International Journal of Emerging Technology in Computer Science & Electronics (IJETCSE)
ISSN: 0976-1353 Volume 24 Issue 9 – JUNE 2017.

Environmental parameter monitoring and Data acquisition for Aquaponics

Akhil Nichani
B.Tech Student
Dept. of Electronics and Communication Engg.
SRM University, Kattankulathur
Chennai, India

Suchet Iyer
B.Tech Student
Dept. of Electronics and Communication Engg.
SRM University, Kattankulathur
Chennai, India

Angad Kumar
B.Tech Student
Dept. of Electronics and Communication Engg.
SRM University, Kattankulathur
Chennai, India

Ms. A. Ramya
Assistant Professor (O.G.)
Dept. of Electronics and Communication Engg.
SRM University, Kattankulathur
Chennai, India

Abstract**—We are moving into a world where we will have 9.6 billion living people by the year 2050. The biggest threat to humankind will not be war or economic collapse but instead will be famine. We need to be able to have food security for 10 billion people by 2050. Aquaponics is a new age method of farming which uses fish waste as the input fertilizer for plants grown in a non-soil based growing system. Using only food given to the fish, we can get two output products, fresh fish and fresh vegetables. This is an emulation of a natural pond ecosystem. It requires constant attention as there many variables that need monitoring to maintain the ecosystem. Monitoring on such a scale is a difficult task for any individual. Thus, we are designing and building a system for environmental parameter monitoring and data acquisition for Aquaponics. This system will monitor the various parameters throughout the system that determine the health of the plants and the fish. The basic parameters that will be monitored are pH, Lux, water temperature, air temperature and relative humidity.**

Keywords— Aquaponics, Data, Environmental

I. Introduction

Hydroponics is a type of Hydroculture, the method of growing plants without soil and instead using nutrient solutions in a liquid solvent. The plants are grown by exposing only their roots to the solution. Their roots may be supported by an inert medium such as gravel or perlite. The nutrient solution required to mix with the liquid solvent can be obtained from various sources including fish waste, duck manure or even normal nutrients.

Aquaponics and the Problem

Aquaponics is a system of agriculture that combines aquaculture and hydroponics to create a symbiotic environment that provides a much higher output than either of its individual environments. Aquaculture is a system where aquatic animals (fish, snails, prawn etc.) are raised in controlled environments for various purposes. Hydroponics requires nutrients to be mixed with a water solvent. Aquaculture is a perfect source for these nutrients (aquaponics).

The problem that arises with an Aquaponics system is the amount of data needs analysis and manipulation to optimize results. The waste that is obtained from the aquaculture environment needs to primarily be broken down into its base nutrients before it can be utilized by the hydroponic environment. Post this, the numerous environmental factors such as air temperature, water temperature, humidity, pH, light intensity etc. and other variables such as water level, water flow etc. also need to be monitored and any change in any of these variables requires a change in the environment. This becomes a massive task, and is quite difficult for any person to manage 24 hours a day, 7 days a week.

Proposed Solution

For the stated problem, we have the following solution. To handle the large data input that comes from an Aquaponics system, we have designed this project to automate the sensing of the required factors and maintain the optimal growth conditions. The environmental factors are observed by the means of various sensors such as temperature sensors for air and water, LUX sensors for light intensity and pH sensors for water acidity. This sensor data is then moved to the database via telemetry. The telemetry channel is created and maintained by an Arduino Uno development board and a Raspberry Pi development board, both utilizing an NRF 24L01 module. This data is then logged and analyzed and any discrepancies in the environment are identified. Once identified, a notification is sent to the person responsible for the environment, thus 2 involving the human only in this last phase of the system.

II. RELATED WORKS

In this section, we will discuss various articles and pieces of literature that are relevant to the concept of Aquaponics.

A. Ipanera - an industry 4.0 based architecture for distributed soil-less food production system

An IoT based system, the design illustrated in this paper [3], revolves around a custom architecture based on the Industry 4.0 standard. The multi-process system uses a network of DAQ and MCU to collect sensor data and monitor the various soilless farms. This information is wirelessly transferred to an IoT cluster and an OTA HTTP server. The required calculations are done by the fuzzy logic controller on the HTTP side.

