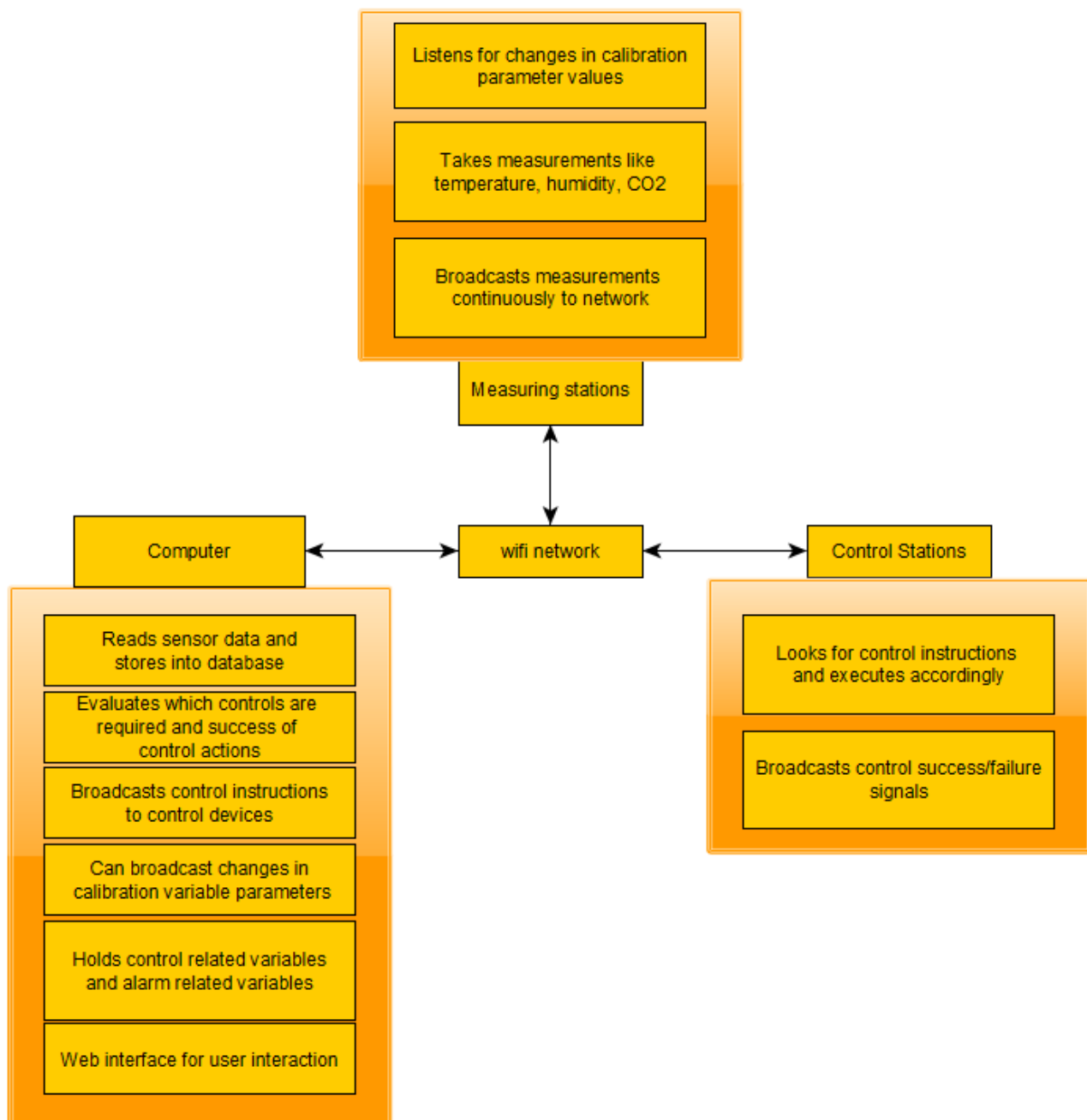


Building a DIY control infrastructure for a hydroponic crop: Part one

Controlling an entire hydroponic crop using electronics is not a trivial task. This includes everything from the automated control of things like relative humidity and ambient temperature, to other variables, such as lights, solution pH, conductivity and temperature. Many paid solutions exist in the market, but, in my experience, none of them offer enough flexibility to accommodate all potential environments, as all the ones I know are closed source and do not allow users to readily modify the firmware/software used to fit the user's particular needs. Through the past 5 years I have setup control infrastructures across several different crops and have usually done so using an entirely DIY infrastructure that focuses on flexibility and power for the end user. In this post I want to talk about how this setup usually works and why I came to these design choices.

