Private Cloud With Wireless Internet Of Things (IoT)

Team Olympus: Clayton Clouser, Daniel Dorsey, Daniel Potichko, and Jess Stacy
Advisors: Robin Pottathuparambil, PhD and Chatwin Lansdowne
Department of Computer Science and Engineering, University of North Texas

Background

- In each mission, sensors have taken data periodically or episodically and create many data streams of importance which must rely on the current space communication architecture.
- Current space communication architecture is characterized by anemic on-board processing platforms and the bottleneck and latency in the backhaul to Earth.

Objective

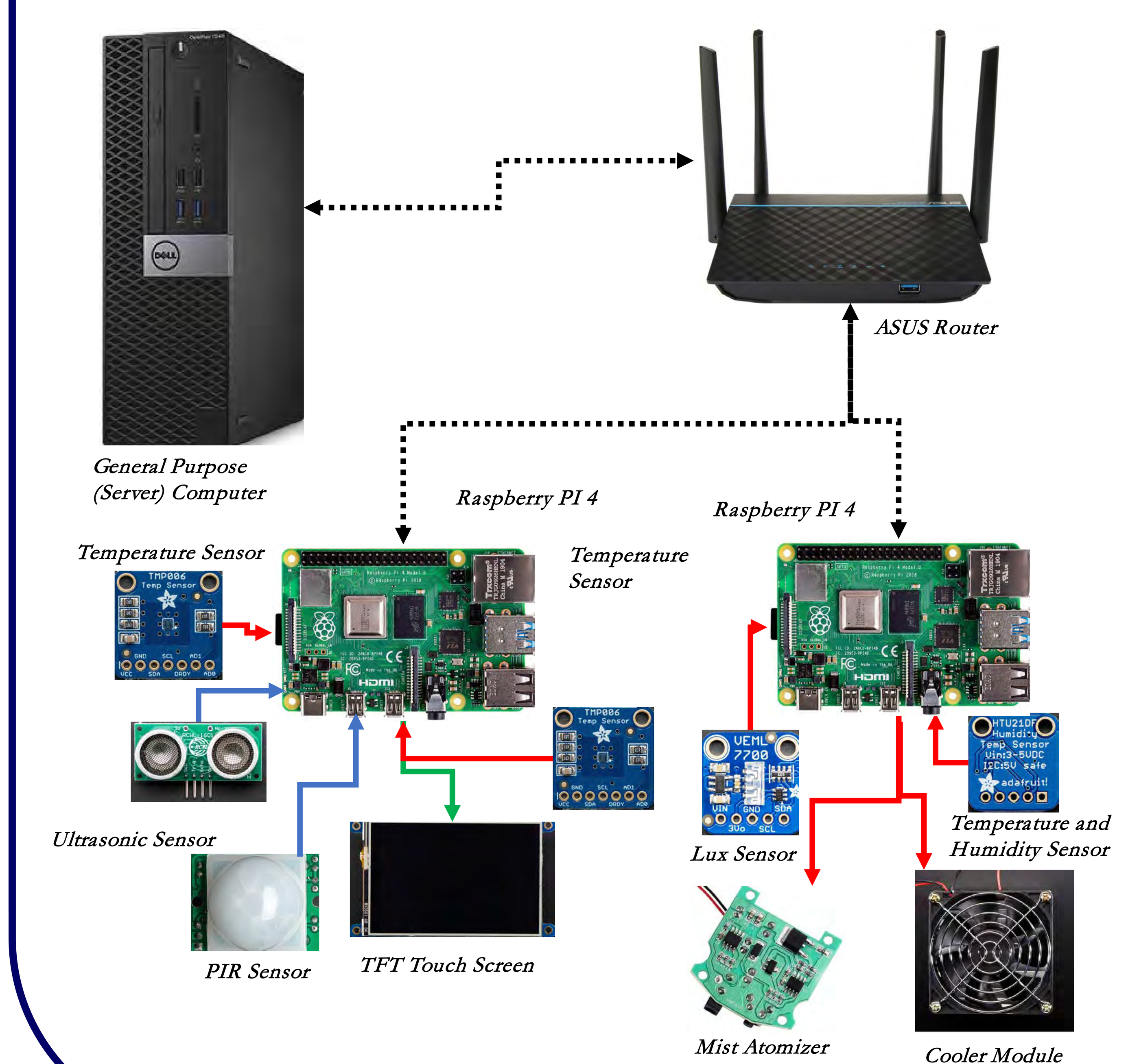
The objective of this project is to:
- Examine an open source private or proprietary IoT cloud package and evaluate one through demonstration.
- Create a private cloud network that collects and process data from sensors through a microcontroller.
- Create applications that demonstrate the ability of the cloud to process sensor data and actuate certain output devices.
- Have a IoT cloud package that allows the user the ability to view and analyze current and past sensor data

Design

The Design for the Cloud Service
- The IoT Cloud Package used in this project is ThingBoard.io running on a general purpose computer in CentOS 7.
- The sensor data is taken from a Raspberry Pi 4 and then sent through a ASUS Router to the general purpose computer using MQTT protocol. The user can then create and view graphs that display the real time data.

Smart Door and Vivarium Application
- ThingsBoard.io will send an RPC call to the Raspberry Pi when it detects a human hand, or when it needs to adjust the vivarium's environment.
- The RPC call goes out when there is a notable difference in distance, temperature, and motion inside the smart door's entry way. With the vivarium a different RPC call goes out when there is a change in light, temperature, or humidity that adjusts the environment of the vivarium.

