Final Project - Precision agriculture

de la Roza, Ignacio
Smith, Ashley

Internet of Things

Abstract

The goal of this project is to partially implement a smart agriculture gadget with a Fishino Piranha board. It can measure the environmental conditions of a particular plant, plantation or greenhouse, upload them to a remote database and control actuators to maintain the environmental conditions in the desired range. The implementation is called *partial* because the real actuators are not used; instead, a set of LEDs indicate the action that would take place.

Contents

1 Project overview

1.1 Operation

The general operation of the device is depicted in figure 1. During the setup, some parameters are defined and the WiFi connection is made. After that, and endless loop is repeated. The *happy path* consist of a measurement, the activation/deactivation of actuators to make environmental parameters fall into the desired range, and sending the acquired data to the database via an HTTP request. In case of a lost measurement, the value is estimated using a moving average. If too many consecutive measurements fail, it is assumed that the sensors are not functioning correctly and thus the actuators are shut down for safety reasons and the user informed of the situation. If the sensors start to function correctly again, the *happy path* is resumed.

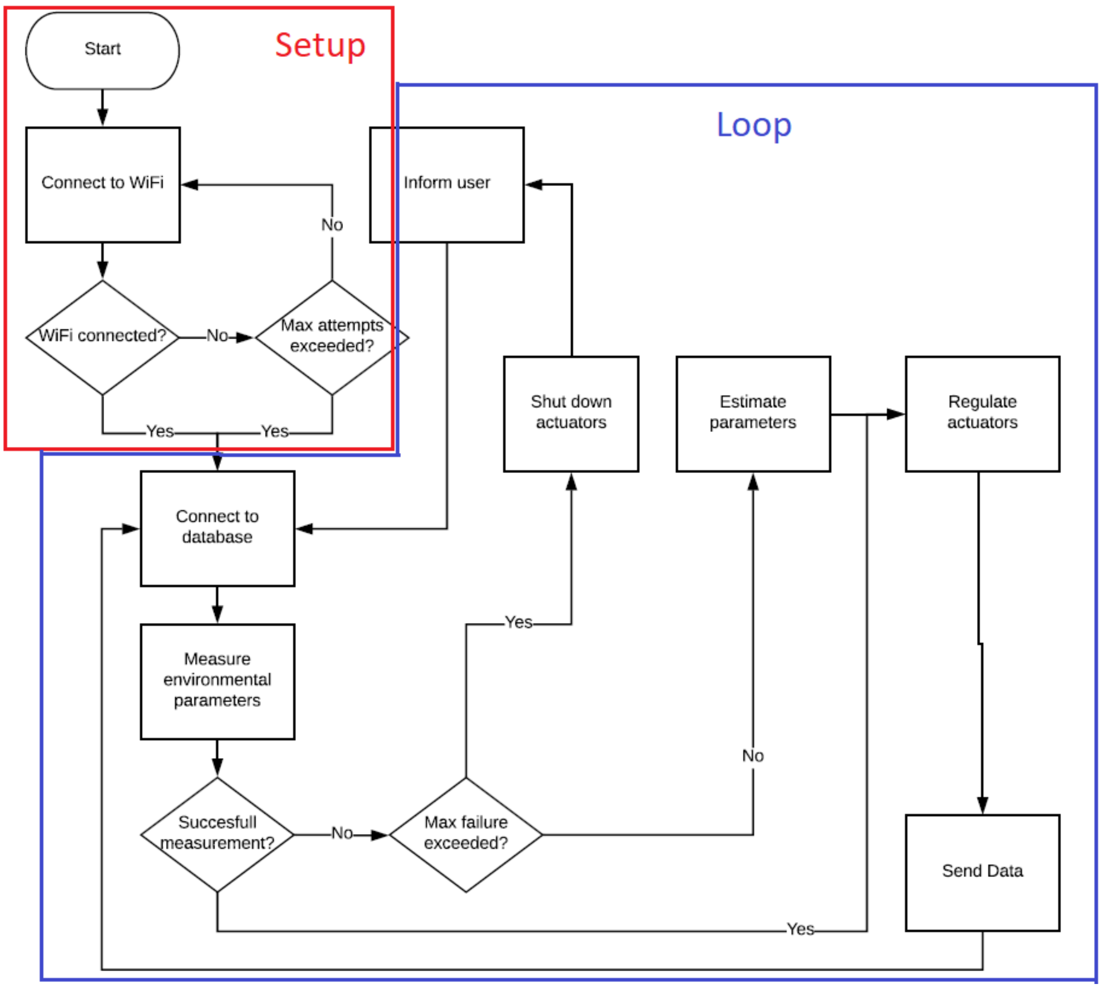

Figure 1: Algorithm implemented

1.2 Hardware used

The list of the parts used is the following:

- Fishino Piranha board
- DHT11 temperature and humidity sensor
- Proto board
- Flat cables
- Computer [1]
- 5V USB power source
- USB to micro USB cable (also used for flashing the device)
- LEDs
- 100Ω resistors

The setup can be seen in figure 2

Picture of the device

Figure 2: Setup used

1.3 Flashing the device

In order to upload the code to the Fishino, the Arduino IDE was used. The drivers for the Fishino Piranha, as well as some Fishino libraries, used in the code, such as *Wireless Fishino*, were all downloaded from the fishino website www.fishino.it. Once all the software is ready, in order to flash the device, it must be connected to the computer with a USB to micro USB cable, the reset button of the Fishino has to be pressed until the *reset light* starts blinking, and then the code can be uploaded through the Arduino IDE.

[1] used to flash the device, store the database and visualize the data. In a real deployment, these three functions would be carried out by different entities

2 Enviromental sensing and control

2.1 Sensors

In this implementation, the environmental parameters measured and controlled are temperature and humidity. This is done with the DHT11 sensor. The physicall connection costists of a shared ground and V_{cc} (5V) with the Fishino board and a single cable through which data is sent serially into the digital pin two (D2). Its acquisition is done periodically, with the same period for both magnitudes and with a 16 bit resolution. It's worth noting that adding another sensed parameters would be very straighfoward and could be done with little modifications to the code. Also, while the control is applied on the same parameters that are sensed, this is not needed, and not controlled parameters could be registered if the extra information was usefull or desirable.

2.2 Estimation and control

Long term data is stored in the data base. The device only retains a small number of samples used for control and prediction. An *on-off control* has been implemented, but the locally stored data could be used to implement algorithms adapted to particular plant dynamics [2].

In case of sensor malfunction, the lost measurement is estimated as the average of some of the previous samples. This estimated measurement is then used to assess which actuators should be activated or deactivated. If too many consecutive measurements are lost, estimation becomes pointless, for the error can grow arbitrarily large. In such a case, actuators are turned off for safety and economy reasons and an *alarm* is turned on, as to alert the user that the device needs maintenance.

3 Database and visualization

3.1 Storage

To store the data, we use the open source time-series database system InfluxData. One of the reasons this particular database was chosen is the fact that it provides integration with data analytics and visualization software as Grafana.

The database was created on a laptop that is connected to the same network as the IoT end device. As data is generated, it is sent through HTTP requests with the format shown in figure 3

[2] the word *plant* here is used as a general control theory term, not as *plant* in botanics.

HTTP format

Figure 3: HTTP request format

When an error occurs and the system shuts down because it doesn't have enough information to act properly, it sends a special packet belonging to a different series that triggers an alarm and informs the user.

3.2 Visualization

The Grafana software provides a solution for the visualization of the time series data stored. A dashboard is created from which the complete history of the monitored environmental parameters can be accessed and easily interpreted. Also, measurement errors are recorded, so that the user can interpret if a sensor is malfunctioning and be specially notified if the failure causes the device to lose control of the environment. The dashboard can be seen in figure 4.

Grafana screenshot

Figure 4: Grafana dashboard

4 Conclusions

The project goals have been reached. The prototype has been successfully implemented and all of the features tested. The device can measure, estimate and control the environmental parameters as desired, as well as upload the data to the database, from which it can be visualized by the user.

The control algorithm and actuator control were correctly implemented, and although it must be noted that a fully functional implementation would possibly imply revisiting their design, this can be done with no mayor modifications to the code, but just some to the specific functions that describe the prediction and control. It was considered that these details, more related to the architecture of particular greenhouses and the field of botanics, exceeded the scope of this work.

In this work, the power of tools such as InfluxData and Grafana, as well as pre-built boards, together with their IDE, is clearly depicted. They allow a straightforward and fast implementation of IoT solutions.

Finally, for future works, the analysis of other wireless connections, as well as battery optimization could be usefull, and possibly necessary, for real outdoor scenarios.