SEP600-AUTOMATED PLANT IRRIGATION SYSTEM PROJECT REPORT

Problem Statement

Traditional irrigation practices and manual watering methods employed by agricultural operations and plant hobbyists lack precision and regulation leading to scenarios where water is either wasted through over-irrigation or plants are put at risk due to under-watering.
This calls for an automated system to adapt to the varying needs of plants, providing a solution that aligns with modern expectations of efficiency, sustainability, and technological integration in plant care.

Background Research

The unsmart systems or the traditional way of irrigation is still quite popular in developing countries such as India,Nepal,Pakistan etc.There are techniques that people use,some of the common ways Byju's mention in their article Traditional Methods of Irrigation are as follows:

- Check Basin Method:Used in irrigation for leveled fields. The field is divided into basins according to the water capacity.
- Furrow Irrigation:Crops are planted in rows.A sprout is formed along the sides of the rows and the water flows between them
- Basin Irrigation Method:A raised platform connected with drains.

For plant hobbyists as well there is a high volume of water being wasted as most of the enthusiasts have no idea on the amount of water the plant requires.
As Byju's states rightly that these methods are based on old ways of thinking and can lead to water wastage.Also population growth, urbanization, and agricultural demand, is resulting in more pressure on water resources.Same goes for plant hobbyist who are into gardening,most of them have little idea on how much water a plant need. Hence there is an urgent need to replace the traditional methods with more sustainable and efficient modern irrigation technologies.

Solution

We call for a better substitute for the current traditional ways of watering plants/crops which can be easily scalable as well that is for small time plant owners to huge crop fields.We created a sensor based system that dynamically responds to the plants needs.
Here how it is advantageous over the other systems:

- Scalable:The system in itself is adaptable and scalable based on the type of plant and the requirement.
- Dynamic:The system responds in real time based on plants actual need.
- Sustainable:Since its dynamic it results in saving a lot of water which would have been wasted and could have resulted in harming the crop/plant as well
- Flexible:You could monitor your soil conditions by sitting at your home using a mobile application.Here we used Blynk mobile app to monitor our plant conditions

