

Beehive Monitoring System

Clayton Gumbrell, Canberra, Australia
October 2022

Entry in the WizFi360 Design Contest, 2022 (<https://maker.wiznet.io/contest/wizfi360>)

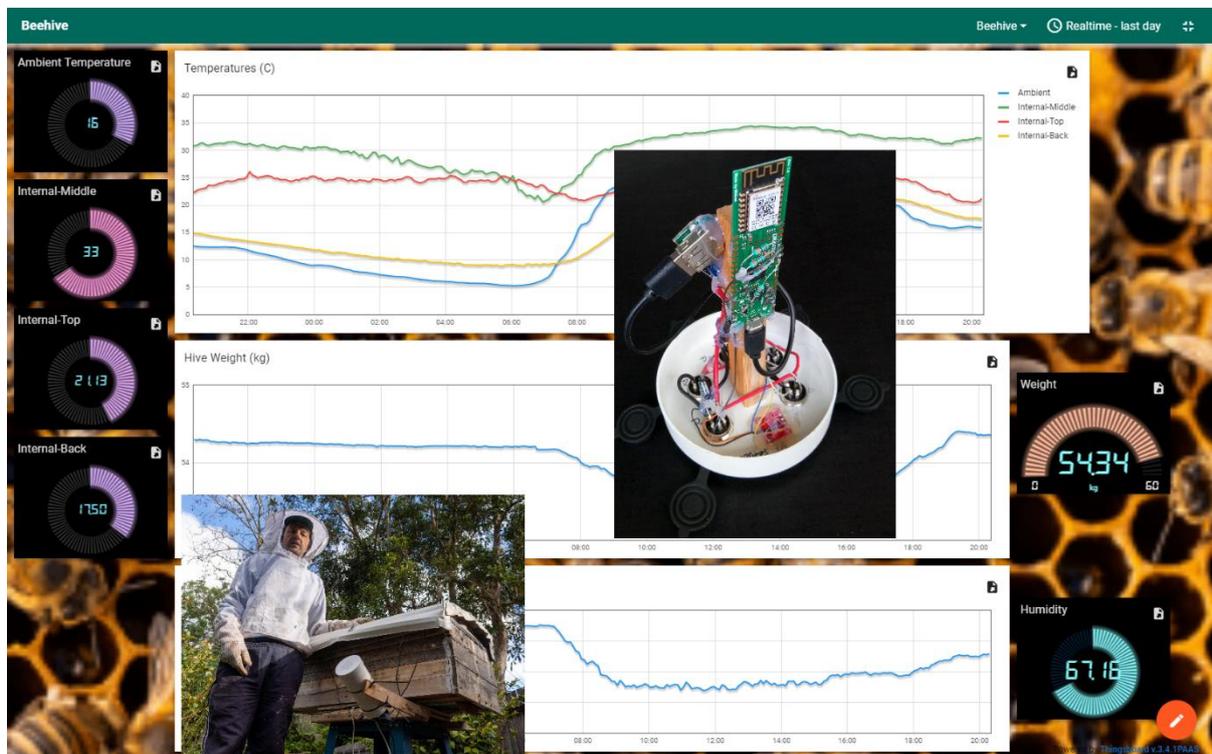
Summary

As a hobbyist beekeeper, I thought that a system to monitor a beehive would provide a useful tool for a beekeeper. The primary useful data is the weight of a hive, as it allows the current honey load to be determined, and the need to visit the hive to extract the honey. However, some additional sensors would also provide some useful information. The Beehive Monitor System monitors the following data

- Beehive Weight
- Ambient Temperature
- Ambient Humidity
- Internal Hive Temperature (in three locations)

The system uses the MQTT protocol sent over Wifi, and logs data to the 'ThingsBoard' cloud service (<https://thingsboard.cloud/>). 'ThingsBoard' allows the data to be stored, and then used to update widgets on a webpage. My system displays the current readings, as well as a timeseries over a selectable timespan.

The live system is available (as of Oct 2022) at <https://thingsboard.cloud/dashboard/3d8a3870-5767-11ed-a40f-6527c4d13844?publicId=8cc98b20-5767-11ed-9bfe-295761657fe1>

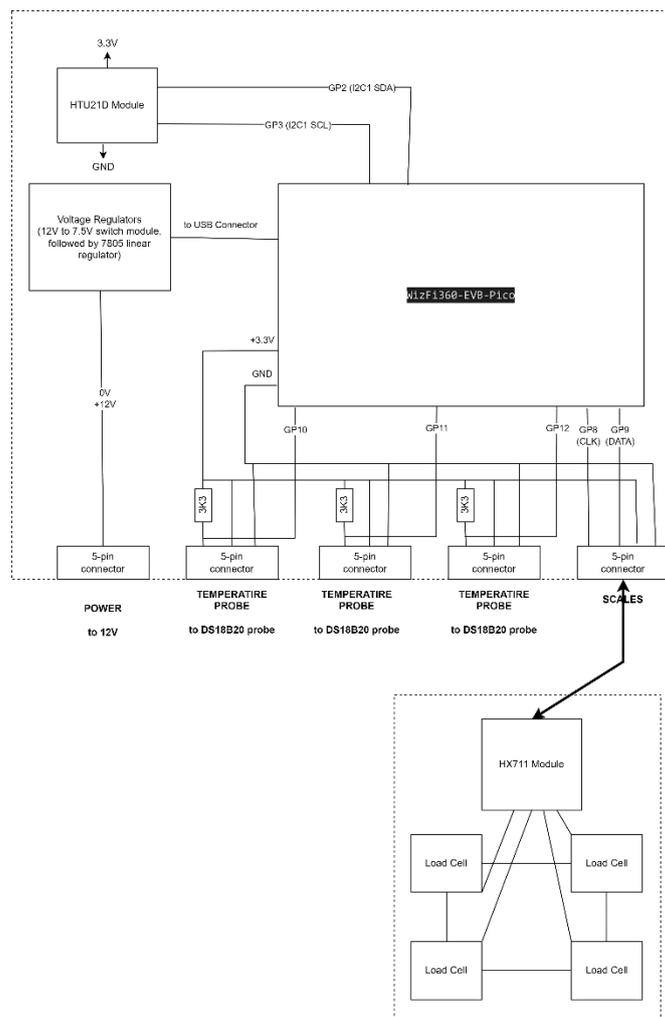


Hardware

The following HW devices were used:

- WizFi360-EVB-Pico
(<https://docs.wiznet.io/Product/Open-Source-Hardware/wizfi360-evb-pico>)
- HX711 load cell amplifier module and four load cells
(<https://www.sparkfun.com/products/13879> and
<https://www.sparkfun.com/products/10245> or similar)
- HTU21D Humidity and Temperature Sensor Module
(<https://www.sparkfun.com/products/13763>)
- DS18B20 1-wire temperature probes (three of)
(https://www.auselectronicdirect.com.au/waterproof-temperature-probe?gclid=Cj0KCQjwwfiaBhC7ARIsAGvcPe59dlib6Ym7jRV8cdfq4YUC_919gWqb9n32ViwaJuLSsjYDgZDz48aAIYjEALw_wcB or similar)
- Power supply regulator – switching regulator (for 12V to 7.5V) followed by a 7805 5V regulator
- Connectors, housing (drainpipe fittings), 12V gel-cell battery, misc components.

These were connected as shown in the circuit diagram.



After the initial prototype was developed, the device was housed into a length of drain tube to provide a watertight enclosure that was still RF transparent. Waterproof 'Aviation' connectors were used for the sensor and power (with a pin removed and block off to ensure the power connector was plugged into the correct location). The Humidity/Temperature sensor was mounted in a hole in the housing to expose the sensor to the outside environment. The Pico device and voltage regulators were mounted on an internal wooden bracket off the front panel and wired in place. This allowed the whole assembly to be removed in one piece and minimize the holes in the housing. Liberal amounts of hot-melt glue secured the components and wiring.

Initially the power supply is a 12V Gel-Cell, but this will be augmented with a solar panel and charge controller.

The weight scales were made of two plywood sheets, with the load-cells mounted in recesses. The HX711 amplified was also mounted in this assembly, with only digital connections to the monitor device. The scales were calibrated using a 10kg weight to determine the raw readings, and from this were calculated appropriate offset and scale factors.

Software

The software is available at https://github.com/claytoninc/WizFi360_BeeLogger

It is based on the WizNet example TCP-Client code from <https://github.com/Wiznet/WizFi360-EVB-Pico-C>

Initially there were problems in getting the system to compile, but after a few tweaks of the CMake file to correct the case of directory names, the code compiled in a WSL environment using GCC 9.2.1 arm-none-eabi (following the RPi pico starters guide).

The code was further developed using other libraries, including Arduino libraries. Some were used directly with little change. Some were only used as reference for developing the final code.

The required MQTT implementation only needs to connect and log data – so only a very limited set of the MQTT features. Simple code was developed to do this feature, based on the ‘ThingsBoard’ sample projects (especially the ‘PubSubClient’ Arduino library)

The one-wire library (for the DS18B20 sensors) used the library from <https://github.com/adamboardman/pico-onewire>

The HX711 (weight sensor) was written, with reference to a number of HX711 Arduino libraries

The HTU21D (humidity/ambient temperature sensor) used the Sparkfun Arduino library (https://github.com/sparkfun/SparkFun_HTU21D_Breakout_Arduino_Library), ported to the RPi Pico

The operation of the system is:

- connect to the Wi-Fi (or confirm the connection)
- take a measurement from all sensors
- connect to the MQTT server
- transfer all valid sensor data to the server
- disconnect from the server
- delay for 20 seconds
- repeat

A simple software loop performs these operations. A watchdog timer ensures that should a problem occur, the system will reset and start again.

Installation

The beehive was removed from its stand, and the scales installed before returning the hive in place.

The Monitor device was mounted to a bracket off the side of the hive, orientated with the connectors down to minimize the risk of water leaks.

The hive was opened, and the three temperature sensors installed in the middle of the hive bee cluster, at the top of the hive, and at the back of the hive.

The 12V battery was connected, and the monitor began logging to the cloud.

Results

The system has only been logging for a few days, but already intriguing data is shown. Once the day warms up, the weight of the hive drops as the bees go out to forage. There is even a weight drop across a period of a few hours – possibly when the bees find a good source of nectar and communicate this to the rest of the hive. On a rainy day, the hive gained weight - presumably due to the water on it.

As expected, the cluster temperature stays warm, with other parts of the hive cooler.

There is some drift in the accuracy of the scales, but as we are interested in trends rather than specifics, this should not be a problem. Over time it is expected that the hive weight will gradually increase as the honey stores increase.

Conclusion

This project has achieved all that was expected. I expect to build a few more of these for my other beehives.

There is always room for improvement. The power supply is probably the main area that can be improved. A solar panel could make the system operate stand-alone. Alternately the software could be adjusted to log a lot less often and sleep otherwise, and so make dry-cell battery operation feasibly.

It will be interesting to see what insights into bee behaviour that this monitor system will reveal as more data is collected.