B. Vegilab and Aquaponics indoor growing system

Fabrication of an Indoor, soil-less farm using the aquaponics method of farming has been researched and illustrated in this paper [1]. A small urban setup, the farm has a custom set of electronics to monitor the lighting and the water flow to the system. The application of a simple GUI to control the various parameters has also been shown.

C. An autonomous Aquaponic system using 6LoWPAN-based WSN

In an attempt to design a Wireless Sensor Network using the communication standard 6LoWPAN, this paper [2], illustrates the research required to design an IoT system for Aquaponics. Various sensors such as Nitrate, pH, and temperature sensors make the sensory network and the data acquired is uploaded to cloud for easy access and monitoring.

III. HARDWARE IMPLEMENTATIONS

A. Sensor Node Side

Figure 1 illustrates the flow of information from the sensors to the Arduino. The various sensors are depicted as blocks, connected to the Arduino individually. The Arduino is the controller that deciphers the information obtained by the sensors and formats it into a bit stream. The bit stream is then transmitted via the NRF Module to the receiver NRF Module on the receiver end.

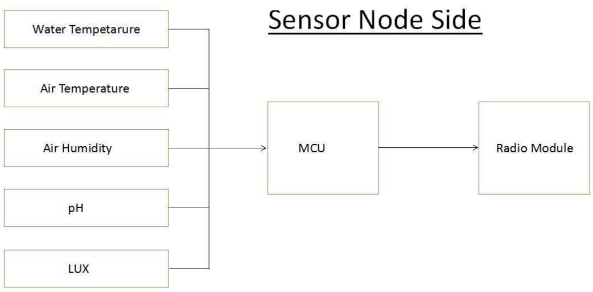

Fig. 1. Sensor Block

1) Light Sensor (BH1750): The BH1750 is an ambient light sensor IC. It uses I2C as it's communication protocol and has wide range and resolution (1-65536 LUX). It is a very low power IC running on 3.6VDC. In it's standard running mode it has a resolution of 1 LUX. It has other advantages such as an Illuminance to Digital Converter, no external parts, and a Light reject function of 50Hz/60Hz. The SDA pin is usually connected to A4 on the Arduino, the SCL to A5, and the ADDR pin to the A3 pin on the Arduino.

Fig. 2. BH1750 – Light Sensor

2) Air Temperature and Humidity (DHT11): The DHT11 is an air temperature and humidity sensor. It is a very reliable sensor and has characteristics such as a temperature resolution of 16 bits and an accuracy of 0.2 degrees, a relative humidity resolution of 16 bits and an accuracy of 1 percent. It is an ultra low-cost digital sensor using a capacitive humidity and a thermistor to take measurements. Its main disadvantage is that it takes reading once every 2 seconds only. It runs on 5VDC and the data pin is connected to Digital pin 3 on the Arduino.

Fig. 3. DHT11 – Air Temperature and Humidity Sensor

3) Liquid Temperature Sensor (DS18B20): The DS18B20 is a programmable-resolution one-wire digital liquid temperature sensor. It requires only one port for communication and can measure temperatures from -55 degrees celsius to 125 degrees 3 celsius. It has a default resolution of 12 bits but can also be programmed to 9 bits. It has an accuracy of 0.5 degrees celsius.

Fig. 4. DS18B20 – Liquid Temperature Sensor

4) pH Sensor: The pH sensor consists of 2 parts - the circuit board which consists of various Op-Amps and ICs to calculate the pH, and the pH probe. The pH probe is a glass electrode a sand core. It requires an input voltage of 5VDC and has a measuring range of 0-14. It has an accuracy of 0.1 pH.

Fig. 5. pH Sensor

5) Arduino Nano: The Arduino Nano is a small, complete, and bread-board friendly development board based around the ATmega328. Its functionality is similar to the Arduino Duemilanove. It doesn't have a DC power jack, and uses a Mini USB type B port to for power and as a replacement for the usual USB type A port. It can be programmed and deployed in different conditions without facing any issues. It has 14 digital pins and 8 analog pins for interfacing with different senors and modules as required.

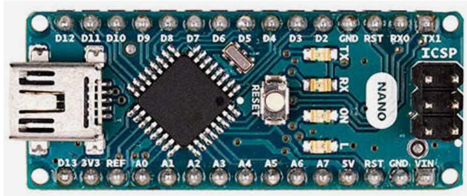

Fig. 6. Arduino Nano

B. Server Node Side

Figure 7 illustrates the flow of information from the NRF module to the server via the Raspberry Pi. As the bit stream is received by the NRF, it is then moved to the Raspberry Pi, where it is converted to information that can be displayed. This information is being displayed while simultaneously being moved to cloud, where it is logged for future reference.

