mEcosystem

by Christopher Hannon
Illinois Institute of Technology
Chief Science Officer (CSO) of mEcosystem Labs
Embedded Systems ITMT 593
A20265098
channon@iit.edu

Table Of Contents

Introduction

Aquaponics is an agricultural process that combines aquaculture with hydroponics. Aquaculture is the keeping of fish and marine life and hydroponics is the growing of plants in soilless media. mEcosystem is the name given to this line of home aquaponic systems that contain embedded controllers. The embedded system controls all factors in the ecosystem required for the growing of food.

Acknowledgment

This project is developed from mEcosystem Labs including Neil Griffin as CEO, as well as Dane Christianson of Moving Parts llc. as mechanical engineering consultant, Nik Rokop as business advisor, and Jeremy Hajek as technology advisor. This project was developed for use as a new business venture utilizing IPROs interprofessional community atmosphere and ITMT 593 as the technological foundation for the embedded system's operation.

What is Aquaponics?

Aquaponics is the growing of plants with fish waste. Fish waste gets broken down by bacteria into nitrates that the plants use to grow. The plants then take the nitrates out of the water and collect solid waste in their roots. This process serves to filter the water in the aquarium making the water clean. The solid wastes continue to get broken down by bacteria in the plant roots. This cyclic process takes a few weeks to establish as the plants start to grow and bacteria begins to cultivate. mEcosystem uses flood and drain techniques to cycle the water through the system.

Getting Started

There are many difficulties to overcome in starting an aquaponic system. When starting out, the tank is void of Ammonia. Ammonia is produced through fish waste and excess food decomposing. Fish excrete ammonia through their gills as well. Without large amounts of ammonia, the bacteria that breaks it down into nitrates does not cultivate. When starting, I added a few fish a week as to not overproduce ammonia. For flood and drain systems, the water including the waste must be pumped up into the plant beds periodically. Every 90 minutes is okay for a baseline. Initially the ammonia is produced quicker than the naturally occurring bacteria can grow. This results in a high PH and if it gets too high will harm the fish and kill the plants. The solution to this is large frequent water changes that help reduce the concentration of ammonia from the water.

A method to speed up the growth of bacteria is to take it from an existing system. The media in the grow beds of mEcosystems is Growstone Hydroponic Grow Substrate. These rocks are made from recycled glass and enable the growth of bacteria in their porous material. This also allows the fish water and air to reach the bacteria and roots of the plants. Taking some of these rocks and adding it to a new system can help speed up the process of growing bacteria

Marine Life

Traditionally, commercial Aquaponic systems use edible fish such as cod, tilapia, catfish, bass and perch. Tilapia are the most common because they grow the fastest, easiest and can withstand low temperature water. Commercial systems differ mainly from mEcosystems because they have large tanks that can house the fish.

I use a variety of marine life in mEcosystems. Initially The system started exclusively with tiger barbs. Soon after I expanded to other barbs, tetras, mollys, and zebrafish. In order to create a more well rounded ecosystem, I introduced a crayfish (small lobster), a variety of snails including nerite and mystery snails. There is also bacteria in the gravel and small microbial life.

By having a variety of organisms, more of the waste can be broken down. For example, the microbial life will consume the uneaten food that drifts to the gravel and the snails will eat algae that grows commonly on the glass tank.

Plant Life

Many plants can be grown in aquaponic systems. Lettuces, kale, swiss chard, chives, spearmint, green onions, cilantro, coriander, lavender, lemon basil, and tomatoes have all been grown and show great promise. Plants like Broccoli, onions, and peppers are being tested to see how well they grow.

Embedded Solution

By Utilizing an embedded system, the controls of the mEcosystem can be autonomous allowing for easier use of the system that performs reliably and is smart. The internet of things revolution has let to services like temboo that help put sensor data to the cloud.

Control Factors

In mEcosystem, there are numerous factors that require control. When growing plants indoors, the plants must be supplemented with grow lights that are specialized for growing plants. In mEcosystem there are two fluorescent lights each with four T5 bulbs. These lights must be turned off and on at various times of the day. Additionally there is a water pump that must run periodically for various durations. There is a heater that maintains the temperature of the water. The fish require a small light to live during the day. Lastly, there is an air pump that should run continuously.

