.

Author Information

Contributions



References

1. Nash, D. B. Open-source science. *Biotechnol Heal.* **2**, 8–8 (2005). 23393443[pmid].

2. Matteau, D., Baby, V., Pelletier, S. & Rodrigue, S. A small-volume, low-cost, and versatile continuous culture device. *PLOS ONE* **10**, 1–13, DOI: 10.1371/journal.pone.0133384 (2015).
3. James M. Gill, I. The internet of things in the life sciences laboratory. *SLAS TECHNOLOGY: Transl. Life Sci. Innov.* **23**, 405–406, DOI: 10.1177/2472630318793101 (2018). PMID: 30232943, https://doi.org/10.1177/2472630318793101.
4. Cressey, D. The DIY electronics transforming research. *Nature* **544**, 125–126 (2017).
5. Perkel, J. M. The Internet of Things comes to the lab. *Nature* **542**, 125–126 (2017).
6. Pilizota, T. & Yang, Y.-T. "do it yourself" microbial cultivation techniques for synthetic and systems biology: Cheap, fun, and flexible. *Front. Microbiol.* .
7. Yoshida, M. *et al.* Time-programmable drug dosing allows the manipulation, suppression and reversal of antibiotic drug resistance in vitro. *Nat. Commun.* **8**, 15589 EP – (2017). Article.
8. Toprak, E. *et al.* Building a morbidostat: an automated continuous-culture device for studying bacterial drug resistance under dynamically sustained drug inhibition. *Nat. Protoc.* **8**, 555 EP – (2013).
9. Wong, B. G., Mancuso, C. P., Kiriakov, S., Bashor, C. J. & Khalil, A. S. Precise, automated control of conditions for high-throughput growth of yeast and bacteria with evolver. *Nat. Biotechnol.* **36**, 614 EP – (2018).
10. Takahashi, C. N., Miller, A. W., Ekness, F., Dunham, M. J. & Klavins, E. A low cost, customizable turbidostat for use in synthetic circuit characterization. *ACS Synth Biol* **4**, 32–38 (2015).
11. AU Liu, P. C., AU Lee, Y. T., AU Wang, C. Y. & AU Yang, Y.-T. Design and use of a low cost, automated morbidostat for adaptive evolution of bacteria under antibiotic drug selection. *JoVE* e54426, DOI: 10.3791/54426 (2016).
12. AU Miller, A. W., AU Befort, C., AU Kerr, E. O. & AU Dunham, M. J. Design and use of multiplexed chemostat arrays. *JoVE* e50262, DOI: 10.3791/50262 (2013).
13. Hoffmann, S. A., Wohltat, C., Müller, K. M. & Arndt, K. M. A user-friendly, low-cost turbidostat with versatile growth rate estimation based on an extended kalman filter. *PLOS ONE* **12**, 1–15, DOI: 10.1371/journal.pone.0181923 (2017).
14. Cooper, V. S., Warren, T. M., Matela, A. M., Handwork, M. & Scarponi, S. Evolvingstem: a microbial evolution-in-action curriculum that enhances learning of evolutionary biology and biotechnology. *Evol. Educ. Outreach* **12**, 12, DOI: 10.1186/s12052-019-0103-4 (2019).
15. Milias-Argeitis, A., Rullan, M., Aoki, S. K., Buchmann, P. & Khammash, M. Automated optogenetic feedback control for precise and robust regulation of gene expression and cell growth. *Nat. Commun.* **7**, 12546 EP – (2016). Article.
16. Dößelmann, B. *et al.* Rapid and consistent evolution of colistin resistance in extensively drug-resistant pseudomonas aeruginosa during morbidostat culture. *Antimicrob. Agents Chemother.* **61**, DOI: 10.1128/AAC.00043-17 (2017). https://aac.asm.org/content/61/9/e00043-17.full.pdf.
17. Inc., P. T. Collaborative data science (2015).
18. Andrews, J. M. Determination of minimum inhibitory concentrations. *J. Antimicrob. Chemother.* **48**, 5–16, DOI: 10.1093/jac/48.suppl_1.5 (2001). http://oup.prod.sis.lan/jac/article-pdf/48/suppl_1/5/19569295/0480005.pdf.
19. Perlman, D. Use of antibiotics in cell culture media. **58**, 110 – 116, DOI: https://doi.org/10.1016/S0076-6879(79)58128-2 (1979).
20. Nichol, D. *et al.* Antibiotic collateral sensitivity is contingent on the repeatability of evolution. *Nat. Commun.* **10**, 334, DOI: 10.1038/s41467-018-08098-6 (2019).
21. Kaznatcheev, A., Peacock, J., Basanta, D., Marusyk, A. & Scott, J. G. Fibroblasts and alectinib switch the evolutionary games played by non-small cell lung cancer. *Nat. Ecol. & Evol.* **3**, 450–456, DOI: 10.1038/s41559-018-0768-z (2019).
22. Friedman, J., Higgins, L. M. & Gore, J. Community structure follows simple assembly rules in microbial microcosms. *Nat. Ecol. &Amp; Evol.* **1**, 0109 EP – (2017). Article.
23. Maltas, J. & Wood, K. B. Pervasive and diverse collateral sensitivity profiles inform optimal strategies to limit antibiotic resistance. *bioRxiv* DOI: 10.1101/241075 (2019). https://www.biorxiv.org/content/early/2019/01/04/241075.full.pdf.