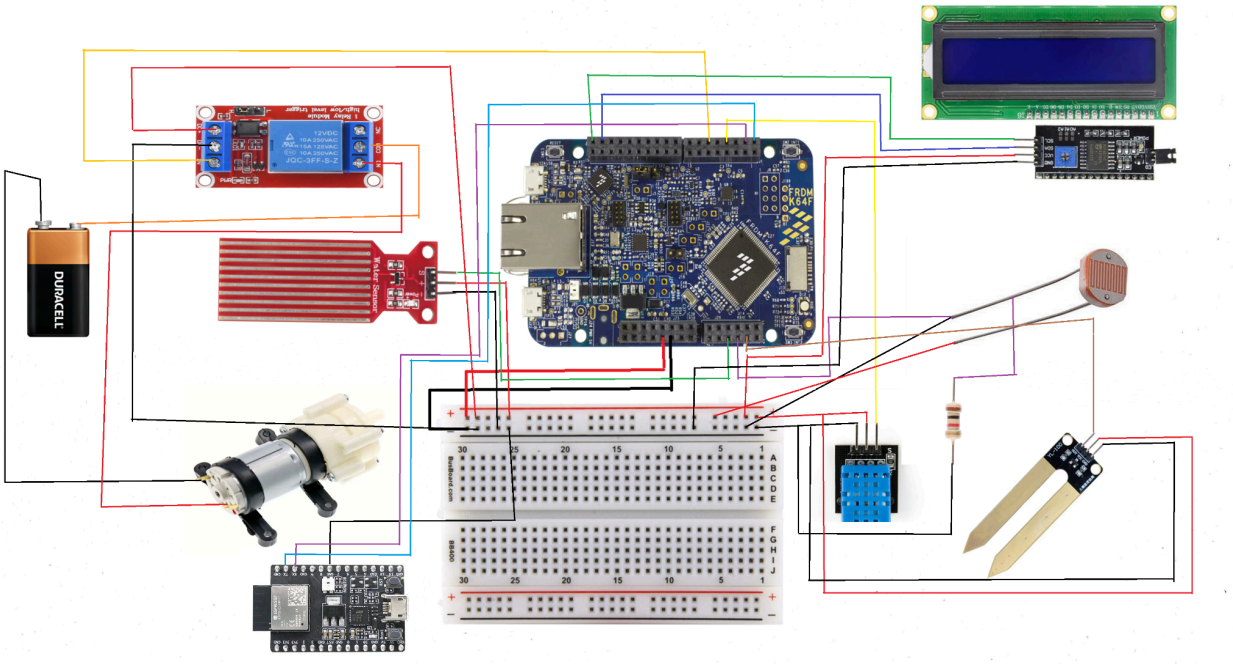
Project Work

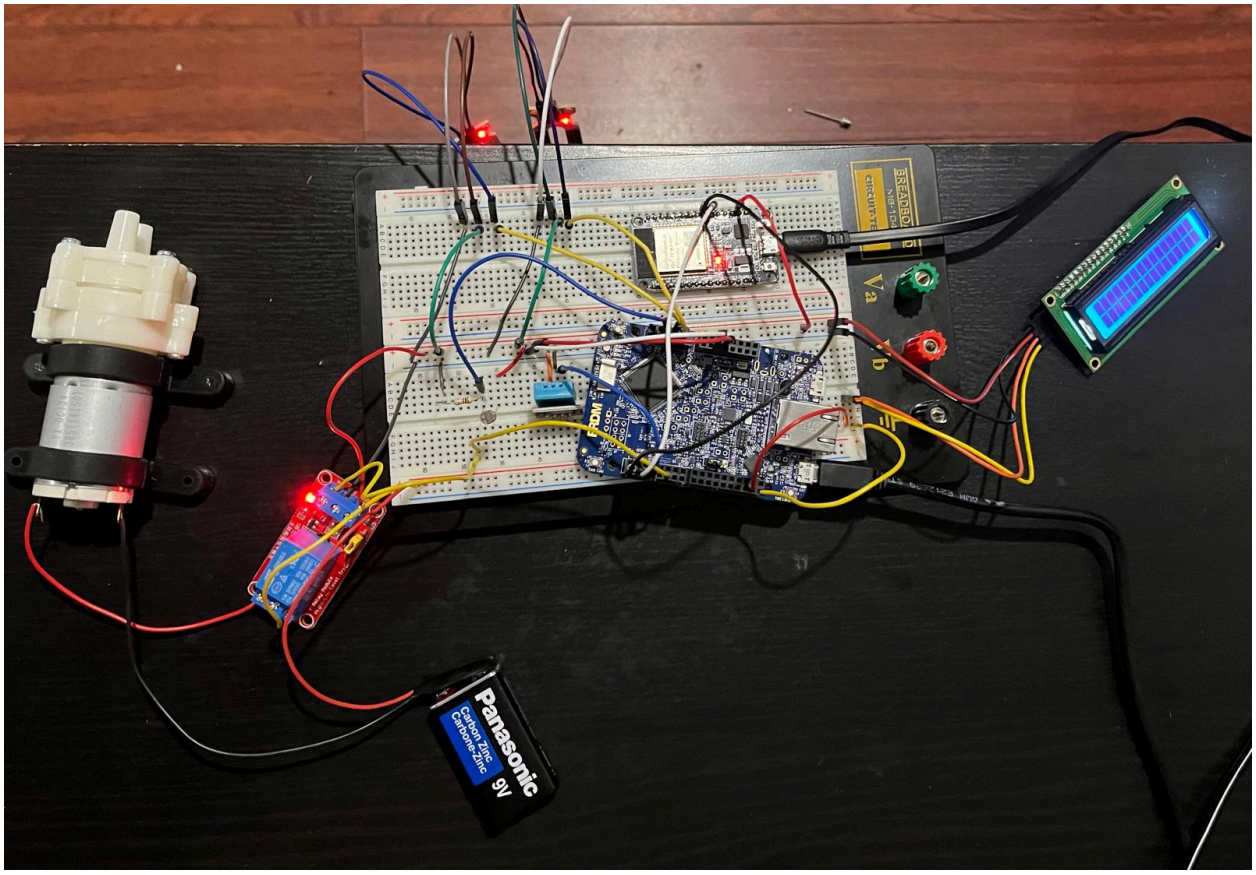
As we started our research on this project at the point when we were selecting the sensors we had to make sure that we do not select all the sensors that will be in close contact with water and soil as they are prone to corrosion. Therefore, we selected sensors like **DHT11 for temperature and humidity, Photoresistor for light sensor, and soil moisture, water level sensor.** For the display we are using an **I2C LCD display** and got the library of the LCD from our lab5 of our course and were able to display our readings and messages.
For the water supply we selected a **simple DC water pump, a relay module to control its functioning and an external 9v battery** for the pump power supply.