Data Collection

The types of data that can be collected range from environmental conditions to aquarium conditions. The temperature of the air and water are the foundations of the systems measurable conditions. Adding to this is the humidity of the air near the plants and of the ambient air. The aquarium must have a stable PH in order for the fish and plants to both be happy.

The Ammonia and the nitrates are also important factors of the system to show the balance and proportions of the biological and chemical makeup. The plants convert CO_2 into O_2 so those factors would be valuable data as well.

In order to determine if there is enough light in the environment (if it's by a window on a sunny day), a luminosity sensor would be helpful. Finally a camera on the system that could create timelapses is informative of how the plants are growing.

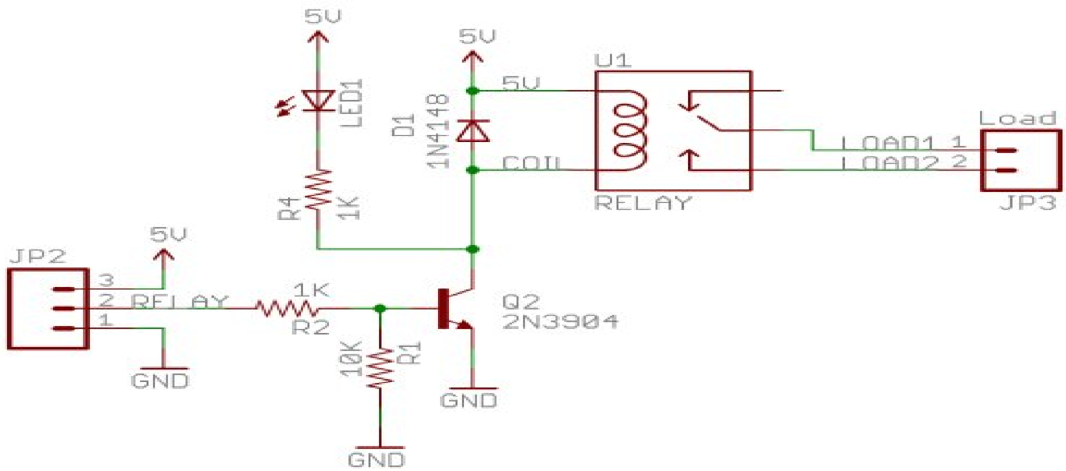
Components

There are a number of sensors currently working with mEcosystems, Temperature, humidity, and PH. The sensors mentioned in the Data Collection section include sensors that are not cost effective or that are not yet implemented. In order to control the lights, pump, etc, there are six relays on the embedded system that can control AC outlets. The arduino that is used is called the Arduino Yun. This arduino has the same 16Mhz clock as the regular arduino uno and the same memory. However the Yun has an added benefit of a linux processor onboard that has a 400Mhz processor and comes with built in ethernet and wireless systems. There is also a micro SD slot for additional memory and data storage. The Linux chip adds the benefit of being able to push data to the cloud as well as host an onboard web application that allows for more robust configuration of the arduino sketch.

There are six fans that circulate the air in the system. One fan at each side of a mEcosystem blow air at the plants to strengthen them as they grow. This is similar to how the wind affects plants naturally.

Relay Circuit

The relay circuit are built from a diode, transistor, 2 resistors and the relays themselves. The relays require a minimum of 20mA to change state. Since a digital pin does not have enough current to change state the transistor is needed to connect the 5v power to ground through the relay. The transistor allows for this when the digital output supplies its additional current to the transistor. The diode restricts power from the electromagnetic interference of the relay changing state preventing damage to the arduino.



There are six relays that each control one outlet. Four additional outlets are in the system that are always on and do not require relays. They power the arduino, fans, and air pump.