System Diagram



Wireless Connection
Connected via Wires
I^2C
SPI

System Components
- General Purpose Computer
- 2 TMP006 Sensors
- 1 PIR Sensor
- 1 TFT Touch Screen
- 2 Raspberry PI 4
- 1 ASUS Router
- 1 Ultrasonic Sensors
- 1 HTU21D-F Temperature & Humidity Sensor
- 1 VEML7700 Lux Sensor
- 1 Peltier Thermo-Electric Cooler Module
- 1 WHDTS Ultrasonic Atomization Maker Mist Atomizer

Implementation

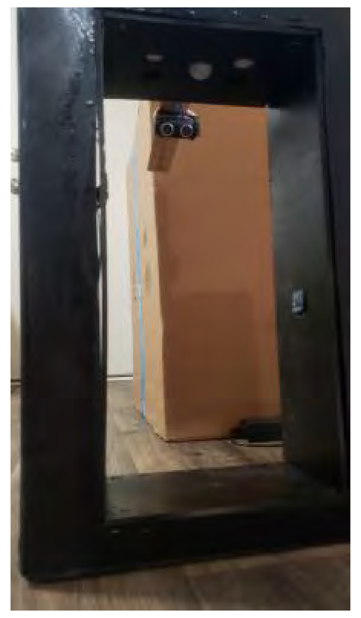


Figure 1: Model of the Smart Door

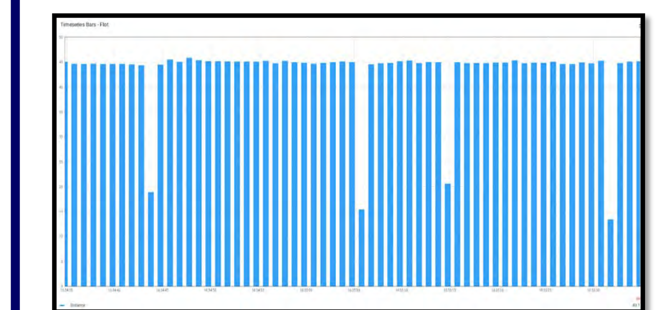
The cloud system and sensors are used in a smart door application to detect and when someone goes through the door on a smaller scale. The cloud system is also used to self adjust an environment in a vivarium.

Figure 1. shows the smart door and the placement of the sensors.

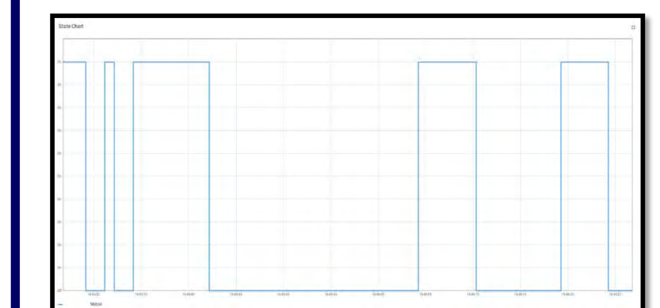
Figure 2. shows the vivarium that the second application monitors.

Figure 2: The Vivarium

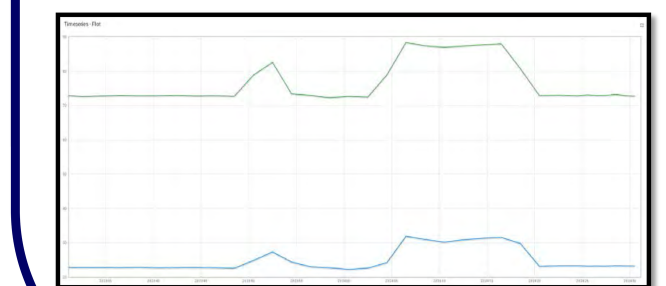
Telemetry of ThingsBoard



Plot 1: Plot of Time vs Distance detected by ultrasonic sensor.



Plot 2: Plot of Time vs Motion Detected by PIR.



Plot 3: Plot of Time vs Temperature Detected by TMP006.

With ThingsBoard we tested each sensor and graphed the data for each scenario. ThingsBoard can show each one with bar graphs, analog signal graphs, and line graphs. These are some of the graphs from application 1.

Plot 1 shows the distance detected by the Ultrasonic Sensor.

Plot 2 is the analog graph for the PIR motion values over time.

Plot 3 is the line graph that shows the current trend of the temperature being picked up by the TMP006.

Conclusion and Future Work

We have found a working IoT platform that supports a private network along with creating a private network the uses MQTT protocol to allow a microcontroller to connect, send, and receive data from ThingsBoard. The current system allows a user to connect a microcontroller to the private network and connect to ThingsBoard.io with MQTT protocol. We have also created two different case studies for ThingsBoard where we tested a passive system that can be used to self-sustain a closed environment and an active system that test how ThingsBoard reacts to controlled actions.

For future development with ThingsBoard a team would be able to implement remote connections to the private ThingsBoard network. This would allow the team to simulate the connection between a team on Earth connecting to a closed system in space.

Acknowledgements

Faculty Advisor: Robin Pottathuparambil, PhD
Lab Manager: Alejandro Olvera
NASA Sponsors: Dr. Tim Urban and Talia Jurgens
NASA Mentors: Chatwin Lansdowne
UNT Computer Science and Engineering Department