Most of the peripherals we bought from amazon.ca, but did not contain any documentation about the connections regarding the sensors so we had to figure it out ourselves and found one of the documentation for the water and soil sensors in soil and water sensor manual. And for photoresistor as well the connection and the code was similar to soil and water sensor as we connected to Analog input pin and just read the float values. For the DHT11 sensor we thought it would be the same as the above sensors but in DHT11 we were supposed to take two input values for both temperature and humidity. But we found out from the internet that the DHT11 sensor does not require analog input pins as it spits out a digital signal on the data pin from DHT 11 senor description. First we tried making our DHT 11 library it were not successful as the sensor were giving some magic numbers:

```
198
199  float DHT::ReadTemperature(eScale Scale)
200  {
201      if (Scale == FARENHEIT)
202          return ConvertCelciustoFarenheit(_lastTemperature);
203      else if (Scale == KELVIN)
204          return ConvertCelciustoKelvin(_lastTemperature);
205      else
206          return _lastTemperature;
207  }
208
209  float DHT::CalcHumidity()
210  {
211      int v;
212
213      switch (_DHTtype) {
214          case DHT11:
215              v = DHT_data[0];
216              return float(v);
217          case DHT22:
218              v = DHT_data[0];
219              v *= 256;
220              v += DHT_data[1];
221              v /= 10;
```

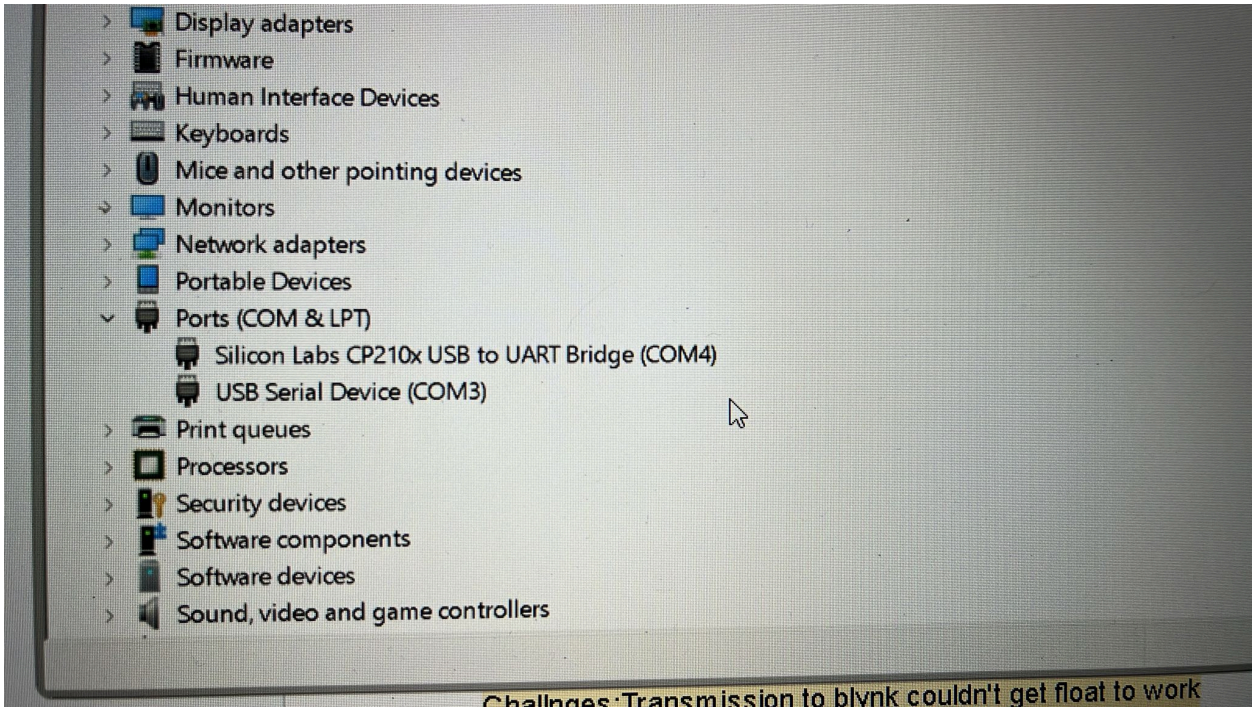
After some research we were able to find a library for the DHT 11 sensor and we were able to read the temperature and humidity values from the mbed website. At first we were having problems printing out the float values but from lab3 of our course we were able to fix it using the **mbed_app.json**. As we were using 8 peripherals there were also some issues with **wire management** and felt confused at times but were able to manage it cleanly after we made the circuit the second time.