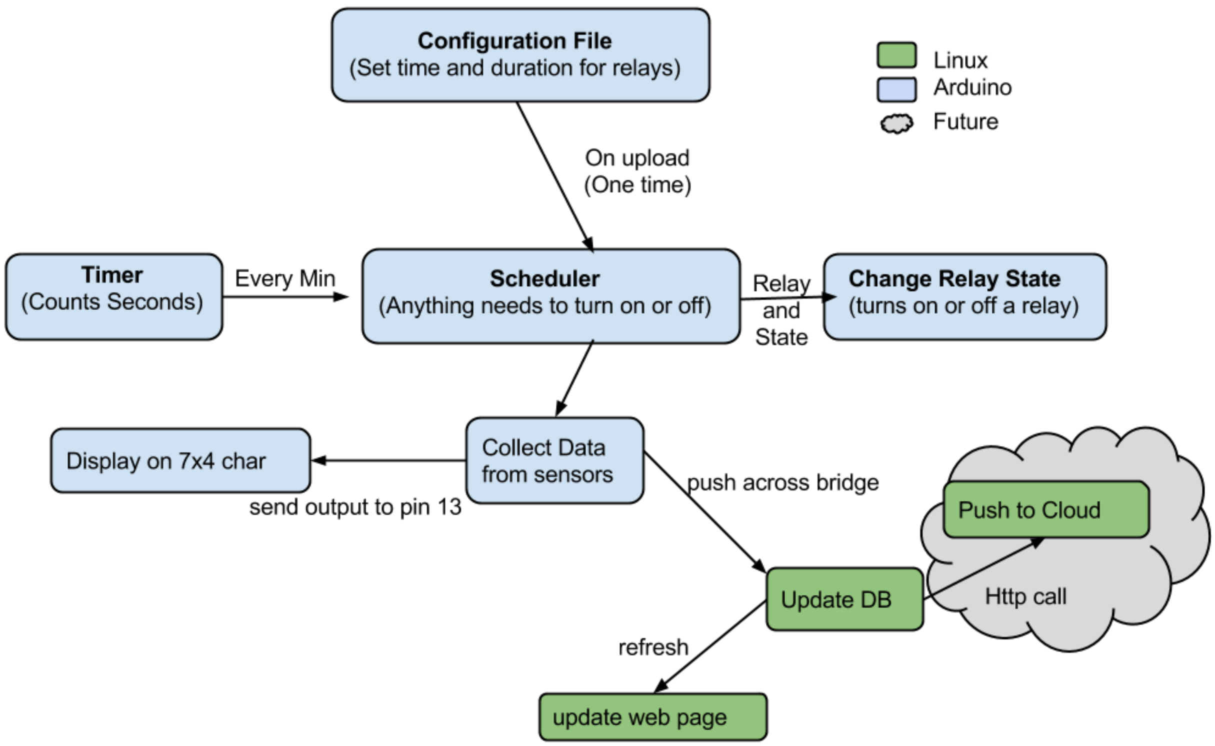
Energy Consumption

Component	**Wattage**	**Quantity**	**Total**
Plant Light	100	2	200
Fish Light	25	1	25
Air Pump	10	1	10
Heater	100	1	100
Pump	35	1	35
Arduino/Fans	8	1	8
		Total:	378

The lights only run for about 12 hours a day depending on the type of plants and and the pump only runs for a fraction of the time about 5% on. Lastly depending on the type of fish and the temperature of the environment the heater will not always be on. This will reduce the average energy consumption to about 200 Watts on average. at $0.17 a kilowatt hour, is about $0.82 a day maximum.

Embedded Code

The embedded code revolves around 2 main components, the timer and scheduler.



The configuration File is set on arduino sketch upload. It will set a few data structures in the beginning of the sketch. The timer counts time telling the scheduler to run. The scheduler determines if any relays need to be changed state. It will then collect the sensor data and push it to the linux side of the board as well as to a 7x4 character display.