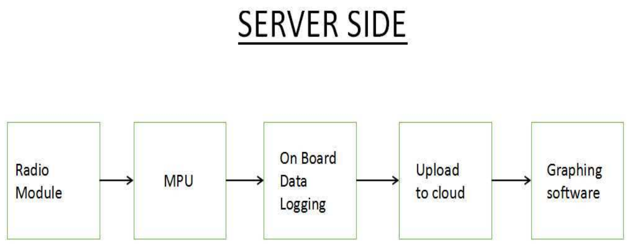

Fig. 7. Server Block

1) Raspberry Pi: The Raspberry Pi is a miniature computer. It is a low cost and low power processor with easy controller interfacing capabilities. It is very versatile and can be used for different applications as required. It uses a Broadcom BCM2836 SoC with a 900 MHz 32-bit quad-core ARM Cortex-A7 processor. I has 1 GB of RAM and 17 GPIO pins. It runs on 5VDC and can interface with almost any peripheral.

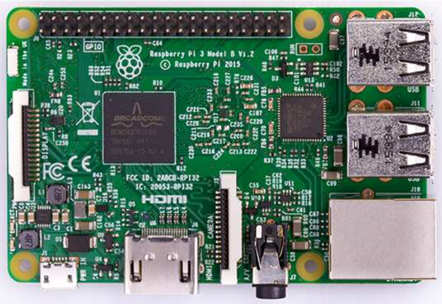

Fig. 8. Raspberry Pi 3

IV. SOFTWARE UTILIZED

A. Arduino IDE

The software used to program an Arduino is an Integrated Development Environment. It is based off the Processing Language project. The syntax followed is a mix of C and C++. The IDE makes coding the Arduino simple and manageable even for people with limited coding experience. The Arduino has been used in this project to interface all the various sensors with the microcontroller.

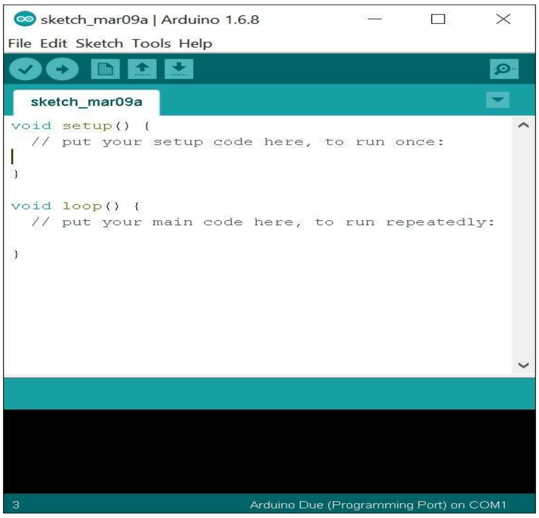

Fig. 9. Arduino IDE

B. Processing

Processing is an open source software that allows for programming of a visual interface. It is a uniquely built IDE for electronic arts and visual design. In this project, Processing was used to make a real-time display, which can be monitored on the Raspberry Pi.

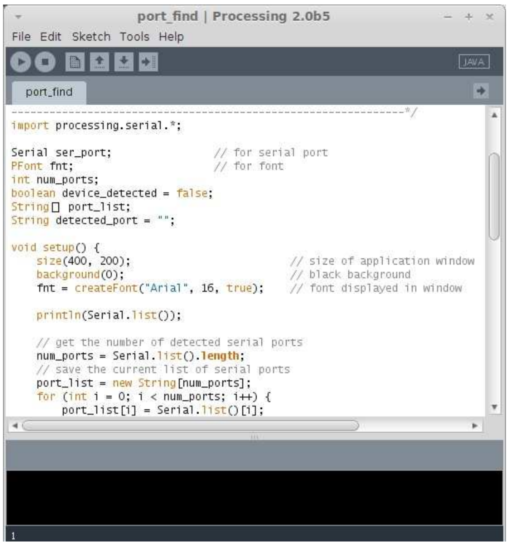

Fig. 10. Processing IDE

C. Raspbian

Raspbian is a fully customized Debain-based Linux OS for the Raspberry Pi. Not only is it an operating system, but it also has numerous packages which have all the tools required to get the Raspberry Pi working. We used Raspbian as the operating system for the Raspberry Pi.