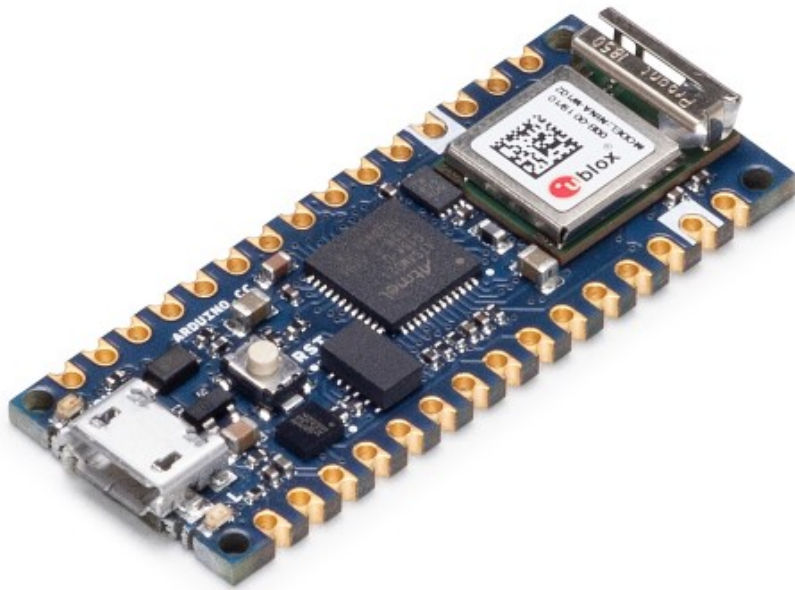


Usual network configuration I used to built electronic monitoring/control infrastructures for hydroponics

In general the infrastructure I setup relies on the use of wifi for the communication of the devices. This is because it's usually the easiest to setup, although it might not be the most power efficient or the most desirable in all cases. I generally divide devices into three camps. There is a main device – which is usually a capable computer – which serves as the “central hub” for the entire setup. This computer contains the main database that stores all information about devices, sensor readings, calibration variables, alarms, etc and is in

charge of deciding which control actions to take given the sensor reading it is receiving and the control devices it has access to. This central computer usually hosts a website as well, where the user can easily modify things, issue manual control actions, add new devices, set up alarms, etc. The computer can be duplicated as well, to prevent this from being an important point of failure. In several cases we have used a raspberry pi to play this role.

The second and third group of devices are usually Arduinos whose main role is to either take readings (measuring stations) or execute control actions (control stations). Some arduinos might actually serve both purposes as an arduino can often be fit with things like pH/EC probe readings as well as relays that control peristaltic pumps that are used to push pH up/down or nutrient solution into a solution tank. It is worth noting that the decision of what to do for control is never taken by any control station but all they do is interpret control messages from the computer and then try to execute those actions and then give back some response of what's going on. Measuring stations, on the other hand, are only trusted with the task of taking some measurement from the environment and broadcasting it to the network, the only thing they might listen for are messages issued by the computer to modify their calibration, whenever this is required.



The arduino nano includes wifi capabilities and offers a very convenient low-power core for measuring stations that do not require high power to operate sensors

Measuring stations can be fully customized to have as many reading as the user desires and can be implemented to do all sorts of things, from measuring temperature and humidity, to measuring air-flow, to measuring media water content. This allows for dozens of different temperature and humidity reading spots using different kinds of sensors, to monitoring things such as irrigation flow and solution ORP and dissolved oxygen values.

The entire setup relies on the use of the mosquitto (mqtt) protocol in order to have each device broadcast a specific topic with a specific reading that other devices can subscribe to. The computer will listen to all the devices it sees within its database and it is therefore easy for a new device to be added by a user, since it only requires the inclusion of the device into the database. The measure/control stations can subscribe to the specific topics their interested in for calibration or control actions and can act whenever they receive these messages. All the devices are automatically added to a web platform and alarms can easily be set for any of the measurements carried out by the measuring stations.

A big advantage of this approach is that control actions can be made as complex as the user desires. This includes doing things like implementing reinforcement learning based controls for things like temperature/humidity allowing the computer to use a wide array of measurements in order to take control actions, not relying solely on the measurement of one limited sensor to make these decisions. This allows a computer to use information such as outside temperature to decide how much air it wants to get into the facility for control, or how long it wants to turn on humidifiers in order to allow the desired level of humidity within a grow room.

With all this said, DIY control is definitely not the easiest route to take. Implementing something like the above will require the purchasing of a lot of different electronics – which are sometimes expensive depending on what the user wants – and does require a lot of time programming firmware and deploying software so that the desired outcome can be achieved. With that said, the unparalleled level of control is often worth it and can be accompanied by substantial gains in the information available to the user, which often leads to improvements in yields and the significantly quicker catching of potentially important problems.

On the next part of this post, I will talk about some of the practical aspects of this project, such as which arduinos and sensors I usually use and how these are setup to communicate with the central computer. If you want to learn more about how I can help you set this up for your crop please feel free to contact me using the website's [contact form](#).