Timer

The timer requires a prescaler and a counter so that the interrupt at the scheduled time of 1 Hz which is one time per second. The configuration is as follows:

```
noInterrupts();             // disable all interrupts
TCCR1A = 0;
TCCR1B = 0;
TCNT1  = 0;
OCR1A = 62500;              // compare match register 16MHz/256/2Hz
TCCR1B |= (1 << WGM12);     // CTC mode
TCCR1B |= (1 << CS12);      // 256 prescaler
TIMSK1 |= (1 << OCIE1A);    // enable timer compare interrupt
interrupts();               // enable all interrupts
```

The interrupt counts seconds, minutes, and hours programmatically. Every minute the scheduler flag gets set so that the scheduler runs.

```
ISR(TIMER1_COMPA_vect) {          // timer compare interrupt service routine
               //This happens every 1 second (earth time)
               s++;
               if (s == 60) { //aka 1 minute
                 m++;
                 s = 0;       // reset second count
                 //check sched flag
                 sched = 1;  // this runs the scheduling routine to check if any relays
       need to be checked
               }
               if (m == 60) { // aka 1 hour
                 h++;
                 m = 0;       // reset min count
               }
               if (h == 24) {
                 h = 0;
               }
              }
              //////////////////////////  END TIMER 1 INTURRUPT ////////////////
```

Scheduler

This is how the scheduler knows when to change a relay. It operates similar to the cron table by checking the current time against when the event is to occur. By converting the hours into minutes we can use the modulus function to determine if the pump should run no matter the interval set.

```
//////////////////////////  main loop  ////////////////////////////
void loop() {
 // put your main code here, to run repeatedly:
 if (sched) {
   Serial.println("in sched");
   sched = 0;
```

```
    //check stuff
    //lightT
    if (m == lightTmin) {
      if (h == lightThr) {
        Serial.println("relaying T");
        relayOn(lightT);
      }
    }
    if (m == (lightTmin + lightTdmin)) {
      if (h == (lightThr + lightTdhr) % 24) {
        relayOff(lightT);
      }
    }
```

Communication to Webserver

```
  #if (YUN)
   if (client.connected()) {
     //handle client
     String data = (String(lightTmin) + ',' + String(lightBmin) + ',' + String(lightFmin) + ',' +
                    String(lightThr) + ',' + String(lightBhr) + ',' + String(lightFhr) + ',' +
                    String(lightTdhr) + ',' + String(lightBdhr) + ',' + String(lightFdhr) + ',' +
                    String(lightTdmin) + ',' + String(lightBdmin) + ',' + String(lightFdmin) + ',' +
                    String(pumpFreq) + ',' + String(pumpD) + ',' +
                    String(setTemp) + ',' +
                    String(s) + ',' + String(m) + ',' + String(h)
                   );
     client.println(58.123456, 5);
     Serial.println(data);
   }
   else {
     client = server.accept();
     if (client.connected()) {
       Serial.println("User is connected!");
     }
   }
  #endif
   printTime();
```

```
}
////////////////////////////// END MAIN LOOP /////////////////////////////
```

Linux Code

The linux operating system runs MIPS on a 400Mhz Linino processor. A python micro web framework called Flask runs on the local wifi creating a web application of the mecosystem data. The purpose is to open a connection between the linux and the arduino so that the sensor data can be read and scheduling can set. So far in development the arduino sends a data event to the web server and the web server runs a static website.

Target Market

This product is designed for people who are interested in growing food in their home with an aquaponic system. This system is different than those on the market because it is more like a piece of furniture. Most aquaponic systems for home use do not use a glass fish tank or an embedded system for control.

Competition

Kijani Grows is the closest competitor out of Oakland, California. Kijani Grows sells systems with embedded technology that allows updates via twitter. The main difference that sets mEcosystems apart is that their system is large and would not be for indoor use. In California they may not have to supplement their plants with outside light but in Chicago it is necessary.

Nelson and Pade run an aquaponic consulting company out of Wisconsin. They do not make home use systems but create large scale commercial systems for large indoor farms. They are not a direct competitor but are considered the leading experts in aquaponics.