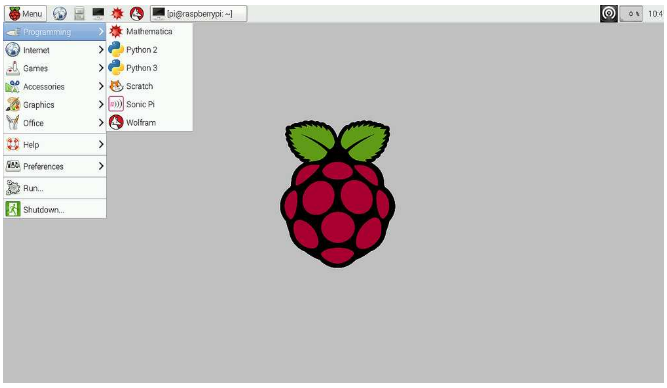

Fig. 11. Raspbian OS

D. Python

Python is another programming language for high level and general programming. It is very similar to C, C++ and Java but is easier to use. We used Python to run the codes on the Raspberry Pi. It interfaces the NRF module with the Raspberry Pi and uploads the data to the SQL server.

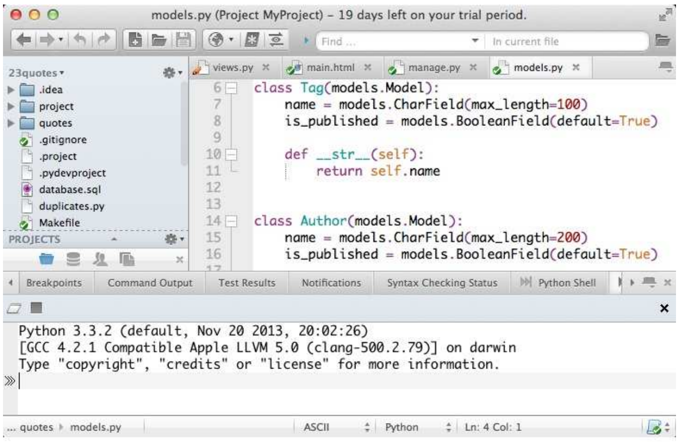

Fig. 12. Python IDE

E. Graphing and Database

We used multiple software to handle all the server side communications and data handling. The data was first uploaded to an SQL database. This data is then pushed to the cloud based graphing software, which is adept at giving a visual output to the data points provided to it. This setup makes it easy for us to track the environment from anywhere.

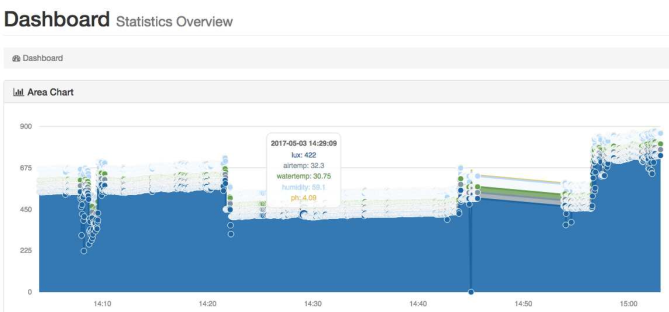

Fig. 13. Sample Graph

V. RESULTS AND DISCUSSION

A. Sensor Node Side

The results of our project have been shown in figure 14. As can be observed from the image, various sensors have been connected to the Arduino Nano, and have been placed in their required environment (Eg: Water Temperature sensor in a cup of water). While this setup is only for show, the actual setup will involve sensors in the field, in the areas from where we require inputs.