Finally our main setup was running and our automated irrigation system was able to use the sensor values to start our pump according to those values using a relay module. After that we were to choose between a cloud service or communication between an external device like a computer or a mobile for visualization. At first we thought of using an **ethernet cable** as we already had one and use it to connect our microcontroller and a computer with it but it was a bit complex. Therefore we chose an **esp32 module** which had both wifi and bluetooth connectivity for external device connection. The connection was easy as we just had to common the ground of both FRDM K64 controller and esp32 module and connect TX and RX pins for the UART communication. And for the esp32 module we were using the arduino IDE as it has all the libraries we needed for the esp32 module. But for some reason when we tried powering up the esp32 module in one of our laptops and were not able to connect the esp32 module. We tried troubleshooting in the device manager but we were only seeing the FRDM K64 connection only in PORT section and in the Arduino IDE:

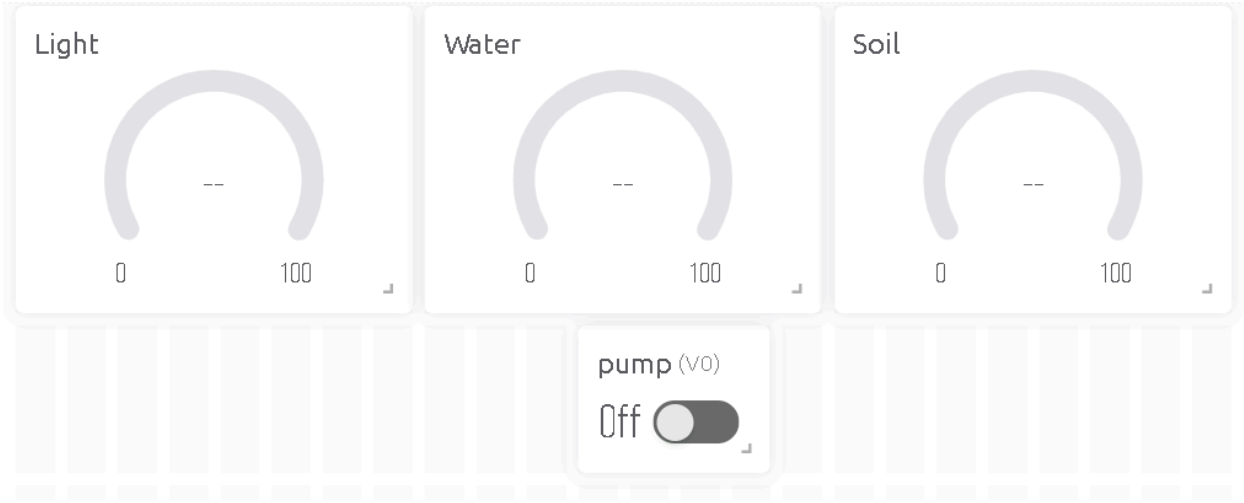
After some troubleshooting in the device manager we tried using a **different PC and Arduino IDE version 1.8.19 instead of the latest 2.3.2 version** and were able to connect it and was able to see another port of the esp32 module in the device manager:



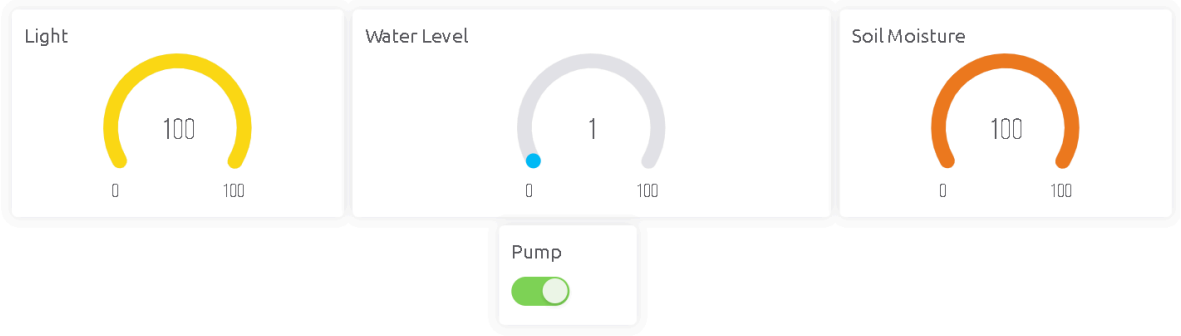
In the Mbed studio we used the file **BufferedSerial.h** and in the Arduino IDE we used **HardwareSerial.h** for the serial communication code. After that there was some issue with the values that we were receiving at the esp32 side as it was **printing out some weird symbols but after we saw that there was a discrepancy in the baud rate of the esp32 with FRDM K64 controller** and was fixed by setting the same baud rate. So we were able to send the data from our FRDM controller to the esp32 module.