Home Users

The mEcosystem model aquaponic systems are designed to fit inside a users' home. The user would be able to grow fresh herbs and vegetables right in their own living space. Users that put this in their home are looking for a science experiment that yields edible results. Since aquaponics is still very new, it is hard to find information online about how to fix a problem in the system. This means that there is a somewhat steep learning curve to the system. Target users would also be familiar enough with technology to initiate code updates on the arduino.

Schools

Because of the symbiotic nature of mEcosystem, the ability to learn from the nature is extreme. Students in Elementary and Middle Schools are learning the biology and chemistry that can be seen within mEcosystem. Schools would buy a mEcosystem from us at mEcosystem Labs and then we would help teach a staff member exactly how it works. Schools would use mEcosystem as an educational tool to explain numerous biological, technological and even design elements to students.

Cost Analysis

Item	Q	cost	Total			
ELECTRONICS						
Arduino Yun	1	75	75			
power adapter	1	9.95	9.95			
ph	1	34.95	34.95			
temp	1		10			
humidity	1		5			
fans	6	5	30			
relay	5	2.95	14.75			
transstor	5	0.5	2.5			
10 k res	5	0.95	0.24			
1k res	10	0.78	0.78			
diode	5	0.92	0.46			
pcb prototype	1	30	30			
female to female jumpers	50+/1					
				E-total	213.63	
AQUARIUM						
water heater	1	17.66	17.66			
light -fish	1	14.96	14.96			
light - plant	2	100	200			

water pump	**1**	**20**				
air pump	**1**	**9.07**	**9.07**			
air stone	**1,2 piece**	**5.09**	**5.09**			
				A-total	**246.78**	
PLUMBING						
6" Container Bin 1/2 Size	**2**	**7.25**	**14.98**			
6" Container Bin Full Size	**2**	**14.49**	**28.98**			
4" Container Insert	**6**	**8.19**	**49.14**			
PVC Piping 1/2"	**2**	**1.36**	**2.72**			
PVC T 1/2"	**4**	**0.68**	**2.72**			
PVC Elbow 1/2"	**6**	**0.28**	**1.68**			
PVC Cap 1/2"	**2**	**0.38**	**0.76**			
					100.98	
STRUCTURE						
3/4" Birch Plywood	**3**	**44.98**	**112.45**	**S-total**		
					Grand total	**673.84**
MISC					**misc budget**	**100**
						~775
					tax	**1.1**
						$850

The total cost is about $850 for each mEcosystem. This seems like a lot for a consumer but there is a lot of costly parts in each system. The market price for a mEcosystem is between

$2000 and $2500 in order to make profit to continuously improve the product and obtain necessary equipment and facilities. With about 40 man hours per system, the profit comes out to $400-900 each. This assumes that each man-hour is worth about $20.

Future Work

In order to get this product to the market there is still much development to go. Firstly the embedded system must be tested and go through certifications to get approved. Secondly, the Linux web server needs to be able to dynamically change and update content while the system itself needs to be made cheaper and and with less labor involved in construction.

Embedded System

For the embedded system, the board would need to be a printed PCB board that can isolate the AC current from the DC current. Secondly the wiring to the outlets needs to be safer with non-stranded wires. The outlets should be changed to GCFI outlets incase water shorts the circuit causing a dangerous situation. In order to create the PCB for the embedded system the design must be sent into a company like fritzing which takes about 4 weeks. This board would cost about $40 each but reduce the labor time by at minimum 3 hours. The system once printed on a PCB can then be submitted for any federal safety testing by officials. This is a requisite for selling mEcosystems.

Code

There are a few important changes that are required to the code before it is ready for shipping. The primary one being creating a mEcosystem library that has an API that is a lot easier to understand as a user. currently there are hundreds of lines of code that all do various tasks. A library with a user friendly API would allow other developers from the community to get involved and improve the product. The Thermostat feature is still lacking in functionality. The best way to do this is to create a PID function that creates a dynamic duty cycle for the heater. This will ensure that the relays do not change state more than required as well as creating a stable and safe temperature change.

Secondly, the linux web application needs to open a communication channel to the arduino code. right now it can only accept the communication in one direction. The Linux server is good because to push an update to the server is as easy as running a script that re downloads the github code and runs the main python webserver method.