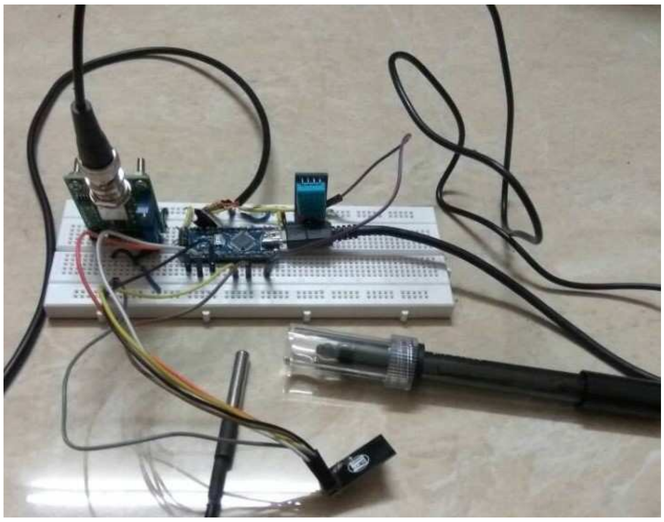

Fig. 14. Sensor Setup

B. Server Node Side

The results on the Server side have been displayed in Figure 15. The image shows us the electronics setup on the server side, and Fig. 16 shows what the final output looks like on the monitor. The GUI output is accompanied by the plotted graph output, giving the user more information to observe.

Fig. 15. Server Setup

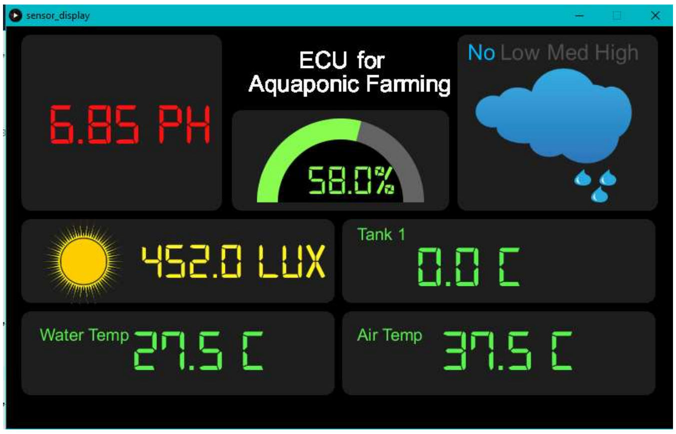

Fig. 16. User Interface displayed via Raspberry Pi 3

VI. CONCLUSION AND FUTURE ENHANCEMENTS

In conclusion, the setup performed as designed. The various sensors gathered their respective parameters from the environment, which consisted of a small container of water to simulate a fish tank for the water temperature and pH and the general climate conditions for the LUX, Air temperature and humidity. These were successfully transmitted wirelessly to the server node where three processes took place - The data was logged on the local data base, followed by which he data was also uploaded to the cloud for the graphing and visualization software, then the server displayed the live data on a dynamic display attached to the Raspberry Pi 3. The experiment and its results were verified by the required authorities.

A concept such as this always has room to advance further. Moving past wired communications, we have implemented wireless transmission of data in this project. The scope of this project is also still quite new as Aquaponics is not yet an emerging revolution in farming. The emergence of Aquaponics in households will result in a boom of application of technologies such as this, so that people need not keep an eye on their farms 24x7. As technology develops further and newer, faster ways to transfer information are developed, this system can further be improved. Eventually, a standalone, big data, system can be created to implement this on a very largescale farm environment. Recent developments in housing technology and concept, has lead us to believe that fully self-sustained environments that develop vertically, rather than horizontally are the future. These constructs will build upwards to save floor space on the Earth. In such an environment, systems like Aquaponics are crucial to provide sustenance for the Human race. Any future enhancements of this project will be useful in such a system.

REFERENCES

[1] Leatherbury, M. U. (2014), Vegilab and aquaponics indoor growing system. , 2014 IEEE Conference on Technologies for Sustainability (SusTech).

[2] N Hari Kumar, Sandhya Bhaskaran, S. H. V. K. (2016), An autonomous aquaponic system using 6lowpan based wsn., 2016 4th International Conference on Future Internet of things and Cloud Workshops.

[3] P.C.P De Silva, P. D. S. (2016), Ipanera - an industry 4.0 based architecture for distributed soil-less food production systems., Proceedings of the 1st manufacturing and Industrial Engineering Symposium.

[4] Rakocy, James E. et al (2013), Update on tilapia and vegetable production in the uvi aquaponic system., University of the Virgin Islands Agricultural Experiment Station.

[5] Santos, J. D. et al (2013), Development of a vinasse nutritive solutions for hydroponics., Journal of Environmental Management.