After that we were to send the data to some cloud service or an external device so we used the Blynk Iot application for this part as it was a free cloud service provided for Iot applications. We were able to find documentation for the setup with our esp32 module in the Blynk website and implemented the code and were able to connect to the Blynk application but were **unable to**

receive any value that we wanted to display:



We tried troubleshooting the code with a simple program sending integer values to the Blynk application and were able to do that so we changed our transmission value to integer when sending it to the Blynk application as we thought there were some issues sending the float values. And the problem was fixed and we were able to visualize the data and control our automated irrigation system from the Blynk Iot application.



WORKING

Video

Health,Environment and Societal Consideration

The entire purpose of the project is to create a sustainable alternative to current practices.Here are some of the considerations:

- The system uses sensors to deliver water only when required , thus reducing wastage and conserving water – a critical aspect in regions facing water scarcity

- The system is designed to operate on low power reducing its carbon footprint.the prototype can integrate solar power.
- The system also introduces technological methods to rural areas which would promote more innovative ideas and technology to the people.

Future Work

As mentioned our project is extremely scalable and can accommodate both small time plant owners to huge farms.Some of the future work that can be done is

- Integrating Big Data:Using Data from all of these sensors we use big data analytics for more irrigation strategies
- Integrating Machine Learning:Using historical data we can train models to predict future watering needs
- Sustainable Energy:As mentioned this is a lower power device hence we can easily integrate renewable sources such as solar power to further enhance the system

References

"Traditional Methods of Irrigation." BYJU'S. Available: https://byjus.com/biology/traditional-methods-of-irrigation/. Accessed on: Apr. 7, 2024.

F. Sapa and T. Bisht, "SEP600-Project Proposal," Google Docs, Feb. 15, 2024. [Online]. Available: https://docs.google.com/document/d/1FGoe7cMq-cFtH1MnE4LcNTfJoyyfq61LlxeDmkZsjHU/edit?usp=sharing. Accessed on: Apr. 7, 2024.

K. R. Mancl, "Smart irrigation model predicts rainfall to conserve water," Cornell Chronicle, Jul. 16, 2019. [Online]. Available: https://news.cornell.edu/stories/2019/07/smart-irrigation-model-predicts-rainfall-conserve-water. Accessed on: Apr. 7, 2024.

Fritzing. Available: https://fritzing.org/. Accessed on: Apr. 7, 2024.

"Weeding Robots: Redefining Sustainability in Agriculture," HowToRobot. Available: https://howtorobot.com/expert-insight/weeding-robots-redefining-sustainability-agriculture. Accessed on: Apr. 7, 2024.

M. Rodell et al., "Machine Learning algorithms can predict irrigation patterns," Journal of Hydrology, Elsevier, 2024. [Online]. Available: https://www.sciencedirect.com/science/article/pii/S0378377424000453. Accessed on: Apr. 7, 2024.

“DHT - fork of DHT component - https://os.mbed.com/teams...,” Mbed, https://os.mbed.com/users/JohnnyK/code/DHT/ (accessed Feb. 28, 2024).

“Lab 3 : DAC and ADC - BSA Lab Manuals,” seneca-bsa.github.io. https://seneca-bsa.github.io/bsa/sep600/lab3/ (accessed March. 14, 2024).

“Lab 5 : Parallel LCD and Interrupt - BSA Lab Manuals,” seneca-bsa.github.io. https://seneca-bsa.github.io/bsa/sep600/lab5/ (accessed March. 14, 2024).

“mbedLCDi2c - LCD HD44780 library with I2C adapter for Mbed 6 | Mbed,” os.mbed.com. https://os.mbed.com/users/sstaub/code/mbedLCDi2c/ (accessed March. 14, 2024).