Conclusion

Aquaponics is the process that combines aquaculture with hydroponics. The embedded system controls all factors in the ecosystem required for the growing of food indoors. mEcosystem is currently being developed for home users as well as schools.

Appendix

Code

Arduino

https://github.com/channon1/mEcosystem/blob/master/mecosystem/mecosystem.ino

Linux

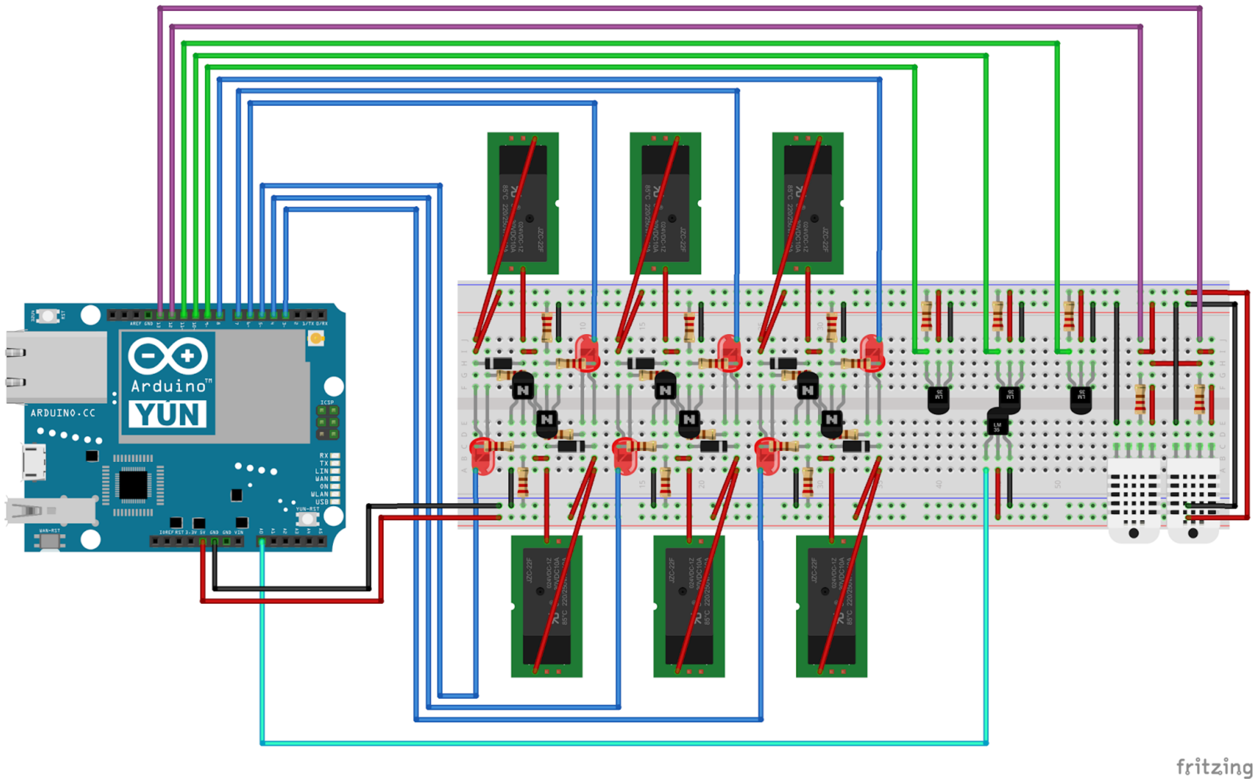
https://github.com/channon1/mEcosystem

Schematics

The construction of the system involves 4 levels; two plant levels, the aquarium level, and the storage or mushroom growing level at the bottom. In all of the levels there is 2 fans to circulate the system and strengthen the plants. The current prototype called Grow20 is below:


3A1
HOMER'S
BUCKET

The fritzing diagram for the embedded mEcosystem is below:



The acrylic case was made on the laser cutter with Autocad.